

This manuscript in post-review format was accepted for publication in *Computer Methods in Applied Mechanics and Engineering*, and can be cited as follows

Z. Csati, N. Moës, T.J. Massart, A stable extended/generalized finite element method with Lagrange multipliers and explicit damage update for distributed cracking in cohesive materials, *Computer Methods in Applied Mechanics and Engineering*, 369, 2020, 113173

A stable extended/generalized finite element method with Lagrange multipliers and explicit damage update for distributed cracking in cohesive materials

Zoltan Csati^{1,2}, Nicolas Moës², and Thierry J. Massart¹

¹Building, Architecture & Town Planning, Université libre de Bruxelles

²GeM Institute, UMR CNRS 6183, École Centrale de Nantes

March 18, 2020

Abstract

A flexible, general and stable mixed formulation is developed to model distributed cracking in cohesive grain-based materials in the framework of the extended/generalized finite element method. The displacement field is discretized on each grain separately, and the continuity of the displacement and traction fields across the interfaces between grains is enforced by Lagrange multipliers. The design of the discrete Lagrange multiplier space is detailed for bilinear quadrangular elements with the potential presence of multiple interfaces/discontinuities within an element. We give numerical evidence that the designed Lagrange multiplier space is stable and provide examples demonstrating the robustness of the method. Relying on the stable discretization, a cohesive zone formulation equipped with a damage constitutive formulation expressed in terms of the traction is used to model propagation of multiple cracks at the interfaces between grains. The damage formulation makes use of an explicit solution procedure, couples the normal and tangential failure modes, accounts for different tension and compression behaviours and takes into account a compression-dependent fracture energy in mixed mode. The framework is applied to complex 2D problems inspired by indirect tension tests of heterogeneous rock-like materials.

Keywords: X-FEM; GFEM; Lagrange multiplier; damage; cohesive zone; explicit solution procedure

1 Introduction

Microstructural modelling of diffuse microcracking in heterogeneous quasi-brittle cohesive grain-based materials is a complex task with many far-reaching applications.

For instance, transfer properties in grain-based rocks are strongly impacted by microcracking, which requires its modelling [28]. The numerical modelling of such phenomena requires three main ingredients: (i) a proper account for the microstructural heterogeneous geometry representation i.e. a description of grain boundaries, (ii) a discretization of the governing equations, allowing a kinematical description of crack propagation, and (iii) the formulation of an efficient crack propagation model. The present work deals with these three aspects, with a focus on the discretization part and on the crack propagation aspect within the X-FEM/GFEM setting.

In certain types of rocks (like granite for example), experiments show that degradation occurs with intergranular cracking. The assumption of dominant intergranular cracking is therefore used in the numerical simulations, with all potential crack paths known a priori (see e.g. [41] for an extensive study on the effect of the cohesive properties). Discontinuities in the displacement field are therefore assumed along all the grain boundaries (interfaces). The continuity of the displacement field must be enforced until the conditions for a crack to open along an interface are met.

The generation of conforming meshes for an assembly of many grains (which can be concave, elongated and of different sizes) may be challenging, even more in 3D configurations. This motivates to use of non-conforming meshes as the extended finite element method (X-FEM) introduced initially for single cracks [31] and later developed for branched and intersecting cracks [15]. A systematic way to handle an arbitrary number of cracks along grain boundaries was then developed in [42] within the generalized finite element method (GFEM). The above contributions add to the classical finite element approximation an enriched approximation through the concept of partition of unity [30]. The enrichment function used may be a Heaviside function [31], a junction function [15] or a grain indicator function [42]. Another way, to inject a discontinuity inside a mesh is to duplicate elements over which the discontinuity is located thus leading to the creation of copies of the initial element [19, 12] and the creation of new duplicated nodes sometimes called virtual [33]. Each duplicated element/node has its own approximation for a given side of the discontinuity. Enrichment based on the partition of unity with piecewise constant functions or the use of copies of elements has been shown to give an equivalent approximation space [5, 43]. This is the way we follow in this work, which will greatly simplify the representation of arbitrary discontinuities.

Using this X-FEM/GFEM setting with non-conforming meshes, the continuity of the displacement along the interfaces can only be enforced weakly. Several approaches exist to prescribe interface conditions. The Lagrange multiplier method (applied without special care to select the Lagrange multiplier space) can result in spurious oscillations in the traction field along the interface. Stabilized methods can be developed, which modify the weak form to achieve stability. The Nitsche method (a variationally consistent version of the penalty method, [35]) introduces a numerical parameter used to tune the stability. An approach relying on a hybrid displacement-stress formalism was proposed in [46], in which the stress unknowns can be statically condensed, and for which stability was illustrated on examples. In the frame of the fictitious node method – a method equivalent to X-FEM [5] – a stabilized Lagrange multiplier method was also proposed in [11]. While the above-mentioned formulations belong to the stabilized methods, there are methods too which are stable by construction. They either enrich the displacement space (bubble stabilization) or

reduce the Lagrange multiplier space (stable Lagrange multiplier method) to reach stability. The latter one is our approach. Their common feature is that they do not require any parameter for stabilization, and are rather constructed to be a priori stable. The bubble stabilization method adds additional (bubble) displacement degrees of freedom, and a stabilization parameter is implicitly introduced from the chosen bubble enrichment [34]. A comparison of some of these methods is provided in [39] where they are tested for specific examples.

A naive application of the Lagrange multiplier method on non-matching meshes results in oscillation as was experienced in [23]. In [32] a discrete Lagrange multiplier field was created by appropriately reducing the number of constraints, thereby providing stability to the method. This approach was later applied to contact problems [18], where the notion of vital edge was introduced to determine which multipliers need to be linked together (see also Section 4). The authors in [4] defined an algorithm that produces a stable Lagrange multiplier space for triangular finite elements with linear interpolation. This algorithm is the starting point in our work. The method in [4] was later extended to triangles with quadratic interpolation [16]. The creation of this reduced Lagrange space is cumbersome, and a reduced number of Lagrange multipliers results in a less rich traction approximation. However, it carries the advantage of not requiring any stabilization parameter. This feature will prove to be useful in our work when considering various microstructures. Furthermore, the interface tractions are determined during the solution procedure, and can be readily exploited in crack propagation criteria. One will see in Section 5 that the update of the crack path at a given step has a very low cost due to the Lagrange multipliers already at disposal.

The main objective of this contribution is to model multiple cracking in cohesive grain-based aggregates in a mesh-independent way. To achieve this, the main originalities presented in this paper are the following.

- A stable discrete Lagrange multiplier space is constructed for the displacement space discretized with bilinear quadrilateral (Q1) elements. The inf-sup conditions are verified computationally to demonstrate that the constructed Lagrange multiplier space is stable. Demonstrating stability for arbitrary interfaces crossing Q1 elements is a generalization of [4].
- An unfitted structured background mesh is sufficient to treat complex heterogeneous aggregates of grains with arbitrary polygonal shapes. Compared to [41], which relies on the GFEM approximation for polycrystals developed in [42], our work extends to Q4 elements and allows tackling the situation when a grain is completely inside an element. We are aware that in this case the function approximation is poor, but at least our method is robust and does not fail even on coarse meshes. We note that realistic microstructures contain small grains compared to the average grains size, therefore this is an important issue. The other additional complexity with respect to [42] is that in our case there is an interplay between the displacement approximation and the Lagrange multiplier approximation. That is, the Lagrange multiplier basis functions has to be constructed in such a way that no linear dependency arises at the junctions. Thirdly, in [42] special care was needed to avoid the linear dependency among

the enrichments, while handling each subdomain independently circumvents this issue, naturally resulting in “floating” subdomains.

- Energy dissipation-based control in the explicit damage framework, required by the use of a combined criterion involving a Mohr-Coulomb criterion with cut-offs, is developed for crack propagation. This is an improvement over [25] in which the authors considered uniform fracture energy and strength linked to the consideration of mode I cracking only, thereby allowing a damage-based control. In our case, the fracture energy is not uniform (increases with the compressive stress) and the strength also depends on the level of compression according to the Mohr-Coulomb failure criterion. Contact is also handled explicitly in this formalism. The combination of these allows taking into account the relevant damaging mechanisms in cohesive grain-based materials. The relative energy dissipation increment we introduce in Section 5.5 has a clear physical meaning.
- The use of the developed framework is illustrated based on an indirect tension test-like configuration, (e.g. Brazilian test) as frequently used on rock specimens in the literature. The authors have not yet met an X-FEM-based mixed method, which was applied for elaborate discontinuity configurations and to realistic microstructural simulations in geomechanics.

To serve these purposes, the paper is organized as follows. In Section 2, the strong and weak forms are defined for the continuous problem. The X-FEM/GFEM discretization of the boundary value problem is next introduced in Section 3. A stable Lagrange multiplier space is defined in Section 4 for quadrilateral elements with bilinear interpolation. A computational verification of stability based on the inf-sup condition is used to confirm that the proposed formulation is indeed stable. The developed Lagrange multiplier approach is complemented in Section 5 to incorporate cohesive cracking. To avoid the use of an initial penalty elastic stiffness of the cohesive zone, an extrinsic cohesive zone method is used. The damage field – through the relative energy dissipation – is used as the controlling variable of an explicit solution procedure and contact in the cohesive zone during unloading is also taken into account. Two illustrations are provided to show the applicability of the method in Section 6 for complex microstructures involving many grains, one of them being focused on a Brazilian-like configuration often used in real rock-experiments. In Section 7 conclusions are drawn and perspectives are shown for further development.

2 Problem formulation

In this section, after introducing some notations, the strong and weak formulations of the boundary value problem (BVP) are given. It is assumed that potential strong discontinuities are present on each interface, and these discontinuities are suppressed by Lagrange multipliers.

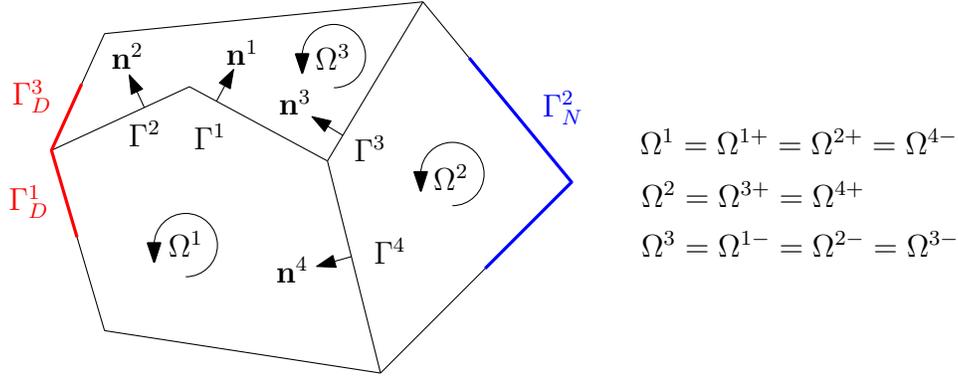


Figure 1: Notations used for the geometrical description of the domain of interest

2.1 Definitions

Let us consider the domain $\Omega \subset \mathbb{R}^2$ partitioned into M non-overlapping *subdomains* Ω^i of polygonal shapes, identified by the index set $I_s = \{1, \dots, M\}$ such that

$$\bar{\Omega} = \bigcup_{i \in I_s} \bar{\Omega}^i, \quad \Omega^i \cap \Omega^j = \emptyset, \quad \forall i, j \in I_s, \quad i \neq j \quad (1)$$

where the overline denotes the closure of a set. Without the loss of generality, it is assumed that $M > 1$. When we refer to the physical nature of the problem, we will use the word *grain* instead of subdomain. A straight segment between two subdomains is called an *interface* and is denoted by Γ^i . Note that two subdomains can have multiple common interfaces. The interfaces are indexed by $I_{\text{int}} = \{1, \dots, N\}$. The neighbouring subdomains to Γ^i are denoted by Ω^{i+} and Ω^{i-} , or $\Omega^{i\pm}$ when dealing with them as a collection. Parts of the boundary of Ω where non-zero Neumann boundary conditions (BC) are prescribed are denoted by Γ_N , whereas Γ_D denotes parts where zero or non-zero Dirichlet BCs are given. It is assumed that $\Gamma_N \cap \Gamma_D = \emptyset$. These boundary parts are replaced by straight segments Γ_N^i ($i \in I_N \subseteq I_s$) and Γ_D^i ($i \in I_D \subseteq I_s$) such that

$$\Gamma_N^i = \Gamma_N \cap \bar{\Omega}^i, \quad \bigcup_{i \in I_N} \Gamma_N^i = \Gamma_N, \quad (2)$$

$$\Gamma_D^i = \Gamma_D \cap \bar{\Omega}^i, \quad \bigcup_{i \in I_D} \Gamma_D^i = \Gamma_D. \quad (3)$$

The subdomain boundaries are oriented counter-clockwise and the interface and boundary segments (Γ^i and Γ_N^i, Γ_D^i) are oriented according to their outward unit normal attached to Ω^{i+} . In other words, the orientation of the subdomain determines the normal and tangent unit vectors of its boundary, and an interface inherits these unit vectors from the first neighbouring subdomain it belongs to. These notations are visualized in Fig. 1 in a general situation. The intuition behind these definitions is the observation that the boundary conditions and the interface conditions can be handled within a unified framework which is advantageous when the constraint equations are weakly enforced, as will be explained in Section 2.3.

2.2 Governing equations

In this work, isotropic linear elastic grains are considered. Note that this assumption is not critical and that grain anisotropy could be considered. The equilibrium, constitutive and kinematic equations for these grains are therefore given as Eq. (4):

$$\begin{aligned}\boldsymbol{\sigma} \cdot \nabla &= \mathbf{0}, & \text{on } \Omega \\ \boldsymbol{\sigma} &= \mathbf{C} : \boldsymbol{\varepsilon}, & \text{on } \Omega \\ \boldsymbol{\varepsilon} &= \nabla^s \mathbf{u} & \text{on } \Omega.\end{aligned}\tag{4}$$

To simplify the presentation, body forces are neglected. The stiffness tensor \mathbf{C} can vary from grain to grain if the material residing in Ω is heterogeneous. The boundary conditions are written as

$$\begin{aligned}\mathbf{u} &= \mathbf{u}_D, & \text{on } \Gamma_D, \\ \boldsymbol{\sigma} \cdot \mathbf{n} &= \mathbf{t}_N, & \text{on } \Gamma_N,\end{aligned}\tag{5}$$

where \mathbf{n} is the outward unit normal to the boundary. Because of the (cohesive) granular structure of the assumed material, it is straightforward to perform the calculation grain-wise as done in [42]. The boundary value problem Eq. (4)–(5) is then written for each grain as:

$$\begin{aligned}\boldsymbol{\sigma}^m \cdot \nabla &= \mathbf{0}, & \text{on } \Omega^m \\ \boldsymbol{\sigma}^m &= \mathbf{C}^m : \boldsymbol{\varepsilon}^m, & \text{on } \Omega^m \\ \boldsymbol{\varepsilon}^m &= \nabla^s \mathbf{u}^m, & \text{on } \Omega^m\end{aligned}\tag{6}$$

$$\mathbf{u}^i = \mathbf{u}_D^i, \quad \text{on } \Gamma_D^i,\tag{7}$$

$$\boldsymbol{\sigma}^j \cdot \mathbf{n}^j = \mathbf{t}_N^j, \quad \text{on } \Gamma_N^j,\tag{8}$$

$\forall m \in I_s, \forall i \in I_D, \forall j \in I_N$, and \mathbf{u}^m is the restriction of \mathbf{u} on Ω^m . In addition, we need to provide the continuity of the primary variable (the displacement) and its derivative along the interfaces. Physically, this means prescribing the continuity of the displacement field

$$\mathbf{u}^{i+}|_{\Gamma^i} = \mathbf{u}^{i-}|_{\Gamma^i}\tag{9}$$

and of the interface traction

$$\mathbf{t}^{j+}|_{\Gamma^j} = \mathbf{t}^{j-}|_{\Gamma^j}\tag{10}$$

$$\boldsymbol{\sigma}^{j+} \cdot \mathbf{n}^j|_{\Gamma^j} = \boldsymbol{\sigma}^{j-} \cdot \mathbf{n}^j|_{\Gamma^j}\tag{11}$$

Introducing the shorthand notation $[[\cdot]]^j = (\cdot)^{j+} - (\cdot)^{j-}$, the above constraints take the form:

$$[[\mathbf{u}]]^i = \mathbf{0}, \quad \text{on } \Gamma^i,\tag{12}$$

$$[[\boldsymbol{\sigma}]]^j \cdot \mathbf{n}^j = \mathbf{0}, \quad \text{on } \Gamma^j.\tag{13}$$

Equations (4)–(5) are equivalent to Equations (6)–(13).

2.3 Mixed continuous formulation

Equations (6)–(13) are cast into the weak form for subsequent discretization. First, we formally derive the equations and then choose the appropriate function spaces based on mechanical reasoning.

The strain energy of the deformable body and the work done by the surface tractions are the sum of the individual contributions on the grains, therefore the energy functional takes the form:

$$\Pi_b = \sum_{m \in I_s} \frac{1}{2} \int_{\Omega^m} \boldsymbol{\sigma}^m(\mathbf{u}^m) : \boldsymbol{\varepsilon}^m(\mathbf{u}^m) \, d\Omega - \sum_{i \in I_N} \int_{\Gamma_N^i} \mathbf{u}^{i+} \cdot \mathbf{t}_N^i \, d\Gamma. \quad (14)$$

Constraint equations are required to enforce the continuity equations (12)–(13). During the discretization, we will see that the finite element nodes do not coincide with the interfaces or the boundaries. Therefore, the continuity equations and the Dirichlet BCs are taken into account weakly. An additional advantage of the weak imposition of Dirichlet boundary conditions is that the system matrix does not need to be modified. The weak solution to (6)–(13) is found by minimizing the total potential energy (14) with respect to the constraints (7), (12). The constrained optimization problem can be turned into a saddle-point formulation by constructing the Lagrangian Π as

$$\Pi = \Pi_b + \Pi_c \quad (15)$$

with

$$\Pi_c = \sum_{i \in I_{\text{int}}} \int_{\Gamma^i} \llbracket \mathbf{u} \rrbracket^i \cdot \boldsymbol{\lambda}^i \, d\Gamma + \sum_{i \in I_D} \int_{\Gamma_D^i} (\mathbf{u}^{i+} - \mathbf{u}_D^i) \cdot \boldsymbol{\lambda}^i \, d\Gamma, \quad (16)$$

and taking its variation with respect to the two fields, i.e. the displacement and Lagrange multiplier fields. In (16), $\boldsymbol{\lambda}^i$ are vector-valued Lagrange multiplier functions. The variational problem is then:

Find $\mathbf{u} \in V$ and $\boldsymbol{\lambda} \in \Lambda$ such that

$$a(\mathbf{u}, \mathbf{v}) + b(\boldsymbol{\lambda}, \mathbf{v}) = f(\mathbf{v}), \quad \forall \mathbf{v} \in V \quad (17a)$$

$$b(\mathbf{u}, \boldsymbol{\mu}) = g(\boldsymbol{\mu}), \quad \forall \boldsymbol{\mu} \in \Lambda \quad (17b)$$

As interfaces and Dirichlet boundaries are handled in the same way, with an abuse of notation, both will be referred to as interfaces. One can unify the interface and Dirichlet boundary conditions by noticing that boundaries are only surrounded by subdomains from one side. Therefore, the displacement jump on an arbitrary internal interface Γ^i or Dirichlet boundary Γ_D is redefined as

$$\llbracket \mathbf{u} \rrbracket^i = \begin{cases} \mathbf{u}^{i+} - \mathbf{u}^{i-} & \text{on } \Gamma^i \\ \mathbf{u}^{i+} & \text{on } \Gamma_D^i \end{cases}$$

The symmetric bilinear and linear forms introduced in (17a)–(17b) are expressed by

$$a(\mathbf{u}, \mathbf{v}) = \sum_{m \in I_s} \int_{\Omega^m} \boldsymbol{\varepsilon}^m(\mathbf{v}^m) : \mathbf{C}^m : \boldsymbol{\varepsilon}^m(\mathbf{u}^m) \, d\Omega \quad (18)$$

$$b(\boldsymbol{\lambda}, \mathbf{v}) = \sum_{i \in I_{\text{int}} \cup I_D} \int_{\Gamma^i \cup \Gamma_D^i} \llbracket \mathbf{v} \rrbracket^i \cdot \boldsymbol{\lambda}^i \, d\Gamma \quad (19)$$

$$f(\mathbf{v}) = \sum_{i \in I_N} \int_{\Gamma_N^i} \mathbf{v}^{i+} \cdot \mathbf{t}_N^i \, d\Gamma \quad (20)$$

$$g(\boldsymbol{\mu}) = \sum_{i \in I_D} \int_{\Gamma_D^i} \boldsymbol{\mu}^i \cdot \mathbf{u}_D^i \, d\Gamma \quad (21)$$

It can be identified from the equilibrium equation (17a) that the Lagrange multiplier represents the traction vector on the interface:

$$\boldsymbol{\sigma}^{i+} \cdot \mathbf{n}^{i+} = -\boldsymbol{\lambda}^i, \quad \boldsymbol{\sigma}^{i-} \cdot \mathbf{n}^{i-} = \boldsymbol{\lambda}^i, \quad (22)$$

where $\mathbf{n}^{i+} = -\mathbf{n}^{i-}$.

In the presence of cracks between the grains, the displacement field is discontinuous. At a junction of interfaces (e.g. when two subdomains have multiple common interfaces), the normal vector to the interfaces changes direction in a discontinuous fashion. Hence, the Lagrange multiplier field representing the tractions across interfaces must be discontinuous as well at such junction points according to (22). Therefore, the weak solution $(\mathbf{u}, \boldsymbol{\lambda}) \in (V, \Lambda)$ to the equations (6)–(13) is sought in the broken Sobolev spaces

$$V = [H^1(\Omega)]^d = \times_{i \in I_s} [H^1(\Omega^i)]^d \quad (23)$$

$$\Lambda = [H^{-1/2}(\Gamma)]^d = \times_{i \in I_{\text{int}} \cup I_D} [H^{-1/2}(\Gamma^i)]^d \quad (24)$$

where \times is the direct product and d is the spatial dimension.

Let us denote by the prime ($'$) the topological dual of a vector space. The existence and uniqueness of the solution to (17) is guaranteed for any $(f, g) \in V' \times \Lambda'$ if the following two conditions are fulfilled [10]:

1. *coercivity of $a(\cdot, \cdot)$ in the kernel of $b(\cdot, \cdot)$* : $\exists \alpha > 0$ such that

$$a(v, v) \geq \alpha \|v\|_V^2, \quad \forall v \in \ker b(v, \boldsymbol{\lambda}) := \{v \in V \mid b(v, \boldsymbol{\lambda}) = 0, \forall \boldsymbol{\lambda} \in \Lambda\} \quad (25)$$

2. *inf-sup condition on $b(\cdot, \cdot)$ (LBB condition)*: $\exists \beta > 0$ such that

$$\inf_{\boldsymbol{\lambda} \in \Lambda \setminus \{0\}} \sup_{v \in V \setminus \{0\}} \frac{b(\boldsymbol{\lambda}, v)}{\|v\|_V \|\boldsymbol{\lambda}\|_\Lambda} \geq \beta \quad (26)$$

Moreover, the a-priori stability estimate

$$\|u\|_V + \|\boldsymbol{\lambda}\|_\Lambda \leq c \|f\|_{V'} + \|g\|_{\Lambda'} \quad (27)$$

holds. The individual fields u and λ are bounded by the data, given as

$$\|u\|_V \leq \frac{1}{\alpha} \|f\|_{V'} + \frac{2\|a\|}{\alpha\beta} \|g\|_{\Lambda'}, \quad (28a)$$

$$\|\lambda\|_\Lambda \leq \frac{2\|a\|}{\alpha\beta} \|f\|_{V'} + \frac{2\|a\|^2}{\alpha\beta^2} \|g\|_{\Lambda'}, \quad (28b)$$

which reflects the importance of the stability constants α and β . If these constants are not bounded away from zero, the errors in the data are significantly magnified and the a-priori estimates (28a)–(28b) can only guarantee a poor approximation.

3 Discretization

The equations of the bulk (Eq. (6)) are formulated on subdomains. The discrete displacement and Lagrange multiplier spaces are created as broken polynomial spaces, i.e. the spaces of piecewise continuous functions of low-degree polynomials, defined on certain submeshes. To create approximation spaces in the bulk and on the surfaces, the *continuous* finite element basis functions defined on a background mesh are restricted to the bulk and to the interfaces to obtain approximations for the displacement and Lagrange multiplier fields, respectively. In what follows, this intuition is made precise.

3.1 Approximations

The piecewise continuous space V_h on Ω can be constructed by patching the subspaces together (\bigoplus denoting the direct sum):

$$V_h = \bigoplus_{i \in I_s} V_h^i \quad (29)$$

and hence it holds that $\mathbf{u}_h|_{\Omega^i} = \mathbf{u}_h^i$. A similar decomposition was used in [42] for polycrystals.

Remark 1 *The displacement approximation can be constructed subdomain-wise (see Eq. (29)). Therefore, it naturally results in a scheme prone to domain decomposition, and hence to parallelism.*

The basis for V_h^i is constructed from a background mesh \mathcal{M} (the notations can be followed in Fig. 2a). The background mesh \mathcal{M} is chosen to be large enough such that it contains $\cup_{i \in I_s} \overline{\Omega^i}$. For efficiency, \mathcal{M} is chosen as a structured quadrilateral mesh. To approximate the displacement field on subdomain i , we introduce the approximation space constructed as

$$V_h^i := \text{span} \left\{ \hat{\psi}_j^i \mid \text{supp} \left(\hat{\psi}_j^i \right) \cap \Omega^i \neq \emptyset, j \in \mathcal{M} \right\}, \quad (30)$$

where $\hat{\psi}_j^i$ are the *vectorial* nodal shape functions cut off on the boundaries of Ω^i :

$$\hat{\psi}_j^i := \begin{cases} \psi_j^i & \text{on } \Omega^i \\ 0 & \text{on } \Omega \setminus \Omega^i \end{cases}. \quad (31)$$

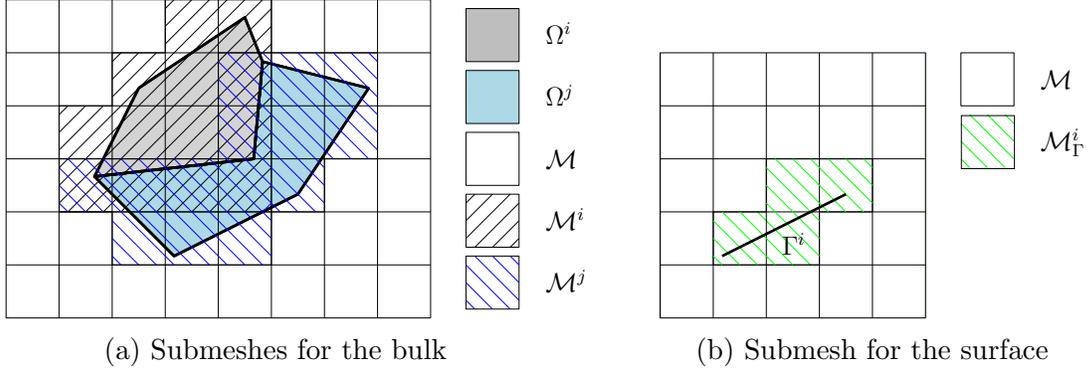


Figure 2: Notations for the local meshes

The approximation function is the same for each component of the displacement and Lagrange multiplier fields. Vectorial shape functions are only used to ease the presentation, in the implementation, scalar basis functions were applied. Since the support of the nodal finite element shape functions we use stretch to the neighbouring elements, we define the submesh $\mathcal{M}^i \subseteq \mathcal{M}$, consisting of those elements e which are at least partially cut by Ω^i :

$$\mathcal{M}^i := \{e \in \mathcal{M} \mid e \cap \Omega^i \neq \emptyset\}. \quad (32)$$

Therefore, a basis for V_h^i can be constructed from the nodal basis functions corresponding to the mesh \mathcal{M}^i . It is then sufficient to consider

$$V_h^i := \text{span}\{\hat{\psi}_j^i \mid j \in \mathcal{M}^i\}. \quad (33)$$

The cut shape functions $\hat{\psi}_j^i$ are replaced by the standard shape functions ψ_j^i and the integrals are evaluated on Ω^i . Note that without constraint equations, \mathbf{u}_h is discontinuous between any two subdomains $\Omega^i \neq \Omega^j$.

Similarly to (32), the submesh containing the interface Γ^i is defined as (see Fig. 2b)

$$\mathcal{M}_\Gamma^i := \{e \in \mathcal{M} \mid e \cap \Gamma^i \neq \emptyset\}. \quad (34)$$

The discrete Lagrange multiplier space, following [4], is assumed to be

$$\Lambda_h^i = \text{span}\left\{\tilde{\psi}_j^i \mid j \in \tilde{\mathcal{M}}_\Gamma^i\right\}, \quad (35)$$

where $\tilde{\mathcal{M}}_\Gamma^i$ is the same mesh as \mathcal{M}_Γ^i but not every node of it defines an independent basis function. The Lagrange multiplier basis functions are created as the weighted trace of the nodal basis functions on the interface:

$$\tilde{\psi}_j^i = \sum_{k \in \mathcal{M}_\Gamma^i} \alpha_{jk} \psi_k|_{\Gamma^i}. \quad (36)$$

The global approximation space is therefore

$$\Lambda_h = \bigoplus_{i \in I_{\text{int}} \cup I_D} \Lambda_h^i. \quad (37)$$

The next task is then to find $\tilde{\mathcal{M}}_\Gamma^i$ and α_{jk} for all interfaces such that an a priori stable formulation is obtained. An algorithm to accomplish this goal is detailed in Section 4. The construction of the discrete spaces V_h in (29) and Λ_h in (37) allows seeking the approximation on each subdomain and on each interface independently.

3.2 Discrete weak formulation

The saddle point problem (17a)–(17b) is now discretized. Using the spaces (33) and (35), the approximations are given by

$$\begin{aligned} \mathbf{u}_h^i &= \sum_{k \in \mathcal{M}^i} \boldsymbol{\psi}_k^i U_k^i, & \mathbf{v}_h^i &= \sum_{j \in \mathcal{M}^i} \boldsymbol{\psi}_j^i V_j^i, & i &= 1, \dots, M \\ \boldsymbol{\lambda}_h^i &= \sum_{k \in \tilde{\mathcal{M}}_\Gamma^i} \tilde{\boldsymbol{\psi}}_k^i L_k^i, & \boldsymbol{\mu}_h^i &= \sum_{j \in \tilde{\mathcal{M}}_\Gamma^i} \tilde{\boldsymbol{\psi}}_j^i M_j^i, & i &= 1, \dots, N \end{aligned} \quad (38)$$

where U_k^i , V_k^i , L_k^i , M_k^i are *vectorial* degrees of freedom. By substituting (38) in (17a)–(17b), the discrete saddle-point system reads:

$$\begin{cases} \text{Find } (\mathbf{u}_h, \boldsymbol{\lambda}_h) \in V_h \times \Lambda_h \text{ such that} \\ a(\mathbf{u}_h, \mathbf{v}_h) + b(\boldsymbol{\lambda}_h, \mathbf{v}_h) = f(\mathbf{v}_h), & \forall \mathbf{v} \in V_h \\ b(\mathbf{u}_h, \boldsymbol{\mu}_h) = g(\boldsymbol{\mu}_h), & \forall \boldsymbol{\mu}_h \in \Lambda_h \end{cases} \quad (39)$$

with the discretized symmetric bilinear and linear forms expressed as

$$a(\mathbf{u}_h, \mathbf{v}_h) = \sum_{i \in I_s} \sum_{j, k \in \mathcal{M}^i} V_j^i K_{jk}^i U_k^i, \quad (40)$$

$$b(\mathbf{u}_h, \boldsymbol{\mu}_h) = \sum_{i \in I_{\text{int}}} \sum_{j \in \tilde{\mathcal{M}}_\Gamma^i} \sum_{k \in \mathcal{M}^{i\pm}} M_j^i B_{jk}^i (U_k^{i+} - U_k^{i-}) + \sum_{i \in I_D} \sum_{j \in \tilde{\mathcal{M}}_\Gamma^i} \sum_{k \in \mathcal{M}^{i+}} M_j^i B_{jk}^i U_k^{i+}, \quad (41)$$

$$f(\mathbf{v}_h) = \sum_{i \in I_N} \sum_{j \in \mathcal{M}^{i+}} V_j^{i+} F_j^i, \quad (42)$$

$$g(\boldsymbol{\mu}_h) = \sum_{i \in I_D} \sum_{j \in \tilde{\mathcal{M}}_\Gamma^i} M_j^i G_j^i, \quad (43)$$

in which $\mathcal{M}^{i\pm}$ are the two submeshes corresponding to the neighbouring subdomains of interface Γ^i .

Remark 2 Note that $V_h \not\subset V$ and $\Lambda_h \not\subset \Lambda$ as the geometry is taken into account during the quadrature. As a result, our discretization is non-conforming, making the stability more difficult to achieve and prove [9].

The stiffness matrix on Ω^i , with $\boldsymbol{\varepsilon}^i(\boldsymbol{\psi}_j^i) = \nabla^s \boldsymbol{\psi}_j^i$, is given by

$$K_{jk}^i = \int_{\Omega^i} \boldsymbol{\varepsilon}^i(\boldsymbol{\psi}_j^i) : \mathbf{C}(E^i, \nu^i) : \boldsymbol{\varepsilon}^i(\boldsymbol{\psi}_k^i) \, d\Omega, \quad (44)$$

the coupling matrix is written as

$$B_{jk}^i = \int_{\Gamma^i \cup \Gamma_D^i} \tilde{\boldsymbol{\psi}}_j^i \cdot \boldsymbol{\psi}_k^{i+} \, d\Gamma \quad (45)$$

and the nodal forces vectors are

$$F_j^i = \int_{\Gamma_N^i} \mathbf{t}_N^i \cdot \boldsymbol{\psi}_j^i \, d\Gamma \quad (46)$$

$$G_j^i = \int_{\Gamma_D^i} \mathbf{u}_D^i \cdot \tilde{\boldsymbol{\psi}}_j^i \, d\Gamma. \quad (47)$$

Note, that we use $\boldsymbol{\psi}_k^{i+}$ in (45) as it is the same as $\boldsymbol{\psi}_k^{i-}$. The assembled matrices have the saddle-point structure

$$\begin{bmatrix} \mathbf{K} & \mathbf{B}^\top \\ \mathbf{B} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{g} \end{bmatrix}, \quad (48)$$

in which \mathbf{u} contains the unknowns U_k^i and $\boldsymbol{\lambda}$ contains the unknowns L_k^i . The algebraic saddle point system (48) is uniquely solvable if the following conditions hold [6]

$$\ker \mathbf{B}^\top = \mathbf{0}, \quad (49)$$

$$\ker \mathbf{K} \cap \ker \mathbf{B} = \mathbf{0} \quad (50)$$

Equation (49) expresses that the Lagrange multipliers are linearly independent, while (50) is the discrete version of the ellipticity-on-the-kernel condition. We emphasize that the second equation is a relaxed version of \mathbf{K} being positive definite, which would not hold in our case. In fact, the null space of \mathbf{K} is spanned by the rigid body modes of the individual grains. Any meaningfully fine mesh contains sufficient number of elements so that the resulting Lagrange multiplier space is rich enough to keep the grains together, preventing some grains floating.

Remark 3 *No special care is taken for the elements partially or completely cut because the element cutting is performed subdomain-wise. It means that those elements that are completely outside of Ω are automatically discarded. Degrees of freedom are associated only to the nodes of the elements that are completely or partially cut by subdomains. Therefore, uncut elements do not increase the problem size.*

4 Stable mixed formulation

Although a unique solution exists if the equations (49)–(50) hold, a robust numerical method must be stable too. In the first part of this section an algorithm is developed to create the Lagrange multiplier space. The second part discusses the stability of the resulting mixed method.

4.1 Construction of the stable Lagrange multiplier space

Some additional notions used in the creation of a stable Lagrange multiplier space are now defined. A *cut edge* is an element edge cut by an interface. A *cut element* is an element the interior of which is fully or partially cut by an interface. An *intersection point* is the intersection of an interface with an edge. An *isolated node* is a node of a cut element that does not lie on a cut edge. A *junction* is the intersection of

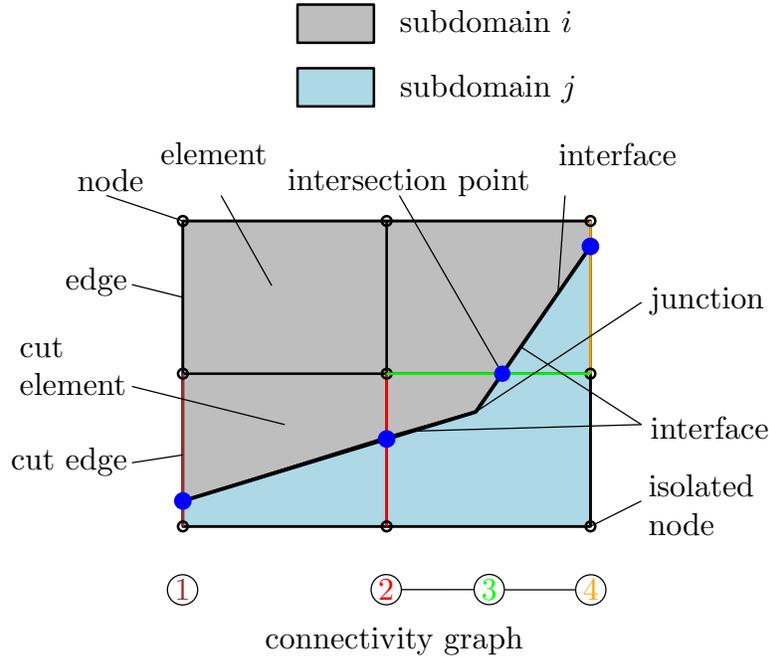


Figure 3: Notions used for the stable Lagrange multiplier space

interfaces. The *connectivity graph* is an undirected graph, the vertices of which are the cut edges, and in which two of such vertices are connected if the corresponding cut edges meet at a common node.

These concepts are represented in Fig. 3. The bottom of Fig. 3 shows the connectivity graph of the intersection points. The vertices of this graph are circled. We can see that vertex ① of the graph matches a cut edge which does not connect to any other cut edges. Therefore, vertex ① is isolated in the graph (not to be confused with the isolated nodes). This graph is used in this section to construct a stable interpolation for the Lagrange multiplier field.

In order to avoid interface meshing, which can be complicated in 3D, the Lagrange multiplier unknowns are defined at the nodes of the background finite element mesh as in [4] (see also [36]). The corresponding question that needs to be addressed consists in determining how to distribute the Lagrange multiplier degrees of freedom to the nodes of the mesh $\tilde{\mathcal{M}}_\Gamma^i$. If each node of $\tilde{\mathcal{M}}_\Gamma^i$ held an independent Lagrange multiplier, the dimension of Λ_h^i would be too high. Therefore, a reduction is performed among them by associating a certain Lagrange multiplier to a *group of nodes*. The identification of these groups is done not element-by-element but at the global level, i.e. regarding all nodes of $\tilde{\mathcal{M}}_\Gamma^i$. The groups are determined according to topological information, i.e. the location of the nodes relative to the interface Γ^i . Once these groups are formed, the interpolation within an element is built by taking the restriction of the bulk element basis functions on Γ^i . An example is provided in Fig. 4, where the Lagrange multiplier on element e is interpolated as $\lambda^e = \psi_2|_{\Gamma^i} L_1^i + (\psi_1 + \psi_3 + \psi_4)|_{\Gamma^i} L_2^i$. Triangle elements with linear interpolation were used in [4] with a single interface. As a consequence, the trace of these linear shape functions resulted in a linear interpolation for the corresponding Lagrange multiplier along the interface cutting the element. The procedure was next extended

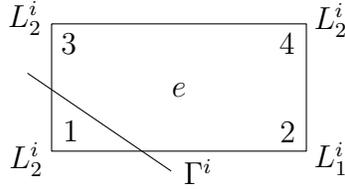


Figure 4: Example for the interpolation of the Lagrange multiplier field within an element

to quadratic triangles in [16]. In [24] the idea of [4] was used with an additional crack tip enrichment. They concluded that both the standard tip enrichments and the vectorial tip enrichments applied with the Heaviside enrichment pass the inf-sup test, provided a constant interpolation is used for the Lagrange multiplier field on the element containing the crack tip.

The present work generalizes [4] by considering the following further improvements: (i) general multiple interface configurations, furthermore (ii) incorporated within bilinear quadrilateral elements forming a Cartesian mesh. Before outlining the algorithm for creating the shape functions for the Lagrange multipliers, let us look at the necessary requirements to be fulfilled by the Lagrange multiplier space. The rationale behind it is that the best approximation property is achieved upon polynomial completeness. We therefore seek a discretization that can at least represent a uniform stress field Σ on Ω . This can be achieved if the material is homogeneous (all grains have the same elastic properties, E and ν) and linear elastic. Setting the Poisson ratio to 0, if a rectangular specimen is clamped on its left edge and pulled horizontally from the right, the displacement field in Cartesian coordinates is

$$\mathbf{u} = (\alpha x + \beta)\mathbf{e}_x. \quad (51)$$

Since this analytical expression for \mathbf{u} is continuous, $[[\mathbf{u}]]^i = \mathbf{0}$, and (17b) is automatically fulfilled. After substitution of (51) into (39) and integration by parts, we can observe that

- As Σ is uniform, the interface traction and therefore the Lagrange multiplier shape functions must form a partition of unity. To be able to represent a uniform stress field with the interpolation, due to the property that $\sum_{k=1}^4 \psi_k = 1$ on an element, all nodes of $\tilde{\mathcal{M}}_\Gamma^i$ must contain a Lagrange multiplier.
- The integrand of

$$\int_{\Gamma_N} \mathbf{v} \cdot \mathbf{t}_N \, d\Gamma \quad (52)$$

is discontinuous even if the traction is uniform on Γ_N because the test functions $\mathbf{v}^m \in [H^1(\Omega^m)]^d$ are discontinuous across the interfaces. Therefore, if we use the standard Gauss-Legendre quadrature and want to integrate exactly, we must integrate on Γ_N segment-by-segment as (42) and (46) suggest.

- As noted at the end of Section 2, the Lagrange multiplier field is discontinuous at junctions.

- Analogously to mortar methods, in which the junction points carry no Lagrange multipliers [38], we use a constant interpolation for the Lagrange multiplier on an element containing a junction.
- In order to avoid over-determination, Dirichlet boundaries are handled in the same way as internal interfaces.

First, a method for building a discrete Lagrange multiplier space is given in case of a single interface in Ω .

To avoid defining a too rich Lagrange multiplier space that causes the instability, some of the multipliers are linked together and are prescribed to have the same values. The nodes of the cut elements on the two sides of an interface are connected. A straightforward approach that consists of using all the cut edges to link the nodes on the two sides of an interface fails to satisfy the LBB condition in the general case. This wrong, naive strategy is illustrated in Fig. 5. In such a strategy, the number of tyings/constraints is too high with respect to what is strictly needed to enforce the continuity of the fields. Therefore, a reduction of the Lagrange multiplier space has to be performed similarly to [4] but tailored to quadrilateral elements. This is done by first selecting some cut edges. The selection of the set of these cut edges to reach a stable approximation is not unique hence we have some freedom in choosing them. The following four rules are applied in this selection procedure based on the connectivity graph illustrated in Fig. 3.

1. A cut edge corresponding to an isolated vertex of the connectivity graph is always selected.
2. A selected cut edge is not allowed to be connected to any other selected cut edge in the connectivity graph.
3. A non-selected cut edge must be connected to at least one selected cut edge in the connectivity graph.
4. A cut edge is never selected if the interface cutting it has a junction in an element containing the cut edge.

The third rule makes the discrete Lagrange multiplier space as large as possible but due to the second rule not too large to avoid dependencies. The fourth rule is responsible for not introducing independent Lagrange multipliers at a junction. Finally, the first rule guarantees that all nodes of the cut elements contain a Lagrange multiplier, thereby satisfying the partition of unity requirement.

Setting the discrete Lagrange multiplier space is performed according to Alg. 1 (follow also in Fig. 6). The links (ellipses in Fig. 5) can also be represented as a graph, shown in Fig. 6.

With these chosen links, the Lagrange multiplier shape functions inside the elements are built as the trace of the nodal (bulk) displacement shape functions on the interface. They are

- constant if there is a change in the normal vector in the underlying element
- linear if the interface passes through opposite edges

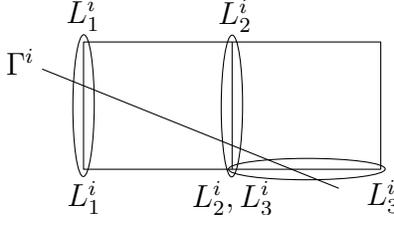


Figure 5: Naive strategy to equate Lagrange multipliers. The node links are denoted by an elliptical shape.

Algorithm 1 Linking Lagrange multipliers

- 1: Nodes of an element which contains a junction are linked together to fulfil the partition of unity
 - 2: Determine the set of selected edges from the cut edges not part of the already processed edges
 - 3: Link the two nodes of a selected edge
 - 4: The remaining node of a cut edge is linked the other node of the cut edge
 - 5: An isolated node is linked to the opposite node of the cut element it belongs to
-

- quadratic without the linear term if the interface passes through neighbouring edges (cf. the third plot in Fig. 6)

Remark 4 *The Lagrange multiplier shape functions constructed in this way are not local to the considered element, their support can extend to the neighbouring elements.*

Note that the use of the intersection points on the selected cut edges would not constitute a robust solution to define a basis for Λ_h^i . This is due to the fact that

- the shape functions need to be created globally and not at the element level since their support extends to neighbouring elements,
- for non-straight interfaces this process would be even more difficult
- the extension would become very complicated for 3D configurations in which surface interfaces have to be considered, requiring global 2D shape functions to be defined.

Remark 5 *The algorithm proposed in [20] forms piecewise continuous Lagrange multiplier shape functions, mentioning that the computation of the weights can be done element-wise. In our case, the weights are also calculated element-wise and the created shape functions are globally C^0 continuous except inside those elements where the normal vector to the interfaces changes in a discontinuous fashion. Moreover, they are piecewise C^1 continuous (see the third plot of Fig. 6).*

For multiple interfaces, the procedure described above is performed on each of them, independently from each other. This is illustrated in Fig. 7 where the Lagrange multiplier field is constant on the element containing the junction.

The proposed general algorithm Alg. 1 for the selection of Lagrange multipliers is now represented for different cases in Fig. 6, where the cut edges selected to define

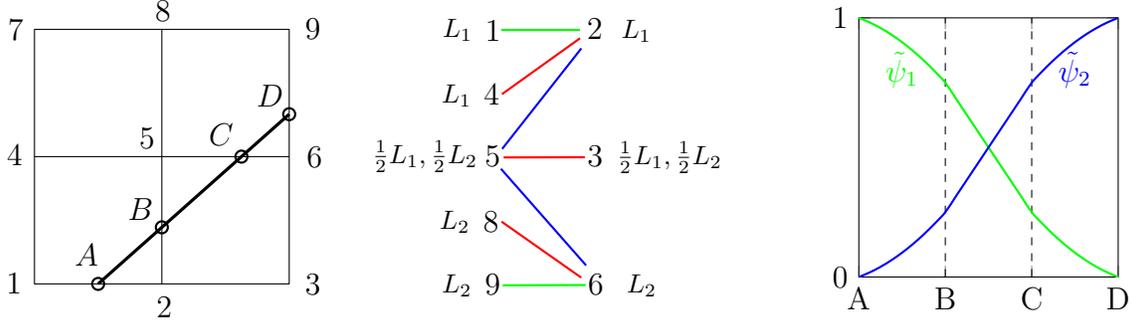


Figure 6: Constructing the Lagrange multiplier basis functions for a single interface

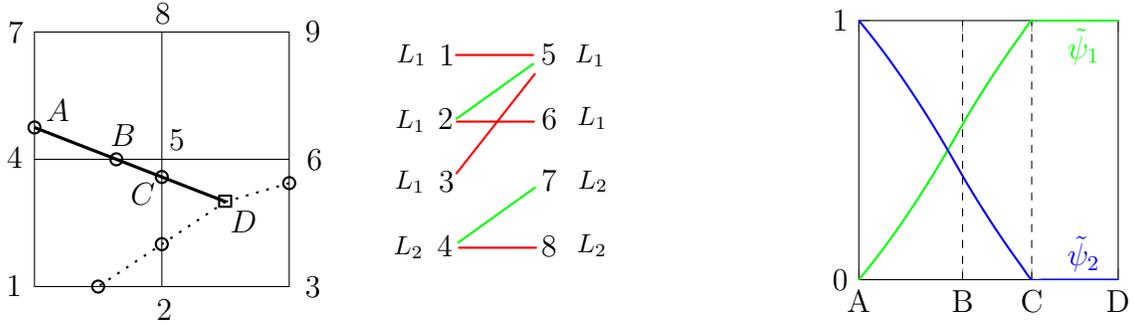


Figure 7: Constructing the Lagrange multiplier basis functions in presence of a junction

them are drawn in green, the other non-selected cut edges are depicted in blue, and the isolated nodes are represented in red. Figure 6 depicts this general case in which the shape function ψ_5 is used to interpolate both L_1 and L_2 . If we chose ψ_5 to interpolate only λ_1 for example, then $\tilde{\psi}_2$ would be constant on the element with nodes 1-2-5-4. In Fig. 7 the selection procedure is shown in presence of a triple junction. As mentioned before, the building of the discrete Lagrange multiplier space is done interface-wise, therefore the method is represented on one of the three interfaces (cf. first figure of Fig. 7). It is shown in the third figure of Fig. 7 that the Lagrange multiplier field is constant on the element containing the junction.

A remaining situation to tackle is when a subdomain is fully embedded inside an element. There is then no intersection point on some of the interfaces. As all the sides of this embedded subdomain (i.e. its interfaces) are inside an element, a constant Lagrange multiplier field is used on all these sides by linking all four nodes of the element together.

In conclusion, the process of mesh tying and stiffness assembly (in the linear elastic case) is the following.

1. Cut the original mesh \mathcal{M} into M pieces including the phantom displacement degrees of freedoms, where M is the number of subdomains.
2. Compute the stiffness matrices on each subdomain (possibly in parallel).
3. Select some of the cut edges to define independent Lagrange multipliers.
4. Create the Lagrange multiplier space according to Alg. 1.

5. Calculate the coupling integrals in Eq. (45).

4.2 Stability of the discretization scheme

We now assess the stability of the designed mixed formulation. Two kinds of tests are performed in this part: the patch test and the inf-sup test.

The patch test is used to evaluate the polynomial completeness of the finite element. For the elasticity equation of compressible solid, the Q1 quadrilateral elements satisfy the patch test, i.e. the linear displacement solution can be reproduced exactly (assuming no under-integration). We investigate here whether the Lagrange multipliers can ensure perfect tying across the interfaces, i.e. zero displacement jump can be achieved. This procedure is applied considering an assembly of six subdomains, as seen in Fig. 8. We note that this non-realistic grain configuration was manufactured to demonstrate the robustness of our method in the presence of concave and tiny subdomains. Zero displacement is prescribed on the left side of the square and uniform normal traction is applied on the right side. The elastic properties of the grains were collectively set to $E = 1$ and $\nu = 0$. As there is no contraction, the vertical displacement component must be zero, and we expect a linear variation of the horizontal displacement along x . It can be observed that the included discontinuities are successfully deactivated by the Lagrange multipliers and that the numerical result matches with the exact solution (51). Hence, the patch test on a Cartesian mesh is passed by the formulation. With coarse meshes, the tiny subdomain resides

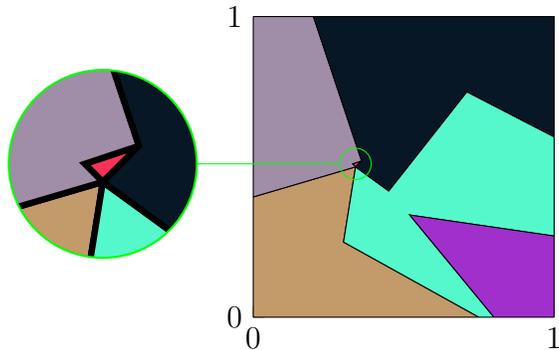


Figure 8: Geometry for the simple tension test

in a single element (Fig. 9a), while on fine meshes, some elements are completely uncut (Fig. 9b). Our discretization scheme was compared with the standard FEM using displacement-based formulation on a conforming mesh. The problem above was considered but with $\nu = 0.3$. The comparison was based on two quantities: the value of the displacement field at the top right corner of the domain, and the strain energy. The two methods showed good agreement in terms of these quantities.

The computation of the domain integrals on polygonal element parts is handled by triangulation. We note that the subdivision of an element into triangles is done only for the sake of computing the integrals in the stiffness matrices (44), no additional degrees of freedom are introduced. The normal and tangential tractions are directly calculated from the Lagrange multipliers knowing the interface normals. The traction is also computed on the Dirichlet boundary Γ_D .

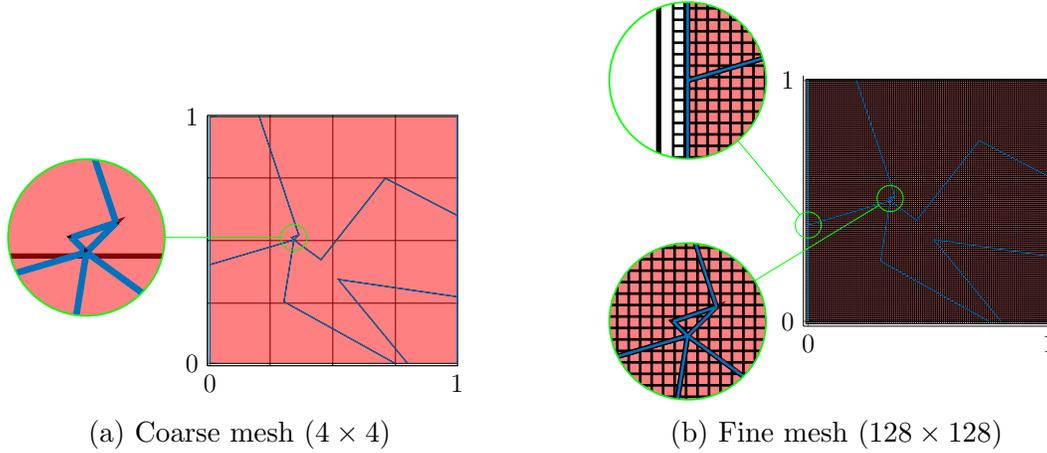


Figure 9: Two example finite element meshes covering the subdomains in Fig. 8

We now focus on the inf-sup conditions. In case of mixed problems of the form (17), the finite dimensional versions of the inequalities (25)–(26) are

$$0 < \alpha^* \leq \inf_{0 \neq u_h \in Z_h} \sup_{0 \neq v_h \in Z_h} \frac{a(u_h, v_h)}{\|u_h\|_V \|v_h\|_V}, \quad (53)$$

$$0 < \beta^* \leq \inf_{0 \neq \lambda_h \in \Lambda_h} \sup_{0 \neq v_h \in V_h} \frac{b(\lambda_h, v_h)}{\|v_h\|_V \|\lambda_h\|_\Lambda} \quad (54)$$

with α^* and β^* being independent of the mesh size h [10]. The second condition (54) is often called *the* inf-sup condition. We will refer to it as LBB¹ condition (which is another commonly used name) to differentiate it from the first one that we will refer to as coercivity condition.

Remark 6 *The continuity of $a(\cdot, \cdot)$ and $b(\cdot, \cdot)$ need not be checked, it automatically comes from the fact that we used the proper norms (see [9]).*

Checking the fulfilment of the pair of conditions (53)–(54) is very difficult for most BVPs. Even if the stability was proved for the continuous problem, it would not imply the stability of the discrete problem due to the non-conforming discretization. The inf-sup test provides practical means to determine the LBB constant. The original version of the test was created for constraints prescribed on domains [14]. For interfacial constraints, the test was slightly modified in [2]. Its idea is to express the LBB constant with the help of an eigenvalue problem. The square root of the smallest non-zero eigenvalue of

$$\begin{cases} \text{Find } \beta \in \mathbb{R} \text{ and } 0 \neq (u_h, \lambda_h) \in V_h \times \Lambda_h \text{ such that } \forall (v_h, \mu_h) \in V_h \times \Lambda_h \\ (u_h, v_h)_{V;\Omega} + b(\lambda_h, v_h) + b(u_h, \mu_h) = -\beta(\lambda_h, \mu_h)_{\Lambda;\Gamma}. \end{cases} \quad (55)$$

is identified as the LBB constant: $\beta^* = \sqrt{\beta_{\min}}$. In our problem, $V^i = [H^1(\Omega^i)]^2$, $\forall i \in I_s$ and $\Lambda^i = [H^{-1/2}(\Gamma^i)]^2$, $\forall i \in I_{\text{int}} \cup I_D$. As it is difficult to deal with fractional

¹Ladyzhenskaya-Babuška-Brezzi

Sobolev spaces, they are replaced by mesh-dependent L^2 -norms as in [4]:

$$\|\mu_h\|_{-1/2;\Gamma}^2 := \sum_{e \in \mathcal{E}} h_e \|\mu_h\|_{0,e}^2, \quad (56)$$

$$\|v_h\|_{1/2;\Gamma}^2 := \sum_{e \in \mathcal{E}} \frac{1}{h_e} \|v_h\|_{0,e}^2, \quad (57)$$

where \mathcal{E} is the one-dimensional mesh on Γ and h_e is the size of element e in \mathcal{E} . We note that in our Lagrange multiplier selection algorithm, no such situation arises that h_e is arbitrarily close to zero, whatever the position of the interface and the mesh used. This property and the norm equivalence allows replacing the one-dimensional mesh segments h_e with the background mesh size h . We also use norm equivalence between the H^1 -norm $\|\cdot\|_{1;\Omega}$ and the energy norm $\|\cdot\|_{E;\Omega} := \sqrt{a(\cdot, \cdot)}$ to replace $(u_h, v_h)_{1;\Omega}$ with $a(u_h, v_h)$, therefore being able to use the already available stiffness matrix. Using these norms, the matrix representation of the eigenvalue problem (55) reads:

$$\begin{bmatrix} \mathbf{K} & \mathbf{B}^\top \\ \mathbf{B} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \boldsymbol{\lambda} \end{bmatrix} = \beta \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & h\mathbf{G} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \boldsymbol{\lambda} \end{bmatrix}, \quad (58)$$

where the stiffness matrix \mathbf{K} and the coupling matrix \mathbf{B} are already defined in (44) and in (45). The Gram matrix $\mathbf{G} = \text{diag } G_{k\ell}^i$ is block diagonal, and the block matrix corresponding to interface i is formed as $G_{k\ell}^i = (\tilde{\boldsymbol{\psi}}_k^i, \tilde{\boldsymbol{\psi}}_\ell^i)_{\Gamma^i}$. The assembly of $G_{k\ell}^i$ is similar to that of $B_{k\ell}^i$. It is a common practice to solve (58) either for \mathbf{u} or for $\boldsymbol{\lambda}$. As \mathbf{K} is singular, we chose to eliminate $\boldsymbol{\lambda}$ from (58) to obtain

$$\frac{1}{h} (\mathbf{B}^\top \mathbf{G}^{-1} \mathbf{B}) \mathbf{u} = \beta \mathbf{K} \mathbf{u}. \quad (59)$$

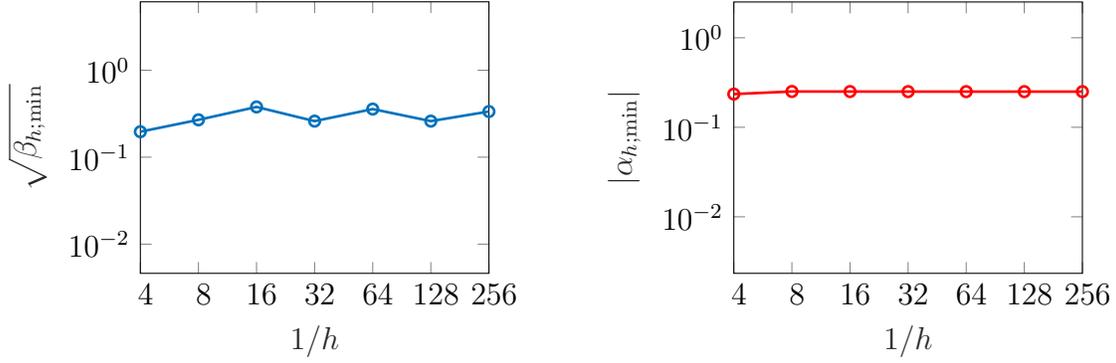
A series of generalized linear eigenvalue problems (59) are solved on increasingly refined meshes. On this basis, the inf-sup constant is approximated as

$$\beta^* = \min_{h_i} \sqrt{\beta_{h;\min}}. \quad (60)$$

For the inf-sup test, the same BVP was considered as for the patch test. Computing $\beta_{h;\min}$ on many meshes increases the reliability of the inf-sup test. Version 2.8 of the Rational Krylov Toolbox [7] was used to compute the smallest eigenvalues. The LBB constants on meshes with $1/h = 4, 8, 16, 32, 64, 128, 256$ are shown in Fig. 10a, giving the hint that the choice of the pair (V_h, Λ_h) is stable.

Most authors only check the LBB condition. It is not a problem if the coercivity condition (25) is proved to hold on V and if a conforming discretization is used. However, our discretization is non-conforming in both the displacement and the Lagrange multiplier fields. Hence, it is necessary to investigate the behaviour of the coercivity constant α^* in (53) as well. Similarly to the LBB condition, this inf-sup condition can also be verified with an associated eigenvalue problem. Let us first introduce (cf. (25)) the discrete version of the kernel of $b(\cdot, \cdot)$, i.e.

$$Z_h = \ker b(v_h, \mu_h) := \{v_h \in V_h \mid b(v_h, \mu_h) = 0, \forall \mu_h \in \Lambda_h\}. \quad (61)$$



(a) LBB constant for different meshes (b) Coercivity constant for different meshes

Figure 10: Stability test

The smallest in modulus eigenvalue of the generalized eigenvalue problem

$$\begin{cases} \text{Find } \alpha \in \mathbb{R} \text{ and } 0 \neq u_h \in Z_h \text{ such that } \forall v_h \in Z_h \\ a(u_h, v_h) = \alpha(u_h, v_h)_{V;\Omega} \end{cases} \quad (62)$$

gives the Brezzi coercivity constant: $\alpha^* = |\alpha_{\min}|$. In most problems, Z_h is not known explicitly. An alternative eigenvalue problem is

$$\begin{cases} \text{Find } \alpha \in \mathbb{R} \text{ and } 0 \neq (u_h, \lambda_h) \in V_h \times \Lambda_h \text{ such that } \forall (v_h, \mu_h) \in V_h \times \Lambda_h \\ a(u_h, v_h) + b(\lambda_h, v_h) + b(u_h, \mu_h) = \alpha(u_h, v_h)_{V;\Omega}. \end{cases} \quad (63)$$

It was proved in [1] that the smallest in modulus eigenvalue of (63) and the smallest in modulus eigenvalue of (62) agree. This time we cannot replace $(u_h, v_h)_{V;\Omega}$ with $a(u_h, v_h)$ as we did before because all the eigenvalues would be infinity or 1. Therefore, the new term is directly evaluated as follows. By the definition of the broken Sobolev spaces and given that in our problem $V^i = [H^1(\Omega^i)]^2$ for $i \in I_s$, we have

$$\begin{aligned} (u_h, u_h)_{1;\Omega} &= \|u_h\|_{1;\Omega}^2 = \sum_{i \in I_s} \|u_h\|_{1;\Omega^i}^2 = \\ &= \sum_{i \in I_s} (\|u_h\|_{0;\Omega^i}^2 + |u_h|_{1;\Omega^i}^2) = \mathbf{u}^\top \mathbf{S} \mathbf{u}. \end{aligned} \quad (64)$$

The subdomain matrices are formed as

$$S_{k\ell}^i = (\psi_k^i, \psi_\ell^i)_{\Omega^i} + (\nabla \psi_k^i, \nabla \psi_\ell^i)_{\Omega^i}, \quad i \in I_s, \quad (65)$$

where the global matrix is $\mathbf{S} = \text{diag } S_{k\ell}^i$ and \mathbf{u} is the global vector of the displacement unknowns. The assembly of the subdomain matrices $H_{k\ell}^i$ and the global matrix \mathbf{S} is done exactly as for the stiffness matrices. However, more Gauss points were used because \mathbf{S} contains the shape functions as well, and not only their derivatives as for \mathbf{K} . The eigenvalue problem (63) with these matrices is

$$\begin{bmatrix} \mathbf{K} & \mathbf{B}^\top \\ \mathbf{B} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \boldsymbol{\lambda} \end{bmatrix} = \alpha \begin{bmatrix} \mathbf{S} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \boldsymbol{\lambda} \end{bmatrix} \quad (66)$$

The calculation of $|\alpha_{\min}|$ is performed with the same method as used for Eq. (59). Figure 10b shows that for the investigated meshes the Brezzi coercivity constant is also bounded away from zero.

Although a numerical evidence is not equivalent to a proof, the patch test and the careful examination of the inf-sup constants give confidence about the stability of the discretization scheme.

5 Cracking

In quasi-brittle materials the nonlinear zone around the crack tip is not negligible. The cohesive model allows convenient implementation, smoother crack opening and a better estimation of the released energy from a traction-separation relation. This enables tracing the softening response in a quasi-brittle failure context. To avoid the costly incremental-iterative procedure most often used in cohesive approaches, an explicit solution procedure based on a dissipation-driven scheme will be used to circumvent the need to solve iteratively nonlinear problems.

Quasi-static crack simulations can be performed by fixing the load (traction or displacement loading) and propagating the cracks by a distance stemming from equilibrium solving. Alternatively, it can be implemented by propagating the cracks by a certain distance and deducing the required load for this propagation to meet equilibrium [17]. The latter approach enables the use of an explicit algorithm, i.e. to set the crack opening increments at the beginning of a step and to deduce the load factor from this information using equilibrium. This principle will be used in the sequel.

5.1 Mixed method for cohesive cracks

With at hand a mixed discretization involving displacement and Lagrange multiplier unknowns, an extrinsic CZM can be used. The use of a mixed formulation involving displacements and tractions will also prove useful to enforce contact conditions precisely. To this end, the two-field formulation of [13] is used.

To simplify the notations, we omit the interface and subdomain indices. The weak form (17a)–(17b) then reads:

$$\left\{ \begin{array}{l} \text{Find } (\mathbf{u}, \boldsymbol{\lambda}) \in V \times \Lambda \text{ such that } \forall (\mathbf{v}, \boldsymbol{\mu}) \in V \times \Lambda \\ \int_{\Omega} \boldsymbol{\varepsilon}(\mathbf{v}) : \mathcal{C} : \boldsymbol{\varepsilon}(\mathbf{u}) \, d\Omega + \int_{\Gamma} \llbracket \mathbf{v} \rrbracket \cdot \boldsymbol{\lambda} \, d\Gamma = \int_{\Gamma_N} \mathbf{v} \cdot \mathbf{t}_N \, d\Gamma, \\ \int_{\Gamma} \boldsymbol{\mu} \cdot \llbracket \mathbf{u} \rrbracket \, d\Gamma = \int_{\Gamma_D} \boldsymbol{\mu} \cdot \mathbf{u}_D \, d\Gamma. \end{array} \right. \quad (67)$$

The cohesive traction $\boldsymbol{\lambda}(\llbracket \mathbf{u} \rrbracket)$ (which is a purely softening function for an extrinsic formulation) is introduced in the weak formulation in its compliance form (see [13])

as follows:

$$\left\{ \begin{array}{l} \text{Find } (\mathbf{u}, \boldsymbol{\lambda}) \in V \times \Lambda \text{ such that } \forall (\mathbf{v}, \boldsymbol{\mu}) \in V \times \Lambda \\ \int_{\Omega} \boldsymbol{\varepsilon}(\mathbf{v}) : \mathbf{C} : \boldsymbol{\varepsilon}(\mathbf{u}) \, d\Omega + \int_{\Gamma} \llbracket \mathbf{v} \rrbracket \cdot \boldsymbol{\lambda} \, d\Gamma = \int_{\Gamma_N} \mathbf{v} \cdot \mathbf{t}_N \, d\Gamma, \\ \int_{\Gamma} \boldsymbol{\mu} \cdot (\llbracket \mathbf{u} \rrbracket - \llbracket \mathbf{u} \rrbracket(\boldsymbol{\lambda})) \, d\Gamma = \int_{\Gamma_D} \boldsymbol{\mu} \cdot \mathbf{u}_D \, d\Gamma. \end{array} \right. \quad (68)$$

With the abuse of notation, $\llbracket \mathbf{u} \rrbracket(\boldsymbol{\lambda})$ denotes the cohesive opening as a function of the cohesive traction, i.e. the compliance form of the cohesive constitutive relation. The second equation of (68) compares the displacement jump calculated from the displacement field with the displacement jump obtained from the cohesive model based on the cohesive traction. Until the activation of a cohesive zone, Eq. (67) and Eq. (68) are the same. Using this formulation, an incremental-iterative solution procedure would be required to solve this nonlinear problem. Since the opening is computed from the Lagrange multiplier in the compliance form as $\llbracket \mathbf{u} \rrbracket = \mathbf{R}\boldsymbol{\lambda}$, some terms in the compliance matrix \mathbf{R} tend to infinity when $\boldsymbol{\lambda}$ tends to $\mathbf{0}$ (i.e. failure of the cohesive zone). This was identified in [13], which motivated the use of a modified Lagrange multiplier method. An effective cohesive traction $\boldsymbol{\zeta}$ is therefore introduced as a Lagrange multiplier defined by

$$\boldsymbol{\zeta} = \boldsymbol{\lambda} + \mathbf{k} \cdot \llbracket \mathbf{u} \rrbracket, \quad (69)$$

where

$$\mathbf{k} = \begin{bmatrix} k_n & 0 \\ 0 & k_t \end{bmatrix} \quad (70)$$

with the parameters k_n and k_t being positive. This allows evaluating the compliance matrix even when $\boldsymbol{\lambda} = \mathbf{0}$. In Section 5.6, lower bounds will be determined for these parameters.

From (68), using (69), one obtains a new weak form expressed as:

$$\left\{ \begin{array}{l} \text{Find } (\mathbf{u}, \boldsymbol{\zeta}) \in V \times \Lambda \text{ such that } \forall (\mathbf{v}, \boldsymbol{\eta}) \in V \times \Lambda \\ \int_{\Omega} \boldsymbol{\varepsilon}(\mathbf{v}) : \mathbf{C} : \boldsymbol{\varepsilon}(\mathbf{u}) \, d\Omega + \int_{\Gamma} \llbracket \mathbf{v} \rrbracket \cdot (\boldsymbol{\zeta} - \mathbf{k} \cdot \llbracket \mathbf{u} \rrbracket) \, d\Gamma = \int_{\Gamma_N} \mathbf{v} \cdot \mathbf{t}_N \, d\Gamma - \int_{\Gamma_D} \mathbf{v} \cdot \mathbf{k} \cdot \mathbf{u}_D \, d\Gamma, \\ \int_{\Gamma} \boldsymbol{\eta} \cdot (\llbracket \mathbf{u} \rrbracket - \llbracket \mathbf{u} \rrbracket(\boldsymbol{\zeta})) \, d\Gamma = \int_{\Gamma_D} \boldsymbol{\mu} \cdot \mathbf{u}_D \, d\Gamma. \end{array} \right. \quad (71)$$

The opening of cracks can cause other cracks to close due to stress redistribution. A contact condition must therefore be provided to avoid interpenetration of the crack faces. During contact the conditions $\llbracket u_n \rrbracket = 0$ and $t_n \leq 0$ hold. Therefore, according to Eq. (69), $\zeta_n \leq 0$. To enforce this in Eq. (71), the interface stiffness is only

decreased by the damage when the cohesive zone is in loading state:

$$\begin{aligned} \int_{\Omega} \boldsymbol{\varepsilon}(\mathbf{v}) : \mathcal{C} : \boldsymbol{\varepsilon}(\mathbf{u}) \, d\Omega + \int_{\Gamma} \llbracket \mathbf{v} \rrbracket \cdot (\boldsymbol{\zeta} - \mathbf{k} \cdot \llbracket \mathbf{u} \rrbracket) \, d\Gamma &= \int_{\Gamma_N} \mathbf{v} \cdot \mathbf{t}_N \, d\Gamma - \int_{\Gamma_D} \mathbf{v} \cdot \mathbf{k} \cdot \mathbf{u}_D \, d\Gamma, \\ \int_{\Gamma} \boldsymbol{\eta} \cdot (\llbracket \mathbf{u} \rrbracket - \chi(\zeta_n > 0) \llbracket \mathbf{u} \rrbracket(\boldsymbol{\zeta})) \, d\Gamma &= \int_{\Gamma_D} \boldsymbol{\eta} \cdot \mathbf{u}_D \, d\Gamma. \end{aligned} \quad (72)$$

where χ is the indicator function acting only in normal direction to activate contact upon closure.

In Section 5.3, the mixed formulation (72) is recast with the introduction of interfacial damage. It will provide a damage propagation criterion and will allow for a segregated solution procedure, i.e. the system solving for $(\mathbf{u}, \boldsymbol{\zeta})$ is separated from the damage update.

5.2 Failure criterion

Mode I crack propagation was mainly considered in previous efforts [25]. Experiments are however often carried out with triaxial tests or involving multiaxial stress states for rock and geomaterials. The modelling of such tests involves compressive-shearing loading on interfaces, and a pure mode I crack propagation model would not be sufficient for such cases. Furthermore, even if uni-axial tension is applied to heterogeneous materials, the slanted interfaces undergo shearing as well. Assuming only mode I propagation is therefore not justified for our purpose.

A model uncoupling failure along the normal and tangential directions is to be avoided as residual normal (tangential) tractions should not be present after the interface has failed in tangential (normal) direction.

A single Mohr-Coulomb criterion would predict an overestimated uni-axial tensile strength. Such a criterion is therefore combined with a Rankine criterion to construct the *Mohr-Coulomb criterion with tension cut-off* (see e.g. [37]). Expressed in the space of normal and tangential cohesive tractions, t_n and t_t , the initial damage criterion consists of the equations

$$\begin{aligned} t_n - f_t &\leq 0, \\ \tan \phi t_n + |t_t| - c &\leq 0, \end{aligned} \quad (73)$$

giving the tension cut-off and the Mohr-Coulomb parts of the criterion, where f_t is the tensile strength, c is the cohesion and ϕ is the angle of friction.

The *Mohr-Coulomb criterion with tension and compression cap* (see Fig. 11) consists of an additional part, described by

$$\begin{aligned} t_n - f_t &\leq 0, \\ \tan \phi t_n + |t_t| - c &\leq 0, \\ (t_n + f_c) \tan \psi - |t_t| &\leq 0, \end{aligned} \quad (74)$$

where f_c is the compressive strength. In [40], the compression cap was an ellipsoid. A compression cap will be shown necessary so that the parameter \mathbf{k} to be selected

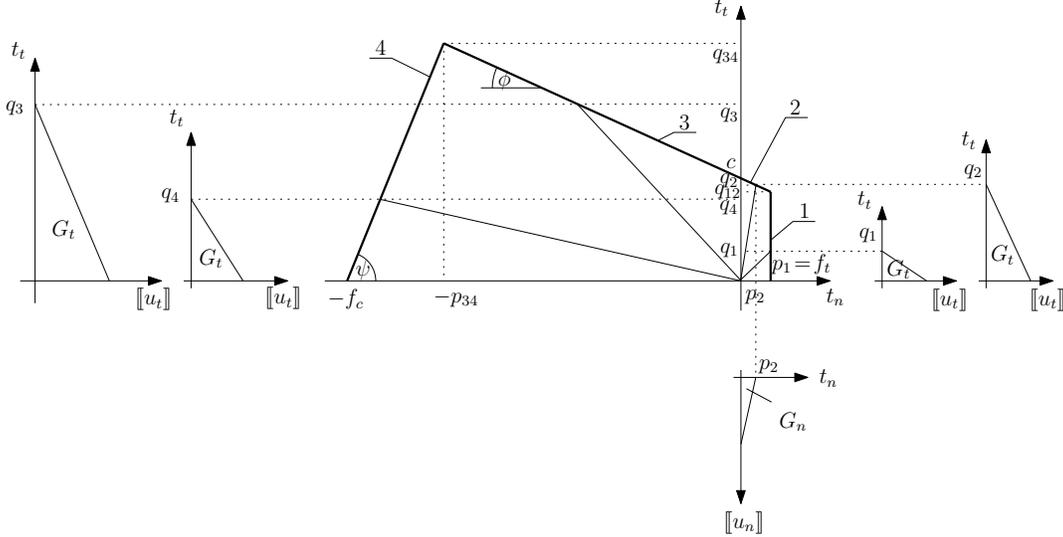


Figure 11: Failure criterion

for defining the augmented Lagrangian remains finite for all traction component combinations (see Section 5.6).

The failure criterion determines which combination of the tractions causes damage initiation and therefore the start of softening. Figure 11 shows the Mohr-Coulomb criterion for positive shear stresses with cut-off in the tension and cap in the compression regimes. One can differentiate four loading modes, depending on which part of the criterion becomes active when the softening begins. Label 1 marks the tension cut-off part, label 2 the part of the Mohr-Coulomb line being in the tension region, label 3 the Mohr-Coulomb line in the compressive regime, while label 4 indicates the compression cap. The corresponding traction-separation relations are also indicated in the figure.

5.3 Damage-based cohesive zone formulation

In what follows, we consider an extrinsic cohesive model (i.e. without any elastic behaviour before softening), suitable for initially perfect interfaces. Following [25], the CZM is reformulated with an interfacial damage variable d . This allows formulating evolution equations, which simplifies the numerical implementation of the CZM. Using only one damage variable for both the normal and tangential directions ensures that whenever the interface is fully failed, both the normal and the tangential cohesive tractions vanish. The free energy functional is defined as

$$\varphi(\llbracket \mathbf{u} \rrbracket, d) = \frac{1}{2} \left(\frac{1}{d} - 1 \right) \llbracket \mathbf{u} \rrbracket \cdot \mathbf{k} \cdot \llbracket \mathbf{u} \rrbracket \quad (75)$$

and \mathbf{k} is defined in Eq. (70).

Another internal variable y is defined as the dual of d , so that evolution laws can be written. The state laws, i.e. the expressions of the cohesive traction \mathbf{t} and of the cohesive energy release rate y , respectively, are derived from the potential (75)

according to:

$$\mathbf{t} = \frac{\partial \varphi}{\partial \llbracket \mathbf{u} \rrbracket} = \left(\frac{1}{d} - 1 \right) \mathbf{k} \cdot \llbracket \mathbf{u} \rrbracket. \quad (76)$$

$$y = -\frac{\partial \varphi}{\partial d} = \frac{1}{2d^2} \llbracket \mathbf{u} \rrbracket \cdot \mathbf{k} \cdot \llbracket \mathbf{u} \rrbracket \quad (77)$$

Equation (76) justifies the choice of the Helmholtz free energy expression: for a vanishing damage, the interface stiffness is infinite, i.e. an extrinsic CZM is obtained. We do not define the equivalent displacement jump and the equivalent traction as it is usually done for mode-mixity. The reason is that the failure criterion discussed in Section 5.2 mostly resides in the compression regime where $\llbracket \mathbf{u} \rrbracket = \llbracket u_t \rrbracket \mathbf{e}_t$ and so no equivalent displacement jump is required.

The Lagrange multiplier $\boldsymbol{\lambda}$ in the pure Lagrange multiplier formulation (68) was identified as the traction vector: $\mathbf{t} = \boldsymbol{\lambda}$. The augmented Lagrange multiplier $\boldsymbol{\zeta}$ in Eq. (69) does not carry a specific meaning. On the other hand, using equations (69) and (76), one defines the modified cohesive law

$$\boldsymbol{\zeta} = \frac{1}{d} \mathbf{k} \cdot \llbracket \mathbf{u} \rrbracket. \quad (78)$$

Determining the fracture energy – discussed in Section 5.4.2 – requires the components of the traction vector. However, only $\llbracket \mathbf{u} \rrbracket$ and $\boldsymbol{\zeta}$ are available after solving (72). The previous equations allow us to determine the traction vector as

$$\mathbf{t} = (1 - d)\boldsymbol{\zeta}. \quad (79)$$

In this context, the Lagrange multiplier $\boldsymbol{\zeta}$ can be thought of as the effective traction vector. With the introduction of the damage variable, the damage indicator χ from the second equation of (72) can be computed as

$$\chi := \begin{cases} 1, & \zeta_n > 0 \quad \text{or} \quad \zeta_n \leq 0 \text{ and } d = 0 \\ 0, & \text{otherwise} \end{cases} \quad (80)$$

Written in terms of the normal and tangential components, and taking into account contact, Eq. (79) is computed as

$$\begin{aligned} t_n &= (1 - \chi d)\zeta_n, \\ t_t &= (1 - d)\zeta_t. \end{aligned} \quad (81)$$

Substituting $\llbracket \mathbf{u} \rrbracket$ ($\boldsymbol{\zeta}$) from Eq. (78) into Eq. (72) defines the final set of equations that will be used in the discretization procedure:

$$\left\{ \begin{array}{l} \text{Find } (\mathbf{u}, \boldsymbol{\zeta}) \in V \times \Lambda \text{ such that } \forall (\mathbf{v}, \boldsymbol{\eta}) \in V \times \Lambda \\ \int_{\Omega} \boldsymbol{\varepsilon}(\mathbf{v}) : \mathbf{C} : \boldsymbol{\varepsilon}(\mathbf{u}) \, d\Omega + \int_{\Gamma} \llbracket \mathbf{v} \rrbracket \cdot (\boldsymbol{\zeta} - \mathbf{k} \cdot \llbracket \mathbf{u} \rrbracket) \, d\Gamma = \int_{\Gamma_N} \mathbf{v} \cdot \mathbf{t}_N \, d\Gamma - \int_{\Gamma_D} \mathbf{v} \cdot \mathbf{k} \cdot \mathbf{u}_D \, d\Gamma, \\ \int_{\Gamma} \boldsymbol{\eta} \cdot (\llbracket \mathbf{u} \rrbracket - \chi d \mathbf{k}^{-1} \cdot \boldsymbol{\zeta}) \, d\Gamma = \int_{\Gamma_D} \boldsymbol{\eta} \cdot \mathbf{u}_D \, d\Gamma. \end{array} \right. \quad (82)$$

The second term of the integrand of the second equation of Eq. (82) written in the basis of the coordinate system local to the interface should be interpreted as

$$\begin{bmatrix} k_n^{-1} & 0 \\ 0 & k_t^{-1} \end{bmatrix} \begin{bmatrix} d & 0 \\ 0 & d \end{bmatrix} \begin{bmatrix} \chi \zeta_n \\ \zeta_t \end{bmatrix},$$

in which the contact condition makes the problem nonlinear. To solve the global system in a linear fashion, at a given damage step s , $\zeta^{(s-1)}$ is used as an approximation in (82).

The discretization of equations (82) results a similar matrix as in (48). The discretization of the displacement and the Lagrange multiplier fields does not require a mesh on the interface. To keep the robustness, we use the same basis for the damage interpolation as for the Lagrange multiplier interpolation:

$$d^i = \sum_{j \in \tilde{\mathcal{M}}_{\Gamma}^i} D_j^i \tilde{\psi}_j^i, \quad (83)$$

where $\tilde{\psi}_j^i$ is a component of $\tilde{\boldsymbol{\psi}}_j^i$. The new discretized formulation then reads:

$$\begin{aligned} a(\mathbf{u}_h, \mathbf{v}_h) &= \sum_{i \in I_s} \sum_{j, k \in \mathcal{M}^i} V_j^i K_{jk}^i U_k^i + \sum_{i \in I_{\text{int}}} \sum_{j, k \in \mathcal{M}^{i\pm}} (V_j^{i+} - V_j^{i-}) P_{jk}^i (U_k^{i+} - U_k^{i-}) + \\ &+ \sum_{i \in I_D} \sum_{j, k \in \mathcal{M}^{i+}} V_j^{i+} P_{jk}^i U_k^{i+}, \end{aligned} \quad (84)$$

$$b(\mathbf{u}_h, \boldsymbol{\eta}_h) = \sum_{i \in I_{\text{int}}} \sum_{j \in \tilde{\mathcal{M}}_{\Gamma}^i} \sum_{k \in \mathcal{M}^{i\pm}} E_j^i B_{jk}^i (U_k^{i+} - U_k^{i-}) + \sum_{i \in I_D} \sum_{j \in \tilde{\mathcal{M}}_{\Gamma}^i} \sum_{k \in \mathcal{M}^{i+}} E_j^i B_{jk}^i U_k^{i+}, \quad (85)$$

$$c(\boldsymbol{\zeta}_h, \boldsymbol{\eta}_h) = \sum_{i \in I_{\text{int}}} \sum_{j, k \in \tilde{\mathcal{M}}_{\Gamma}^i} E_j^i C_{jk}^i Z_k^i, \quad (86)$$

$$f(\mathbf{v}_h) = \sum_{i \in I_N} \sum_{j \in \mathcal{M}^{i+}} V_j^{i+} F_j^i + \sum_{i \in I_D} \sum_{j \in \mathcal{M}^{i+}} V_j^{i+} H_j^i, \quad (87)$$

$$g(\boldsymbol{\eta}_h) = \sum_{i \in I_D} \sum_{j \in \tilde{\mathcal{M}}_{\Gamma}^i} E_j^i G_j^i \quad (88)$$

where the new matrices and vector are given by

$$P_{jk}^i = - \int_{\Gamma^i \cup \Gamma_D^i} \boldsymbol{\psi}_j^i \cdot \mathbf{k}^{-1} \cdot \boldsymbol{\psi}_k^i \, d\Gamma, \quad (89)$$

$$C_{jk}^i = - \int_{\Gamma^i \cup \Gamma_D^i} d^i \tilde{\boldsymbol{\psi}}_j^i \cdot \mathbf{k}^{-1} \cdot \tilde{\boldsymbol{\psi}}_k^i \, d\Gamma, \quad (90)$$

$$H_j^i = - \int_{\Gamma_D^i} \mathbf{u}_D^i \cdot \mathbf{k} \cdot \boldsymbol{\psi}_j^i \, d\Gamma. \quad (91)$$

The resulting algebraic system reads

$$\begin{bmatrix} \mathbf{K} + \mathbf{L} & \mathbf{B}^{\top} \\ \mathbf{B} & \mathbf{C} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{z} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{g} \end{bmatrix}, \quad (92)$$

where the stiffness matrix \mathbf{K} , the coupling matrix \mathbf{B} and the vector \mathbf{g} are the same as in Section 3.2 (cf. Eq. (44), (45), (47)). The matrix \mathbf{L} has a block structure

$$\mathbf{L} = \begin{bmatrix} -\mathbf{P} & \mathbf{P} \\ \mathbf{P} & -\mathbf{P} \end{bmatrix} \quad (93)$$

due to the jump operator.

Remark 7 *The Lagrange multiplier field ζ is approximated in the same way as the original Lagrange multiplier λ . This is made possible because both λ and $[[\mathbf{u}]]$ (ζ) are defined in the same basis (as opposed to $[[\mathbf{u}]]$, which is given in the nodal basis).*

Remark 8 *The matrix \mathbf{L} couples displacement degrees of freedom from neighbouring subdomains. It implies that the block-diagonal structure of \mathbf{K} is not preserved in the system matrix.*

The matrix of \mathbf{k} in the $\{\mathbf{e}_n, \mathbf{e}_t\}$ basis attached to the interface is diagonal:

$$\mathbf{k} = \begin{bmatrix} k_n & 0 \\ 0 & k_t \end{bmatrix}. \quad (94)$$

Note that k_n and k_t do not represent by any means the elasticity of the CZM, as they are only used to deal with completely open cracks. At the start of the simulation, assuming no initial damage ($\mathbf{C} = \mathbf{0}$) and no penalty term ($\mathbf{L} = \mathbf{0}$), the stable $\mathbf{u}-\lambda$ approximation described in Section 4 ensures an accurate calculation of the interfacial traction field.

Note that only the \mathbf{C} matrix contains the damage variable. Therefore, only this matrix needs to be updated during crack propagation, with all other matrices to be computed only once. Furthermore, for efficiency reasons, the integral of the shape functions in (90) can be precomputed so that only the multiplication by the nodal damage values is required during the update part. More specifically, using the damage interpolation (83), matrix (90) can be written as

$$C_{jk}^i = \sum_{\ell \in \tilde{\mathcal{M}}_T^i} \tilde{C}_{jkl}^i D_\ell^i, \quad (95)$$

where

$$\tilde{C}_{jkl}^i = - \int_{\Gamma^i \cup \Gamma_D^i} \tilde{\psi}_j^i \cdot \mathbf{k}^{-1} \cdot \tilde{\psi}_k^i \tilde{\psi}_\ell^i d\Gamma \quad (96)$$

can be precomputed and stored. Equation (96) shows that computing \mathbf{C} requires more quadrature points than what is needed for the computation of \mathbf{B} in (45) and \mathbf{P} in (89). Algorithm 3 in Appendix A details the preprocessing phase for the presented discretization framework.

5.4 Computing the energetic quantities

Given the damage state, the discrete displacement and Lagrange multiplier values are obtained by solving the generalized saddle point problem (92). The new damage

state at each point is computed from the evolution laws given by the classical Kuhn-Tucker relations

$$\dot{d} \geq 0, \quad (97)$$

$$y - y_c(d) \leq 0, \quad (98)$$

$$(y - y_c(d)) \dot{d} = 0, \quad (99)$$

prescribing damage growth as an irreversible process, and expressing that damage does not increase when the energy release rate remains under a critical value characteristic of the interface. Damage propagation occurs when the critical energy release rate is reached. In (98)–(99), $y_c(d)$ is an increasing function of d , so that an increasing energy is required to further increase the damage on an interface. Section 5.5 deals with the discretization of (97)–(99).

Depending on the normal and tangential traction components at a given location, four characteristic loading modes can be distinguished. The notations can be followed in Fig. 11. The remainder of this section is separated into three parts. First, the energy release rate is determined, then assumptions on the mixed mode fracture energy are introduced, finally the critical energy release rate is computed.

5.4.1 Energy release rate

As \mathbf{k} is diagonal according to Eq. (70), the energy release rate in Eq. (77) can be rewritten as

$$y = \frac{1}{2d^2} k (\llbracket u_n \rrbracket^2 + \llbracket u_t \rrbracket^2), \quad (100)$$

where the assumption $k = k_n = k_t$ was made. Note that the displacements are available in the mesh nodes, therefore u_n and u_t are not directly available on the interface. Taking their restriction on the interface results in an oscillatory approximate displacement field. Therefore, we want to express $\llbracket u_n \rrbracket$ and $\llbracket u_t \rrbracket$ with the help of the Lagrange multipliers because they are free of oscillations. This is doable because the Lagrange multipliers are in connection with the tractions, which are in turn known from the cohesive law. Since the softening in normal and tangential directions is separated according to (76), we have

$$\begin{aligned} t_n &= \left(\frac{1}{d} - 1 \right) k \llbracket u_n \rrbracket, \\ t_t &= \left(\frac{1}{d} - 1 \right) k \llbracket u_t \rrbracket. \end{aligned} \quad (101)$$

Based on this, the energy release rate can be decomposed to normal and tangential parts as

$$y = y_n + y_t, \quad y_n = \frac{1}{2k} \left(\frac{\max\{0, t_n\}}{1-d} \right)^2, \quad y_t = \frac{1}{2k} \left(\frac{t_t}{1-d} \right)^2, \quad (102)$$

where we postulated that no energy is released in pure compression. Although y is not defined for $d = 1$ above, it does not pose any problem because a completely failed cohesive zone cannot release energy anyway. In the implementation, this issue can be avoided by noticing that d can be eliminated from y by using (79):

$$y = y_n + y_t, \quad y_n = \frac{1}{2k} (\max\{0, \zeta_n\})^2, \quad y_t = \frac{1}{2k} \zeta_t^2. \quad (103)$$

5.4.2 Decomposing the mixed mode fracture energy

We make the hypothesis that the mixed mode fracture energy G_c can be additively decomposed into a normal and a tangential part (G_n and G_t , respectively):

$$G_c = G_n + G_t, \quad (104)$$

where G_n and G_t are determined from the failure criterion.

For geomechanical applications, the mixed mode fracture energy under compressive stress states increases with compression. Therefore, a model inspired by [27, 44] is applied, imposing a linear increase of the fracture energy as a function of the normal compressive stress component on the interface. The expressions of G_n and G_t , respecting (104), are collected in Tab. 1 for the four possible loading modes. For a general loading mode, the point (p, q) on the failure surface gives the strength values. In this table, G_I and G_{II} are the pure mode I and mode II fracture energies, and $a > 0$ is a coefficient which sets the growth of the mixed mode fracture energy with increasing compressive confinement on the interface. As a simplification, it is assumed that once the damage initiates at a location, the fracture energy for that location does not change with t_n if t_n further evolves in the course of degradation.

5.4.3 Critical energy release rate

Given a cohesive relation expressed in traction-separation form $\mathbf{t}(\llbracket \mathbf{u} \rrbracket)$, the critical energy release rate $y_c(d)$ is determined such that the potential-based cohesive model is equivalent to the physical cohesive relation. The linear extrinsic cohesive relation can be given by two parameters, the strength and the fracture energy (see Fig. 11):

$$t_n = p - \frac{p^2}{2G_n} \llbracket u_n \rrbracket, \quad (105)$$

$$t_t = q - \frac{q^2}{2G_t} \llbracket u_t \rrbracket. \quad (106)$$

Equating them with the tractions derived from the free energy in Eq. (101), the cohesive openings $\llbracket u_n \rrbracket$ and $\llbracket u_t \rrbracket$ can be isolated. They are then substituted to Eq. (100), and using the fact that $y = y_c$ during damage propagation, the critical energy release rate is obtained as

$$y_c(d) = \begin{cases} \frac{k}{2} \left[\left(\frac{A_n}{(1-d)k + B_n d} \right)^2 + \left(\frac{A_t}{(1-d)k + B_t d} \right)^2 \right] & \text{tension and shear} \\ \frac{k}{2} \left(\frac{f_t}{(1-d)k + f_t^2/(2G_I)d} \right)^2 & \text{pure tension} \\ \frac{k}{2} \left(\frac{c}{(1-d)k + c^2/(2G_{II})d} \right)^2 & \text{pure shear} \\ \frac{k}{2} \left(\frac{A_t}{(1-d)k + B_t d} \right)^2 & \text{compression and shear} \end{cases} \quad (107)$$

where $A_n = p$, $B_n = p^2/(2G_n)$, $A_t = q$, $B_t = q^2/(2G_t)$. The restrictions on the value of k can be found in Section 5.6. Table 1 summarizes the computation of the energetic quantities for the four loading modes.

quantity	loading mode (see Fig. 11)			
	1	2	3	4
G_n	$\left(1 - \frac{q}{c}\right) G_I$	$\left(1 - \frac{q}{c}\right) G_I$	0	0
G_t	$\frac{q}{c} G_I$	$G_{II} - \frac{c-q}{c-q_c} \left(G_{II} - \frac{q_c}{c} G_I\right)$	$G_{II} - at_n$	$G_{II} - ap_{34}$
y_n	$\frac{\zeta_n^2}{2k}$	$\frac{\zeta_n^2}{2k}$	0	0
y_t	$\frac{\zeta_t^2}{2k}$	$\frac{\zeta_t^2}{2k}$	$\frac{\zeta_t^2}{2k}$	$\frac{\zeta_t^2}{2k}$
$y_{c,n}$	$\frac{k}{2} \left(\frac{A_n}{(1-d)k + B_n d} \right)^2$		with $A_n = p$, $B_n = \frac{p^2}{2G_n}$	
$y_{t,n}$	$\frac{k}{2} \left(\frac{A_t}{(1-d)k + B_t d} \right)^2$		with $A_t = q$, $B_t = \frac{q^2}{2G_t}$	

Table 1: Energetic quantities for the different loading modes

5.5 Damage update algorithm

The crack propagation is governed by the evolution equations (97)–(99). Upon crack initiation or crack extension, a given amount of energy dissipates, governed by the damage variable d . To explain the shortcoming of using d for energy dissipation, let us introduce the so-far dissipated energy at any location: $G : [0, 1] \rightarrow [0, G_c]$ defined as

$$G(d) = \int_0^d y_c(\hat{d}) d\hat{d}. \quad (108)$$

Upon complete failure, $G(1) = G_c$. The dissipated energy increment for a variation of d is

$$\Delta G = \int_d^{d+\Delta d} y_c(\hat{d}) d\hat{d}. \quad (109)$$

Substituting y_c from (107) into (109), we realize that ΔG depends on the strength and fracture energy parameters p , q , G_n , G_t . It means that the energy dissipation is in general not the same for two locations i and j even if $d_i = d_j$ and $\Delta d_i = \Delta d_j$. In case of a pure Rankine failure criterion (without Mohr-Coulomb failure in compressive states), and only one interface, as considered in [25], the strength and the fracture energy is spatially uniform for all locations and therefore the dissipated energy at a given step is proportional to the damage increment. This is no longer the case for multiple interfaces having different material parameters or for non-uniform fracture energy on a single interface. This would cause that with a pure control by the damage d , the majority of the energy dissipation may not be attributed to the most favourable locations, as expected physically.

To cure this behaviour, a physically meaningful quantity, the so-far dissipated energy G introduced above (rather than the damage), is chosen which describes the

degradation of the cohesive interfaces, in order to drive the computation. We wish to express the evolution equations with G . Since $G'(d) = y_c$ from (108) and $y_c > 0$, Eq. (97) is replaced by

$$\dot{G} \geq 0. \quad (110)$$

By parametrizing y_c with G , the other two evolution equations (98) and (99) are

$$y - y_c(G) \leq 0, \quad (111)$$

$$(y - y_c(G)) \dot{G} = 0. \quad (112)$$

Combining these results, the discrete versions of the evolution equations at step n read

$$\Delta G_i^{(n)} \geq 0, \quad (113)$$

$$f_i^{(n)}(\mu, G_i) \approx f_i^{(n-1)} + \alpha_i^{(n-1)} \Delta \mu^{(n)} + \beta_i^{(n-1)} \Delta G_i^{(n)} \leq 0, \quad (114)$$

$$f_i^{(n)} \Delta G_i^{(n)} = 0 \quad (115)$$

in which

$$\alpha_i^{(n-1)} = 2\mu y_i \left(d_i^{(n-1)}, \mathbf{t}_i^{(n)} \right), \quad \beta_i^{(n-1)} = -\frac{y'_c \left(d_i^{(n-1)} \right)}{y_c \left(d_i^{(n-1)} \right)}. \quad (116)$$

Step numbers are in superscript and between parentheses, while location indices are in subscripts. In this section, the index i refers to a location on the set of all the interfaces where the Lagrange multipliers are defined (i.e. on the internal interfaces and on the Dirichlet boundaries). In other words, we do not make a distinction on which interface a location lies. This allows the damage field evolving automatically, no topological information is needed. The set of all locations is denoted by \mathcal{I} . Some of these locations have completely failed by step n . The not yet failed ones are indexed with $\mathcal{J} \subseteq \mathcal{I}$.

Computing $\Delta G_i^{(n)}$ is performed similarly, but with relevant modifications, as it was done in [25] for $\Delta d_i^{(n)}$. Rather than prescribing load increments, the quasi-static problem is solved stepwise by prescribing fractions of G_c increments², and deducing the load level μ multiplying a given reference load F_{ref} ,

$$F = \mu F_{\text{ref}} \quad (117)$$

such that equilibrium is obtained for the prescribed dissipation state. The step indices are omitted in the sequel for clarity, unless otherwise stated. From Eq. (103) we see that $y \propto \zeta^2$ and the linearity of Eq. (82) implies that $\zeta \propto F \propto \mu$, where the last proportionality comes from the definition of the load factor in Eq. (117). It shows that the energy release rate scales quadratically with the load factor. The energy release rate obtained for the reference load can then be scaled in such a way that the evolution criterion (111) is satisfied for all discrete Lagrange multipliers:

$$f_i(\mu, G_i) = \mu^2 y_i - y_c(G_i) \leq 0. \quad (118)$$

²We will show soon how damage increments can be obtained from dissipated energy increments.

Based on this information, the value of the load factor can then be identified as corresponding to the most critical location:

$$\mu = \min_{i \in \mathcal{J}} \sqrt{\frac{y_c(G_i)}{y_i}}. \quad (119)$$

Here, we note that y_c must be computed with the true fracture energy, not with the one which comes from the arbitrarily chosen reference load. This means that for the calculation of G_c in (104), with G_n and G_t given in Tab. 1, the true normal traction component is taken into account to evaluate the mixed mode fracture energy. This is obtained by finding the intersection of the initial damage surface with the half-line stemming from the origin and given by the loading direction.

Although (118) is fulfilled with the chosen μ , it does not optimize the new damage front position, as one could wonder whether damage should grow only at one position (i.e. the most critical one), or at other positions too. An iteration between the old and the new damage states could be used to answer this question [8], however a simpler scheme is used here. At this point, α_i are known and β_i are determined from the cohesive model. Instead of the inequality (114), a prediction is used based on

$${}^{\text{pred}}f_i(\mu, G_i) = f_i + \alpha_i {}^{\text{pred}}\Delta\mu + \beta_i \Delta G_i = 0. \quad (120)$$

To avoid the use of an iterative scheme, an explicit approach is used and we solve for ${}^{\text{pred}}\Delta\mu$ based on a chosen maximum energy dissipation increment ΔG_{max} . First, the predicted load factor increment is expressed as

$${}^{\text{pred}}\Delta\mu = \min_{i \in \mathcal{J}} \frac{-f_i - \beta_i \Delta G_{\text{max}}}{\alpha_i} \quad (121)$$

and the corresponding dissipation increments are computed using

$$\Delta G_i = \max \left\{ 0, \frac{-f_i - \alpha_i {}^{\text{pred}}\Delta\mu}{\beta_i} \right\}. \quad (122)$$

Note that Eq. (122) filters out the possibly negative increments, thereby obeying Eq. (113).

The remaining question is how to choose the step size ΔG_{max} . Since the fracture energy, in general, is different from location to location, dissipating a fixed amount of energy would not be reasonable. Therefore, ΔG_{max} is chosen to be a fraction of the fracture energy corresponding to the most critical location (i.e. where the load factor is determined for in Eq. (119)) at a given step:

$$\Delta G_{\text{max}} = \Delta \tilde{G}_{\text{max}} G_{c,\ell}, \quad 0 < \Delta \tilde{G}_{\text{max}} \leq 1, \quad \ell \text{ is the most critical location} \quad (123)$$

We call the constant of proportionality $\Delta \tilde{G}_{\text{max}}$ *maximum allowed relative energy dissipation increment* and it is a parameter of the solution procedure. As opposed to $\Delta \tilde{G}_{\text{max}}$, the step size ΔG_{max} is not a global parameter. Generally, it changes from step to step because the most critical location also changes as the body continues

to deform and degrade. Given the explicit nature of the approach, a jagged load-displacement response is expected, with larger oscillations for a larger $\Delta\tilde{G}_{\max}$. This choice for $\Delta\tilde{G}_{\max}$ may cause another location i to dissipate more than what is available: $G_i^{(n)} = G_i^{(n-1)} + \Delta G_i^{(n)} > G_{c,i}$. In that case, we set $G_i^{(n)} = G_{c,i}$. This inaccuracy is due to the explicit nature of our method; decreasing $\Delta\tilde{G}_{\max}$ in (123) results in a better resolution of the true dissipation.

One more ingredient is required for the algorithm to work properly. At any location, as soon as the damage initiates, the strength and the fracture energy are assumed to be constant for that location during the subsequent softening (cf. Section 5.4.2). One can imagine a situation in which later in the course of softening, due to stress redistribution, the local loading state at a location significantly changes. For instance, a location on an interface was under compression, when G_c was fixed. Later in the simulation, if that location experiences tension for which the corresponding fracture energy is much smaller, fixing G_c to the compressive value would result a major inaccuracy. To avoid these cases, a tolerance is set such that energy starts being dissipated only if this dissipation is greater than that tolerance. Of course, the tolerance b must be relative, and is therefore set to be a fraction of $\Delta\tilde{G}_{\max}$:

$$b = \frac{1}{f_1} \Delta\tilde{G}_{\max}, \quad f_1 > 0 \quad (124)$$

By introducing the *relative energy dissipation increment*, defined as

$$\Delta\tilde{G}_i = \frac{\Delta G_i}{G_{c,i}}, \quad (125)$$

the dissipation increment predicted by (122) is not accepted if

$$\Delta\tilde{G}_i < b, \quad \text{and } G_i = 0 \quad (126)$$

As the weak form (82) is coupled with d and not with G , we determine $\Delta d_i^{(n)}$ from the following equation:

$$\Delta G_i^{(n)} = \int_{d_i^{(n-1)}}^{d_i^{(n-1)} + \Delta d_i^{(n)}} y_c(\hat{d}) \, d\hat{d}. \quad (127)$$

Finally, the damage field is updated to obtain the new damage state at every location:

$$d_i^{(n)} = d_i^{(n-1)} + \Delta d_i^{(n)}. \quad (128)$$

Algorithm 4 summarizes the update of the damage field.

5.6 Choosing the k parameter

As mentioned in Section 5.4, the critical energy release rate must be an increasing function of the damage. Taking the derivative of $y_c(d)$ in Eq. (107), one can easily derive the sufficient condition

$$k > k_{\min} = \max \begin{cases} \max\{B_{n,\max}, B_{t,\max}\} & \text{tension and shear} \\ f_t^2/(2G_I) & \text{pure tension} \\ c^2/(2G_{II}) & \text{pure shear} \\ B_{t,\max} & \text{compression and shear} \end{cases} \quad (129)$$

The positivity of $y'_c(d)$ is required on all parts of the failure criterion. The maximum values are to be found so that k is independent of the stress state. If k depended on the local stress values, a complicated and very costly algorithm would have to be used which results a non-symmetric matrix and all global matrices would need to be assembled at each step. This would clearly destroy all the advantages of the presented method.

In the compression region of the failure criterion, along the Mohr-Coulomb line (label 3 in Fig. 11), only B_t is active and is given by

$$B_t(q) = \frac{q^2}{2(G_{\text{II}} + ap(q))} = \frac{q^2}{2\left(G_{\text{II}} + a\frac{q-c}{\tan\phi}\right)}. \quad (130)$$

Asymptotically, $B_t = \mathcal{O}(q)$ and so it is unbounded. That is why the compression cap was introduced to the model, which restricts the strength q to the interval $[c, q_{34}]$. Function B_t takes its global maximum either at the end points $(0, c)$ and $(-p_{34}, \pm q_{34})$ or where its derivative vanishes. Therefore,

$$B_{t,\max} = \max\left\{\frac{c^2}{2G_{\text{II}}}, \frac{q_{34}^2}{2(G_{\text{II}} + ap_{34})}, \frac{2\tan\phi(ac - G_{\text{II}}\tan\phi)}{a^2}\right\}. \quad (131)$$

It can be seen in Fig. 11 that

$$p_{34} = \left|\frac{c - f_c \tan\psi}{\tan\phi + \tan\psi}\right|, \quad q_{34} = c + p_{34} \tan\phi \quad (132)$$

In the tension region of the failure criterion, the safe choice

$$\max\{B_{n,\max}, B_{t,\max}\} = \frac{c^2}{2G_{\text{I}}} \quad (133)$$

is made. Taking into account the values in (131) and in (133), the lower bound on the parameter k is set to

$$k_{\min} = \max\left\{\frac{c^2}{2G_{\text{I}}}, \frac{c^2}{2G_{\text{II}}}, \frac{q_{34}^2}{2(G_{\text{II}} + ap_{34})}, \frac{2\tan\phi(ac - G_{\text{II}}\tan\phi)}{a^2}\right\}. \quad (134)$$

For optimal conditioning, k must be selected such that the conditioning of (92) is good, in which matrices \mathbf{L} and \mathbf{C} contain k . For the (1,1) block to be well-conditioned, entries of similar magnitude must be present in \mathbf{K} and in \mathbf{L} . In one-dimensional finite elements, $k \approx Eh$ is a good choice, where h is the characteristic mesh size.

5.7 Outline of the global solution scheme

Combining the discretization proposed in Section 3 with the present damage update procedure, a crack propagation problem for the heterogeneous grain assembly is solved according to Alg. 2.

Finally, we note that when computing the energetic quantities, we do not need to know which interface a certain Lagrange multiplier corresponds to. In other words, the update procedure is automatic, no geometrical or topological information is required.

Algorithm 2 Quasi-static simulation

- 1: Inputs: parameters required by the algorithms called by this algorithm
 - 2: Determine k_{\min} by (134) in Section 5.6
 - 3: Get global matrices \mathbf{K} , \mathbf{P} , \mathbf{B} and vectors \mathbf{f} , \mathbf{g} by Alg. 3
 - 4: Initialize the damage field
 - 5: **for** n **from** 0 **to** N_{step} **do** ▷ damage stepping loop
 - 6: **if** $n = 0$ **then**
 - 7: Precompute and store $\tilde{\mathbf{C}}$ (96)
 - 8: **end if**
 - 9: Create $\mathbf{C}^{(n)}$ from $\tilde{\mathbf{C}}$ using (95)
 - 10: Solve system (92) with $\mathbf{C}^{(n)}$
 - 11: Save variables of interest, if desired
 - 12: Update the damage field (Section 5.5, Alg. 4)
 - 13: **end for**
-

6 Examples

The proposed framework is now illustrated by means of two examples. In Section 6.1, the three-point bending of a concrete structure is simulated and compared with experimental data. This is a classical mode I test for the fracture of quasi-brittle materials. Although there is a single vertical interface, it incorporates two main components developed here: a non-matching mesh and the damage formulation of the CZM. The second example is a test inspired by the Brazilian test (indirect tension test) applied to a heterogeneous material made of a dense packing of grains. This second example is of course more challenging because it involves interfaces that are slanted with respect to the main loading directions and that are therefore locally subjected to mixed loading. Moreover, the contact formulation is important as well in this second problem.

6.1 Three-point bending

The body has length $L = 1088$ mm, height $c = 500$ mm, depth 40 mm and is supported in its bottom left and bottom right corners over a width $v/2 = L/72$. It is loaded in the middle of the top edge over a width $v = L/36$. The geometrical and material data of this test are taken from [21]. The concrete is considered to be isotropic and linear elastic with Young's modulus $E = 37$ GPa and Poisson's ratio $\nu = 0.21$.

f_t (MPa)	$t_{n,k}$ (MPa)	$[[u_{n,c}]]$ (μm)	$[[u_{n,1}]]$ (μm)	G_f (N m^{-1})	G_F (N m^{-1})
3.92	0.588	94.8	25.3	49.56	70

Table 2: Cohesive properties of the three-point bending specimen from [21]. Notations can be followed in Fig. 13.

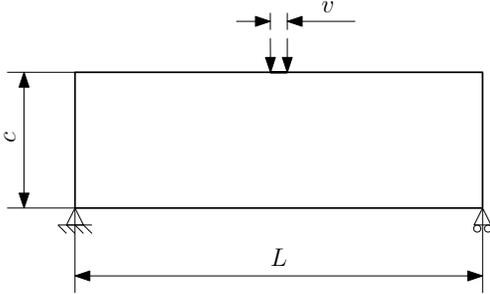


Figure 12: Sketch of the three-point bending problem

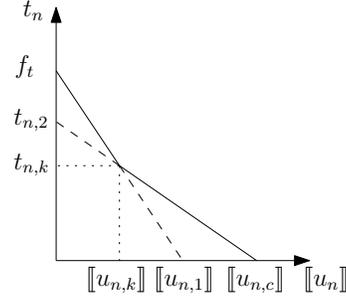


Figure 13: Bilinear softening function for pure mode I loading

The linear cohesive model is not a suitable model for characterizing the softening of concrete. Using a bilinear softening law is appropriate to approximate the load-displacement response measured in laboratories [3]. The slope of the first part of the model, shown in Fig. 13, influences the peak load, while the second part characterizes the pull-out process of the grains [47].

The cohesive properties can be found in Tab. 2. Two fracture energy quantities are defined in [3]: G_F is the total fracture energy, i.e. $G_F = 1/2(f_t [[u_{n,k}]] + t_{n,k} [[u_{n,c}]])$, while $G_f = 1/2 f_t [[u_{n,1}]]$ corresponds to the area under the initial slope.

The numerical parameter k , involved in the definition of the effective cohesive traction vector ζ , appears in the computation of the energy release rate and of the critical energy release rate (Tab. 1), as well as in the weak form (82). As the fracture energy is uniform in this test setup, the minimum value of k is determined as

$$k_{\min} = \frac{f_t^2}{2G_f} \approx 155 \text{ N mm}^{-3} \quad (135)$$

according to (129). In this problem $k = 2000 \text{ N mm}^{-3}$ was chosen, which is large enough to avoid any numerical artefacts. We observed that k could take values from a large interval without causing numerical difficulties. The simulation is performed in plane strain conditions with different maximum damage increments Δd_{\max} . This is possible because the single interface is in mode I, and therefore a computation based on a control driven by $\Delta \tilde{G}_{\max}$ would lead to the same solution.

The resulting load-CMOD (crack mouth opening displacement) curves in Fig. 14 show excellent agreement with the experimental data and indicate that the solution converges as $\Delta d_{\max} \rightarrow 0$. The oscillations due to the explicit nature of the damage update clearly decrease with small enough damage steps.

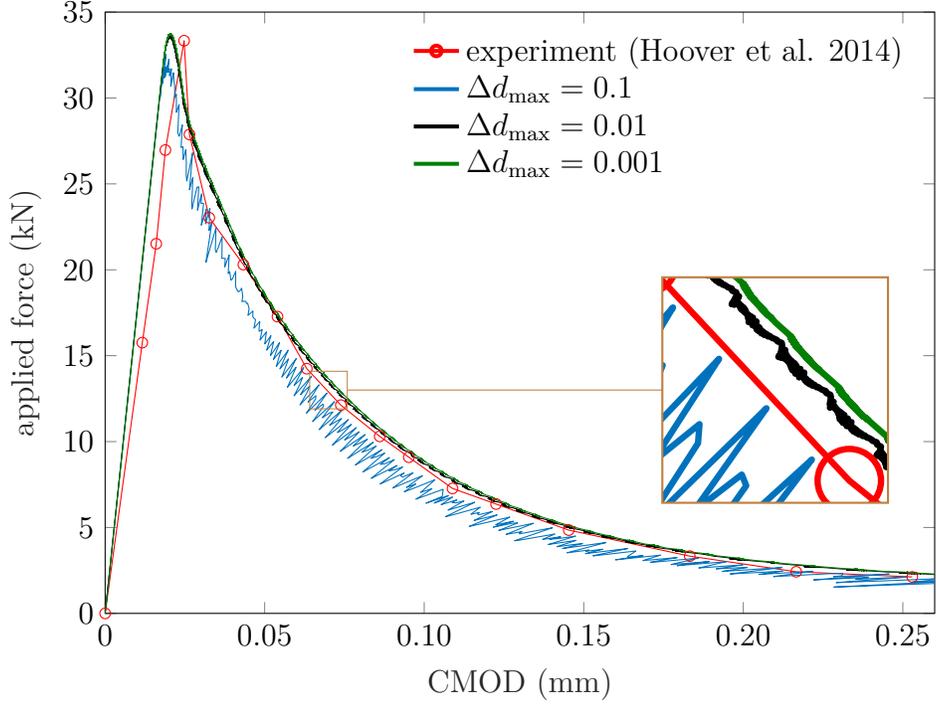


Figure 14: Load-displacement curve of the three-point bending test

6.2 Brazilian test inspired simulations

The Brazilian test, also called as indirect tension test, is a widely used laboratory test to indirectly evaluate the tensile strength of quasi-brittle materials under the assumptions of uniform homogeneous material properties [26]. Using the Brazilian test for complex heterogeneous rocks is however made more difficult to interpret as the heterogeneity introduces deviations with respect to a pure mode I tensile stress state in the centre of the body, and the resulting cracking may be more jagged and distributed. In this section, we follow the softening response of the sample until complete failure.

From a computational point of view, the Brazilian test on grain-based materials is a good example to test the robustness of our discretization method and of the damage propagation criterion. It involves interfaces that are slanted with respect to the main loading direction and that are therefore locally subjected to mixed mode loading. Moreover, the contact formulation is important as well in this second problem. A simplified 2D model of this test configuration is depicted in Fig. 15. The circle is approximated as a regular polygon. The number of sides was set to 44 and the boundary conditions were explicitly given along the top and bottom interfaces. This choice allows a proper representation of the geometry, while at the same time not causing numerical problems in the presence of excessively concentrated forces at the top. The non-zero Neumann boundary condition at the top and the zero Dirichlet boundary condition at the bottom are prescribed along one side of the polygon. We observed that a too small Dirichlet boundary fails to properly fix the body and the disk rotates as the load is applied. The application of the loading in the described manner corresponds to a central angle of about 8.18° , which is close to 10° ,

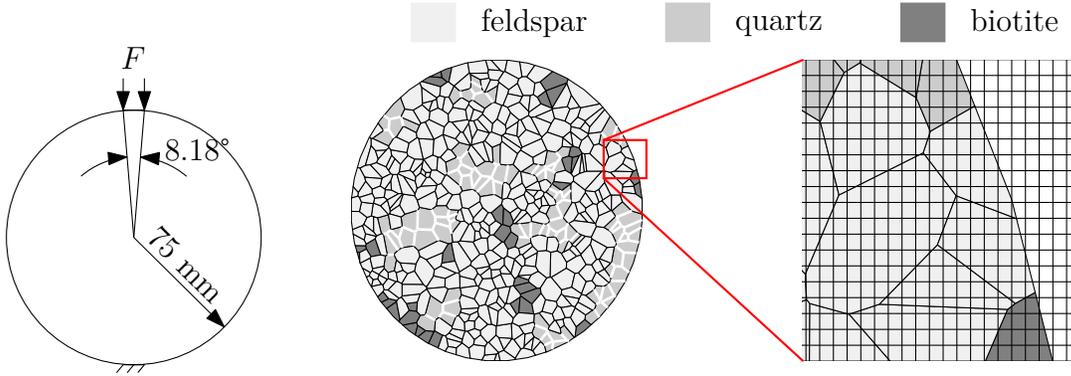


Figure 15: Sketch of the Brazilian test problem and the distribution of the mineral species

\tilde{K} (-)	$\Delta\tilde{G}_{\max}$ (-)	nElem (-)
1.01	0.1	150×150

Table 3: Solution procedure parameters for the Brazilian test

recommended by the standard [22]. We consider a disk of diameter $D = 150$ mm. The simulation is performed under plane strain conditions. To make the example representative, a sample with 500 grains was generated. Figure 15 shows that it provides more than ten grains along the diameter, as recommended by the standard [22]. The circle, or more precisely its polygonal approximation, was embedded into a square background mesh consisting of 150×150 elements. We note that the large number of elements in the background mesh which do not belong to any grain does not increase the computational time (cf. Remark 3).

The simulation was carried out with the solution procedure parameters in Tab. 3 and with the cohesive properties in Tab. 4, the latter inspired by the literature [45]. In Tab. 3, $\tilde{K} = k/k_{\min}$ and nElem denotes the number of elements in the background mesh. The first five parameters in Tab. 4 characterize the Mohr-Coulomb criterion with tension cut-off and compression cap (cf. Fig. 11), the next two parameters denote the pure mode I and mode II fractures energies, while the last one governs the growth of the fracture energy with increasing compressive stress. Although the pure mode I fracture energy is larger than the one measured in [45] for granite, the fracture energy given in our model is characteristic to the meso-scale, i.e the individual interfaces. Experiments on the other hand measure the fracture energy at the sample-scale, which therefore corresponds to the macro-scale. Similarly to the fracture energies, the strengths f_t , c and f_c may differ from what is measured in laboratories. If quantitative results are desired, the meso-scale parameters must be obtained by calibration.

A test on another microstructure with the same loading conditions show that even a 150×150 mesh is sufficient to capture the main crack path in the studied example.

f_t (MPa)	c (MPa)	f_c (MPa)	ϕ ($^\circ$)	ψ ($^\circ$)	G_I (N mm $^{-1}$)	G_{II} (N mm $^{-1}$)	a (mm)
6.36	36	500	30	60	1	10	1

Table 4: Cohesive properties of the Brazilian test specimen

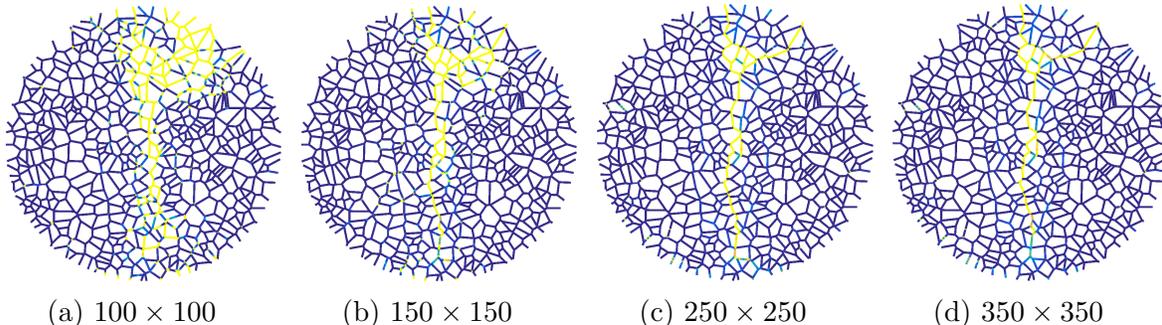


Figure 16: Relative dissipated energy on the interfaces for different mesh sizes

We now investigate how the effect of the heterogeneity influences the preferential crack paths. For the sake of illustration, this example will be simulated with properties of the phases similar to the Lac du Bonnet granite, the material properties of which are taken from reference [29]. The three main mineral species of this type of granite are biotite, quartz and feldspar. The assumed Young moduli and Poisson's ratios are $E_{\text{biotite}} = 40$ GPa, $E_{\text{quartz}} = 100$ GPa, $E_{\text{feldspar}} = 80$ GPa, $\nu_{\text{biotite}} = 0.36$, $\nu_{\text{quartz}} = 0.07$ and $\nu_{\text{feldspar}} = 0.32$. Volume fractions of 0.07, 0.2, 0.73 are assumed for each phase. The constituents were associated to grains such that the same minerals form clusters.

We are interested in the evolution of the microcracking patterns, in the final crack pattern and in the load-displacement response of the samples. With the elastic heterogeneity itself, no significant difference in the solution was found with respect to the elastically homogeneous sample. Therefore, higher tensile strengths and fracture energy values were set for those interfaces that are located between two quartz grains. The combined effect of these two kinds of heterogeneities is expected to influence the crack path. Indeed, cracks will develop around the hard phase, i.e. the quartz grains. In the simulation, the tensile strength and the cohesion of these strengthened interfaces were set to 1.25 times the values given in Tab. 4, while the mode I and mode II fracture energies were multiplied by 10.

As an example, the grain realization in Fig. 15 is considered. The applied load at each step is computed from the load factor. The vertical displacement in the top center, slightly below the loaded segment, is also recorded. These two quantities are plotted in Fig. 18. Three characteristic points on the load-displacement curve are labeled. The corresponding local relative energy dissipations G/G_c , according to the colormap in Fig. 17, are depicted in Fig. 19. The snapshots in Figs. 19a–19c correspond to the red dots displayed in Fig. 18. Looking at the structure in Fig. 19d, and comparing it with the grain configuration shown in Fig. 15, the following observation can be made. As expected, the main crack bypasses the hard phase and tends to go along the boundary of the quartz clusters. As Fig. 19a shows, the crack



Figure 17: Colormap to represent the relative energy dissipation on the interfaces. The zero value corresponds to an undamaged, the value one corresponds to a completely damaged location.

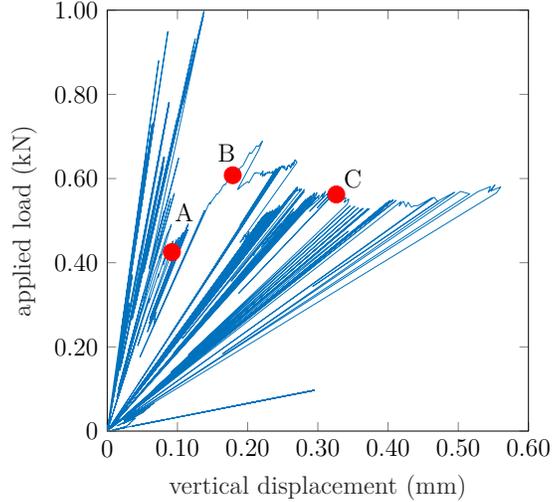


Figure 18: Load-displacement response for the configuration in Fig. 15.

initiates at the center. Then the crack extends towards the perimeter of the disk (see Fig. 19b). Note that the grain boundaries within the quartz clusters negligibly dissipate energy. The energy dissipation becomes more prominent and a secondary crack appears at the top right (see Fig. 19c).

7 Conclusions

A methodology to build a stable Lagrange multiplier space was successfully applied to solve linear elasticity problems on an assembly of subdomains (grains) discretized using quadrilateral bilinear elements in a X-FEM/GFEM setting. The subdomains are initially disconnected and the inter-grain continuity is enforced with Lagrange multipliers distributed at the nodes of the cut elements. The numerical stability tests show that the proposed discretization is stable independently of the mesh size, and the numerical illustrations indicate that the developed method is robust and mesh independent. The methodology was here extended to bilinear quadrilateral elements with arbitrary configurations of discontinuities in an element.

The elasticity equations were further coupled with a cohesive zone model using a damage formulation. The constitutive formulation is based on an extrinsic cohesive zone involving a coupled criterion incorporating a tension failure part and a compression-shear part with a compression-dependent fracture energy. The addition of the treatment of contact in the cohesive zone and the mesh-independent discretization allow the simulations of problems of practical interest. The explicit

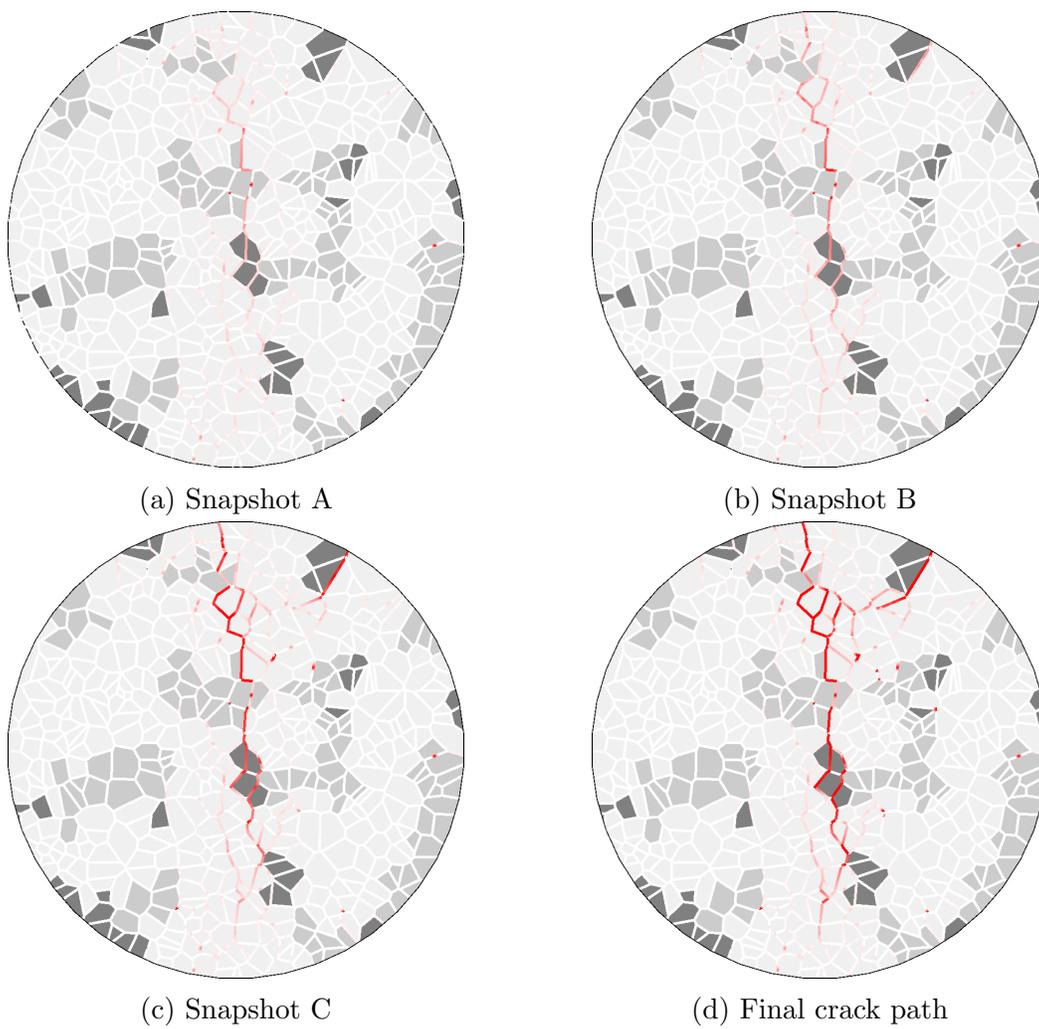


Figure 19: Progressive crack propagation in the configuration corresponding to Fig. 15

damage update algorithm enables the efficient solution of complex problems such as the Brazilian test-inspired configuration.

Extension of the stable Lagrange multiplier method to 3D should be straightforward as most of the algorithms are dimension independent (discretized equations, Alg. 1, etc.).

Acknowledgement

The first author wishes to express thanks to Benoît Lé, who thoroughly explained the original version of the explicit damage update algorithm and answered many related questions. The work of the first author was supported by the Erasmus Mundus Joint Doctorate SEED project (European Commission, 2013-1436/001-001-EMJD).

Appendix

A Precomputing matrices and vectors

Algorithm 3 gives the procedure to precompute the global matrices that do not change with propagating damage. One can see that both the local non-conforming mesh generation and the local assembly are completely independent for each subdomain and interface, allowing straightforward parallelization.

Algorithm 3 Precomputing global matrices and vectors

```
1: ▷ Preprocessing
2: Input: grain configuration, elastic properties of grains, cohesive properties of
   interfaces, boundary conditions
3: Generate a uniform Cartesian mesh covering all the grains
4: ▷ Construct the subdomain stiffness matrices
5: for  $i \in I_s$  do
6:   Create the background mesh  $\mathcal{M}^i$  for the subdomain  $\Omega^i$ 
7:   Find and store  $\emptyset \neq \Omega_e^i = \Omega^i \cap e \in \mathcal{M}^i$ 
8:   for  $e \in \mathcal{M}^i$  do
9:      $\mathbf{K}_e^i \leftarrow (\boldsymbol{\psi}, \boldsymbol{\psi})_{\Gamma_e^i}$  (44)
10:     $\mathbf{K}^i \leftarrow \text{element-to-local}(\mathbf{K}_e^i)$ 
11:   end for
12:    $\mathbf{K} \leftarrow \text{local-to-global}(\mathbf{K}^i)$ 
13: end for
14: ▷ Construct the matrices defined on the interfaces
15: for  $i \in I_{\text{int}} \cup I_D$  do
16:   Create the background mesh  $\mathcal{M}_\Gamma^i$  for the interface  $\Gamma^i$ 
17:   Find and store  $\emptyset \neq \Gamma_e^i = \Gamma^i \cap e \in \mathcal{M}_\Gamma^i$ 
18:   for  $e \in \mathcal{M}_\Gamma^i$  do
19:      $\mathbf{T}_e^i \leftarrow (\boldsymbol{\psi}, \boldsymbol{\psi})_{\Gamma_e^i}$ 
20:      $\mathbf{P}_e^i \leftarrow R_P^i \circ \mathbf{T}_e^i$            ▷  $R_P^i$ : reduction operator to get (89)
21:      $\mathbf{B}_e^i \leftarrow R_B^i \circ \mathbf{T}_e^i$        ▷  $R_B^i$ : reduction operator to get (45)
22:      $\mathbf{f}_e^i \leftarrow R_f^i \circ \mathbf{T}_e^i$      ▷  $R_f^i$ : reduction operator to get (46)
23:      $\mathbf{g}_e^i \leftarrow R_g^i \circ \mathbf{T}_e^i$      ▷  $R_g^i$ : reduction operator to get (47)
24:   end for
25:    $\mathbf{P} \leftarrow \text{local-to-global}(\mathbf{P}^i)$ 
26:    $\mathbf{B} \leftarrow \text{local-to-global}(\mathbf{B}^i)$ 
27:    $\mathbf{f} \leftarrow \text{local-to-global}(\mathbf{f}^i)$ 
28:    $\mathbf{g} \leftarrow \text{local-to-global}(\mathbf{g}^i)$ 
29: end for
30: return  $\mathbf{K}, \mathbf{P}, \mathbf{B}, \mathbf{f}, \mathbf{g}$ 
```

B Damage update

Algorithm 4 Damage update

```

1: Inputs:  $\zeta_i^{(n)}$ ,  $d_i^{(n-1)}$   $\forall i \in \mathcal{I}$ , numerical parameters:  $\Delta\tilde{G}_{\max}$  (123),  $f_1$  (124)
2: for  $i \in \mathcal{I}$  do
3:   Determine  $\zeta_{n,i}^{(n)}$  and  $\zeta_{t,i}^{(n)}$  from  $\zeta_i^{(n)}$  using a change of basis
4:   Calculate the damage indicators  $\chi_i^{(n)}$  according to (80)
5:   Determine the cohesive tractions  $t_{n,i}^{(n)}$  and  $t_{t,i}^{(n)}$  from (81)
6:   if  $G_{n,i}$  exists then  $\triangleright p_i, q_i, G_{n,i}, G_{t,i}$  already fixed
7:     Use  $p_i, q_i, G_{n,i}, G_{t,i}$ 
8:   else
9:     Determine  $p_i, q_i$  from  $t_{n,i}^{(n)}, t_{t,i}^{(n)}$  and Fig. 11
10:    Determine  $G_{n,i}$  and  $G_{t,i}$  from Tab. 1
11:   end if
12:   Compute  $y_i^{(n)}, y_{c,i}^{(n)}, y'_{c,i}^{(n)}$  from Tab. 1
13: end for
14: Find locations  $j \in \mathcal{J} \subseteq \mathcal{I}$  which are prone to further damage, i.e.  $d_j^{(n-1)} < 1$ 
15: Compute the load factor  $\mu^{(n)}$  from (119)
16: Compute  $f_i^{(i)} = \mu^{(n)^2} y_i^{(n)} - y_{c,i}^{(n)}$  and  $\alpha_i^{(n-1)}, \beta_i^{(n-1)}$  from (116)  $\forall i \in \mathcal{I}$ 
17: Calculate the maximum energy dissipation  $\Delta G_{\max}^{(n)}$  from (123)
18: Compute the load factor increment  $\text{pred}\Delta\mu^{(n)}$  from (121) with  $j \in \mathcal{J}$ 
19: Compute the energy dissipation increments  $\Delta G_i^{(n)}$  from (122)
20: for  $i \in \mathcal{I}$  do
21:   if (126) then
22:      $\Delta G_i^{(n)} \leftarrow 0$   $\triangleright$  Exclude excessively small energy dissipation increments
23:   end if
24:   if  $\Delta G_i^{(n)} > 0$  AND  $\Delta G_i^{(n-1)} = 0$  then  $\triangleright$  New initiation location
25:     Store  $p_i, q_i, G_{n,i}, G_{t,i}, \Delta G_i^{(n)}$ 
26:   end if
27:   Compute  $\Delta d_i^{(n)}$  from (127)
28:   Determine  $d_i^{(n)}$  from (128)
29: end for
30: return  $d_i^{(n)}$   $\forall i \in \mathcal{I}$ 

```

References

- [1] Arnold, D. N. and Rognes, M. E. [Stability of Lagrange elements for the mixed Laplacian](#). *Calcolo*, 46(4):245–260, Aug. 2009.
- [2] Bathe, K.-J. [The inf-sup condition and its evaluation for mixed finite element methods](#). *Computers & Structures*, 79(2):243–252, 2001.
- [3] Bažant, Z. P. [Concrete fracture models: testing and practice](#). *Eng. Fract. Mech.*, 69(2):165–205, Jan. 2002.

- [4] Béchet, E., Moës, N., and Wohlmuth, B. [A stable Lagrange multiplier space for stiff interface conditions within the extended finite element method.](#) *Int. J. Numer. Methods Eng.*, 78(8):931–954, 2009.
- [5] Belytschko, T. and Areias, P. M. A. [A comment on the article “A finite element method for simulation of strong and weak discontinuities in solid mechanics” by A. Hansbo and P. Hansbo \[Comput. Methods Appl. Mech. Engrg. 193 \(2004\) 3523–3540\].](#) *Comput. Methods Appl. Mech. Eng.*, 195(9–12):1275–1276, 2006.
- [6] Benzi, M., Golub, G. H., and Liesen, J. [Numerical solution of saddle point problems.](#) *Acta Numerica*, 14:1–137, 2005.
- [7] Berljafa, M., Elsworth, S., and Güttel, S. [A Rational Krylov Toolbox for MATLAB.](#) MIMS EPrint 2014.56, Manchester Institute for Mathematical Sciences, The University of Manchester, UK, 2014.
- [8] Bernard, P. E., Moës, N., and Chevaugeon, N. [Damage growth modeling using the Thick Level Set \(TLS\) approach: Efficient discretization for quasi-static loadings.](#) *Comput. Methods Appl. Mech. Eng.*, 233:11–27, 2012.
- [9] Boffi, D., Brezzi, F., and Fortin, M. *Mixed Finite Element Methods and Applications*, volume 44 of *Springer Series in Computational Mathematics*. Springer Berlin Heidelberg, First edition, 2013. ISBN 978-3-642-36518-8.
- [10] Brezzi, F. [On the existence, uniqueness and approximation of saddle-point problems arising from Lagrangian multipliers.](#) *R.A.I.R.O. Analyse Numérique*, 8(R2):129–151, 1974.
- [11] Burman, E. and Hansbo, P. [Fictitious domain finite element methods using cut elements: I. A stabilized Lagrange multiplier method.](#) *Comput. Methods Appl. Mech. Eng.*, 199(41–44):2680–2686, 2010.
- [12] Burman, E., Claus, S., Hansbo, P., Larson, M. G., and Massing, A. [CutFEM: Discretizing geometry and partial differential equations.](#) *Int. J. Numer. Methods Eng.*, 104(7):472–501, 2015.
- [13] Cazes, F., Coret, M., and Combescure, A. [A two-field modified Lagrangian formulation for robust simulations of extrinsic cohesive zone models.](#) *Comput. Mech.*, 51(6):865–884, Aug. 2012.
- [14] Chapelle, D. and Bathe, K. J. [The inf-sup test.](#) *Computers & Structures*, 47(4):537–545, 1993.
- [15] Daux, C., Moës, N., Dolbow, J., Sukumar, N., and Belytschko, T. [Arbitrary branched and intersecting cracks with the extended finite element method.](#) *Int. J. Numer. Methods Eng.*, 48(12):1741–1760, 2000.
- [16] Ferté, G., Massin, P., and Moës, N. [Interface problems with quadratic X-FEM: design of a stable multiplier space and error analysis.](#) *Int. J. Numer. Methods Eng.*, 100(11):834–870, 2014.

- [17] Ferté, G., Massin, P., and Moës, N. [3D crack propagation with cohesive elements in the extended finite element method](#). *Comput. Methods Appl. Mech. Eng.*, 300: 347–374, 2016.
- [18] Geniaut, S., Massin, P., and Moës, N. [A stable 3D contact formulation using X-FEM](#). *European Journal of Computational Mechanics*, 16(2):259–275, 2007.
- [19] Hansbo, A. and Hansbo, P. [An unfitted finite element method, based on Nitsche’s method, for elliptic interface problems](#). *Comput. Methods Appl. Mech. Eng.*, 191(47-48):5537–5552, Nov. 2002.
- [20] Hautefeuille, M., Annavarapu, C., and Dolbow, J. E. [Robust imposition of Dirichlet boundary conditions on embedded surfaces](#). *Int. J. Numer. Methods Eng.*, 90(1):40–64, 2012.
- [21] Hoover, C. G. and Bažant, Z. P. [Cohesive crack, size effect, crack band and work-of-fracture models compared to comprehensive concrete fracture tests](#). *Int. J. Fract.*, 187(1):133–143, Jan. 2014.
- [22] ISRM. [Suggested methods for determining tensile strength of rock materials](#). *Int. J. Rock Mech. Min. Sci.*, 15(3):99–103, June 1978.
- [23] Ji, H. and Dolbow, J. E. [On strategies for enforcing interfacial constraints and evaluating jump conditions with the extended finite element method](#). *Int. J. Numer. Methods Eng.*, 61(14):2508–2535, 2004.
- [24] Jin, Y., Pierard, O., Wyart, E., and Béchet, E. [Crack Lip Contact Modeling Based on Lagrangian Multipliers with X-FEM](#). In *SEMA SIMAI Springer Series*, volume 12, pages 123–142. Springer International Publishing, 2016. ISBN 978-3-319-41245-0.
- [25] Lé, B., Moës, N., and Legrain, G. [Coupling damage and cohesive zone models with the Thick Level Set approach to fracture](#). *Eng. Fract. Mech.*, 193:214–247, 2018.
- [26] Li, D. and Wong, L. N. Y. [The Brazilian Disc Test for Rock Mechanics Applications: Review and New Insights](#). *Rock Mech. Rock Eng.*, 46(2):269–287, May 2012.
- [27] Lourenço, P. B. [Computational strategies for masonry structures](#). phdthesis, Delft University of Technology, 1995. ISBN 90-407-1221-2.
- [28] Massart, T. J. and Selvadurai, A. P. S. [Stress-induced permeability evolution in a quasi-brittle geomaterial](#). *J. Geophys. Res. - Sol. Ea.*, 117(B7), 2012.
- [29] Massart, T. J. and Selvadurai, A. P. S. [Computational modelling of crack-induced permeability evolution in granite with dilatant cracks](#). *Int. J. Rock Mech. Min. Sci.*, 70:593–604, 2014.
- [30] Melenk, J. M. and Babuška, I. [The partition of unity finite element method: Basic theory and applications](#). *Comput. Methods Appl. Mech. Eng.*, 139(1-4): 289–314, Dec. 1996.

- [31] Moës, N., Dolbow, J., and Belytschko, T. [A finite element method for crack growth without remeshing](#). *Int. J. Numer. Methods Eng.*, 46(1):131–150, 1999.
- [32] Moës, N., Béchet, E., and Tourbier, M. [Imposing Dirichlet boundary conditions in the extended finite element method](#). *Int. J. Numer. Methods Eng.*, 67(12):1641–1669, 2006.
- [33] Molino, N., Bao, Z., and Fedkiw, R. [A virtual node algorithm for changing mesh topology during simulation](#). *ACM Trans. Graphics*, 23(3):385, Aug. 2004.
- [34] Mourad, H. M., Dolbow, J., and Harari, I. [A bubble-stabilized finite element method for Dirichlet constraints on embedded interfaces](#). *Int. J. Numer. Methods Eng.*, 69(4):772–793, 2007.
- [35] Nitsche, J. [Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind](#). *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, 36(1):9–15, 1971.
- [36] Olshanskii, M. A., Reusken, A., and Grande, J. [A Finite Element Method for Elliptic Equations on Surfaces](#). *SIAM J. Numer. Anal.*, 47(5):3339–3358, Jan. 2009.
- [37] Ottosen, N. S. and Ristinmaa, M. [The Mechanics of Constitutive Modeling](#). Elsevier Science, 2005. ISBN 9780080446066.
- [38] Puso, M. A. and Laursen, T. A. [Mesh tying on curved interfaces in 3D](#). *Eng. Computation*, 20(3):305–319, 2003.
- [39] Sanders, J. D., Dolbow, J., and Laursen, T. A. [On methods for stabilizing constraints over enriched interfaces in elasticity](#). *Int. J. Numer. Methods Eng.*, 78(9):1009–1036, 2008.
- [40] Schellekens, J. C. J. [Computational Strategies for Composite Structures](#). phdthesis, Delft University of Technology, 1992.
- [41] Shabir, Z., van der Giessen, E., Duarte, C. A., and Simone, A. [The role of cohesive properties on intergranular crack propagation in brittle polycrystals](#). *Modell. Simul. Mater. Sci. Eng.*, 19(3), Mar. 2011.
- [42] Simone, A., Duarte, C. A., and van der Giessen, E. [A Generalized Finite Element Method for polycrystals with discontinuous grain boundaries](#). *Int. J. Numer. Methods Eng.*, 67(8):1122–1145, 2006.
- [43] Sukumar, N., Dolbow, J. E., and Moës, N. [Extended finite element method in computational fracture mechanics: a retrospective examination](#). *Int. J. Fract.*, 196(1):189–206, 2015.
- [44] van Zijl, G. [Computational Modelling of Masonry Creep and Shrinkage](#). phdthesis, Delft University of Technology, 2000.

- [45] Vasconcelos, G., Lourenço, P. B., Alves, C. A. S., and Pamplona, J. [Experimental characterization of the tensile behaviour of granites](#). *Int. J. Rock Mech. Min. Sci.*, 45(2):268–277, Feb. 2008.
- [46] Wall, W. A. and Gerstenberger, A. [An embedded Dirichlet formulation for 3D continua](#). *Int. J. Numer. Methods Eng.*, 82(5):537–563, 2010.
- [47] Wittmann, F. H., Rokugo, K., Brühwiler, E., Mihashi, H., and Simonin, P. [Fracture energy and strain softening of concrete as determined by means of compact tension specimens](#). *Mater. Struct.*, 21(1):21–32, Jan. 1988.