A NON-INTRUSIVE MESH PARTITIONING ALGORITHM FOR CONFORMING FINITE ELEMENT SIMULATIONS ON FRACTURED DOMAINS

F. FEPPON[1], N. PAUWELS[1], C. DAPOGNY[2], M. AVERSENG[3]

[1] *NUMA Unit, Department of Computer Science, KU Leuven, Belgium.*
[2] *Sorbonne Université, Université Paris Cité, CNRS, Inria, Laboratoire Jacques-Louis Lions, LJLL, F-75005 Paris, France.*
[3] *Laboratoire Angevin de Recherche Mathématique, Université d'Angers, 49000 Angers, France*

Abstract. This article introduces a novel strategy for assembling the finite element matrices of boundary value problems posed in arbitrary 2d and 3d fractured domains with existing third-party solvers, used in a non intrusive manner. For numerical accuracy, the fractured geometry is represented by a conforming mesh; the discontinuous nature of the solution raises the need to create copies of the degrees of freedom of the finite element space according to the topology of the fracture: duplication across regular regions, triplication at Y junctions... This task is difficult to carry out in a general way which is compatible with the non intrusive use of an external finite element solver, as "degenerate" configurations associated to physically coincident, yet topologically disjoint nodes are often incompatible with standard softwares. To overcome this issue, our method leverages a constrained graph partitioning algorithm that constructs a minimal partition of the fractured geometry, into a set of non overlapping "regular" subdomains. Standard finite element kernels can operate independently on each region, and the resulting local contributions are glued into a global stiffness matrix thanks to a suitable label mapping attached to the partition. We demonstrate the robustness of the method through several 2d and 3d numerical examples featuring intricate crack patterns and multi-screen geometries. The proposed framework is implemented in the open-source library `PyMedit`.

Keywords. Fractured media, Finite Element Method, conforming meshes, domain partitioning, graph partitioning, cannot-link constraints.

AMS Subject classifications. 65N30, 65N50, 05C70

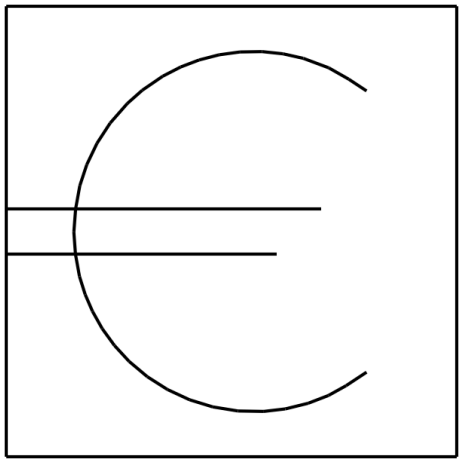
Contents

---

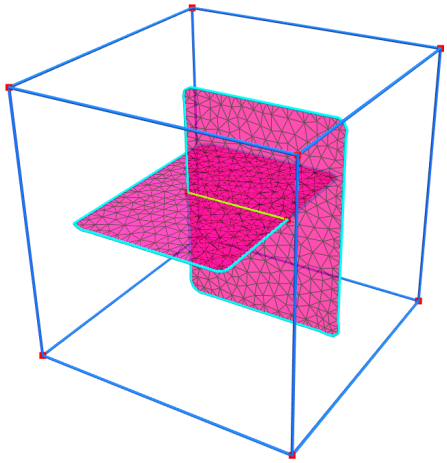
* Corresponding author. Email: `florian.feppon@kuleuven.be`. .

1. Introduction

The numerical simulation of physical phenomena accounted for by Partial Differential Equations (PDE) is a central operation in engineering, and in science in general. The Finite Element Method (FEM) [30, 47] is by now a method of choice to realize this task; although it is well-established to treat situations with regular geometries, a significant number of applications take place on domains presenting internal interfaces, cracks, or multi-screens, as exemplified in Figure 1. The simulation of such scenarios often turns out to be challenging because the physical solution field typically exhibits discontinuities across these singular features. For instance, such situations occur in the modelling of fluid flows in fractured porous media [2, 14, 63, 77], in fracture mechanics [19, 54, 79, 82], or when considering wave propagation and scattering by thin obstacles [22, 23, 27, 31, 32, 61].



(A) A 2d domain containing a €-shaped fracture, presenting multiple branches and intersecting the computational domain boundary.



(B) A 3d domain enclosing a multi-screen fracture

FIGURE 1. Examples of 2d and 3d fractured domains.

Spurred by its ubiquity in applied sciences and industry, the present work is focused on the deployment of the Finite Element paradigm to treat such fractured scenarios. Let us nevertheless mention that alternative numerical methods have been developed in this context, see [1] about the use of Discrete Elements to model the physics near the crack as a granular medium, or [18, 76, 83, 91, 100, 102] about the use of integral equations and the Boundary Element Method.

Multiple variants of the "classical" Finite Element Method have been developed to handle cracks or discontinuous interfaces, that can be classified into two categories. On the one hand, non-conforming methods alleviate the need to mesh explicitly the (often complex) fracture set and rely on a fixed mesh of the computational domain. Among them, the eXtended Finite Element Method (XFEM) [82] is particularly popular. According to the latter, the fracture set is represented implicitly, with the help of "level set" functions; drawing inspiration from the concept of partition of unity [58, 79], the finite element space is enriched with additional functions that capture the discontinuous behavior of the solution along the fracture. Despite its appealing nature, the robust implementation of XFEM for complex 3d topologies, featuring branching or coalescence, is a challenging task: while arbitrary fracture geometries have been addressed in 2d [40, 93, 104], recent 3d efforts are aimed at additional modeling issues, related to so-called cohesive zones [65]. Alternative non-conforming approaches for the numerical simulation of physical PDE featuring cracked geometries use Discontinuous-Galerkin finite elements, where fractures are allowed to exist at quadrature points [52, 92], and phase field methods, which model the effect of the fracture by a diffuse, artificial source term [28, 80, 84, 98].

On the other hand, conforming methods handle fractures by using a computational mesh that contains an explicit discretization of their geometry. This allows to directly impose attached physical boundary conditions, among other advantages. Unfortunately, in general, the construction of such a conforming mesh is difficult, especially when the fracture set shows intricate non-manifold patterns (multiple branches, etc.). In spite of the recent development of remeshing tools for the creation of high-quality meshes conforming to an input, implicitly-defined geometry [9, 38, 81, 99], a systematic algorithm for handling arbitrary 3d crack patterns is still missing. Moreover, in this setting, standard numerical solution techniques require separating physically the mesh entities along the crack, that is, to replicate the physical nodes situated on the fracture into different copies depending on the expected discontinuous nature of the solution in there [46, 66, 70, 72, 90, 95]. To the best of our knowledge, the device of a systematic method to determine the necessary degree of node replication (e.g. duplication at regular regions of the fracture, triplication at a Y-junction) has not yet been investigated. Alternative methods

include the simple element deletion method [89], and the use of zero-thickness interface elements (often coupled with cohesive zone models) [74, 87, 88, 107, 108]. We refer to [50, 51] for a comprehensive comparison of the aforementioned conforming and nonconforming techniques in the physical context of fractured porous media.

The present article is devoted to the numerical simulation of PDE in media containing fractures. We introduce a general, robust and efficient framework based on a conforming mesh of the fractured geometry at play, which conveniently allows for the non-intrusive use of a third-party finite element software to assemble and solve finite element systems: such software can be called in a black-box fashion, without any knowledge of its algorithmic and implementation details. Our strategy hinges on the recent concept of *fractured mesh* [7]: degrees of freedom that are physically coincident but topologically distinct are identified as *generalized vertices*, and this information underpins a partitioning algorithm that splits the computational domain into several "simple" pieces where the assembly of the finite element matrix of the problem under scrutiny can be executed by completely standard techniques.

At first glance, such a strategy could be implemented by mere "mesh-cutting", i.e. by replicating the nodes along the fracture into several copies and re-indexing the relations between nodes and elements according to the information contained in the generalized vertex structure. Unfortunately, this approach does not easily lend itself to interfacing with standard third-party libraries. Indeed, most assembly kernels, such as that of the finite element software `FreeFem` used in this work [59, 60], rely on geometric predicates and spatial data structures, like quadtrees or octrees, which inherently assume a one-to-one mapping between physical coordinates and topological entities: duplicating vertices at identical locations without a proper, intrusive treatment of these entities introduces topological degeneracies—such as zero-measure elements or ambiguous adjacency tables—that trigger fatal errors in geometric search routines and interpolation operators.

To alleviate this difficulty, we introduce a novel assembly algorithm of the finite element matrices attached to arbitrary fractured domains, using any standard software designed for "classical" non-fractured domains, in a non-intrusive manner. The cornerstone of our method is the computation of a *minimal partitioning* of the fractured domain, as the reunion of non-overlapping, open and Lipschitz subdomains adjacent to complementary neighborhoods of the fracture. Duplicated nodes are not handled directly, and standard assembly kernels can be used on each subdomain. Our algorithm provides the necessary mapping to correctly assemble these local contributions into the global finite element matrix of the problem at play, ensuring proper handling of the multiple degrees of freedom attached to each generalized vertex. This partitioning strategy is very general, and it allows to treat a wide range of physical problems (electrostatics, acoustics... in complex fractured domains, in two and three space dimensions.

Related ideas to those presented in this work have emerged with the development of Physics-Informed Neural Networks (PINNs) for fractured domains: in [43, 67, 73], different networks are used to approximate the solution in subdomains adjacent to a crack, with a possible enrichment of their approximation capabilities with near-tip asymptotic solutions [57]. In a different spirit, [109] avoids domain partitioning thanks to a discontinuous embedding of the input space of the network. Although they are built similarly on the fundamental idea that partitioning the domain enables to address duplicated degrees of freedom with standard methods working on non-fractured subdomains, these works, still limited to 2d applications, assume that the domain partitioning is given beforehand. In contrast, our method automatically computes such domain partitioning for arbitrary d-dimensional geometries.

The remainder of this article is organized as follows. Section 2 summarizes the necessary definitions and background on "classical" and "fractured" meshes, and generalized degrees of freedom. In Section 3, we show that if a suitable partition of the mesh is given, the finite element matrix of a variational problem posed on the fractured mesh can be obtained from the local matrices associated to the same problems on non-cracked subdomains—which can be assembled with any third-party software—and domain decomposition restriction operators. This motivates the introduction, in Section 4, of an algorithm that constructs such suitable partition from any simplicial mesh with internal faces marked as those belonging to the crack. We reformulate this domain decomposition task as a minimum graph partitioning problem with cannot-link constraints, and we describe a greedy algorithm dedicated to its solution. In Section 5, we present a method that generates a fractured mesh of a domain from the input of a "level" set description of the fracture, implemented in the open-source remeshing library Mmg [9, 12, 38, 81]. Section 6 summarizes the overall Finite Element assembly and solution procedure and provides details about the chosen implementation. Finally, Section 7 presents some illustrations of numerical solutions and numerical validations of the effectiveness of our method to solve finite element problems on singular 2d and 3d geometries.

Reproducibility. The partition strategy presented in this article is implemented in the open-source library `PyMedit` [48]. The source code associated to the examples of Section 7 is available at the following address:

`https://gitlab.com/florian.feppon/pymedit/-/tree/master/pymedit/examples/fractured_meshes`.

2. Fractured mesh connectivity: generalized simplices

This section briefly recalls basic facts about meshes and it introduces the notion of "fractured" mesh used throughout the article, as an adaptation of the concept of "generalized mesh" formalized in [7]. Intuitively, a fractured mesh is a "classical" simplicial and conforming mesh $\mathcal{M}$ of a domain Ω, in which a special set of facets, accounting for the fracture Γ are identified with a special label. The concept of generalized sub-simplices is used to identify the identities (vertices, faces, etc.) that must be replicated according to the topology of the fracture.

2.1. "Classical" meshes and their properties

Let us first recall the definitions of simplices, sub-simplices, triangulations and meshes in the Euclidean space $\mathbb{R}^d$ ($d = 2, 3$) and a few well-known properties, in the form which is useful for our purpose; we closely follow the exposition of [7].

Definition 2.1 (Simplex and sub-simplices of a simplex)**.**

- An *n-simplex* S is a subset $\{V_1, \dots, V_{n+1}\} \subset \mathbb{R}^d$ of $(n+1)$ elements of $\mathbb{R}^d$, called the vertices of S. For $n = 0, 1, 2$ and 3, an n-simplex is called a vertex, an edge, a triangle and a tetrahedron, respectively.
- The subsets of an n-simplex S with cardinal $(m+1)$, $0 \leq m \leq n$ are called the *m-sub-simplices* of S.
- The closed convex hull of a simplex $S = \{V_1, \dots, V_{n+1}\}$ is denoted by $|S|$:

$$|S| = \left\{ \sum_{i=1}^{n+1} \lambda_i V_i \;\middle|\; 0 \leq \lambda_i \leq 1 \text{ for } 1 \leq i \leq n+1 \text{ and } \sum_{i=1}^{n+1} \lambda_i = 1 \right\}.$$

- An n-simplex S is *non-degenerate* if $|S|$ contains an open ball of dimension n with positive radius.

In the sequel, with a small abuse of notation, we indifferently use the names vertex, edge, triangle, tetrahedron, element and simplex for the defining set of vertices S and the induced closed convex hull $|S|$.

Definition 2.2 (Triangulation)**.**

- A *d-dimensional triangulation* $\mathcal{M}$ in $\mathbb{R}^d$ is a finite set of d-simplices K, called the *elements* of $\mathcal{M}$. By definition, the notation $K \in \mathcal{M}$ indicates that K is an element of $\mathcal{M}$.
- A *sub-simplex* S of $\mathcal{M}$ is a sub-simplex of an element $K \in \mathcal{M}$. The $(d-1)$ sub-simplices of an element $K \in \mathcal{M}$ (resp. of $\mathcal{M}$) are called the *faces* of K (resp. of $\mathcal{M}$).
- For $0 \leq m \leq d$, the collection of all m-sub-simplices of $\mathcal{M}$ is denoted by $\sigma_m(\mathcal{M})$: $\sigma_0(\mathcal{M})$ is the set of vertices of $\mathcal{M}$, $\sigma_1(\mathcal{M})$ is the set of edges, $\dots$, $\sigma_d(\mathcal{M})$ is the set of elements. The set of all the sub-simplices of $\mathcal{M}$ is:

$$\sigma(\mathcal{M}) := \bigcup_{0 \leq m \leq d} \sigma_m(\mathcal{M}).$$

We now arrive at the definition of mesh, which is illustrated on Figure 2a.

Definition 2.3 (Simplicial mesh)**.** A *simplicial* (or *conforming*) *mesh* $\mathcal{M}$ is a d-dimensional triangulation made of non-degenerate elements such that:

$$\forall K, K' \in \mathcal{M}, \quad |K| \cap |K'| = |K \cap K'|;$$

in particular, two different elements in $\mathcal{M}$ have disjoint interiors and their intersection is a sub-simplex of $\mathcal{M}$. The closed "volumic" set spanned by $\mathcal{M}$ is denoted by $|\mathcal{M}|$:

$$|\mathcal{M}| := \bigcup_{K \in \mathcal{M}} |K|.$$

Unless stated otherwise, all meshes considered in this article are assumed to be conforming. For convenience, we recall the proof of the following well-known facts about simplicial meshes, see also [94, p. 22], [15, Lemma 11.1.2].

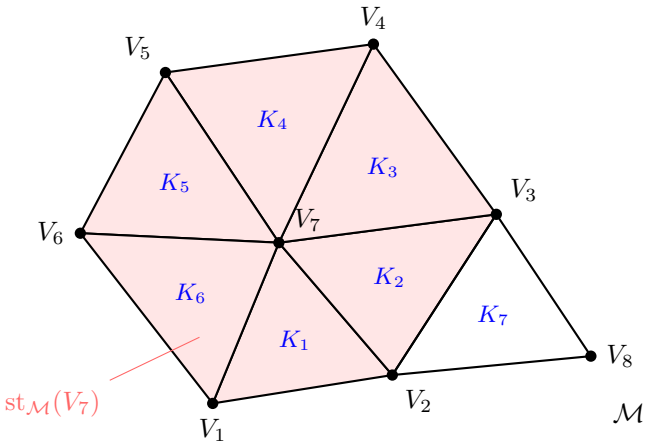
Proposition 2.1 (Boundary of a mesh)**.** *Let $\mathcal{M}$ be a simplicial mesh in $\mathbb{R}^d$; then:*

(i) Each $(d-1)$ sub-simplex of $\mathcal{M}$ is included in one or two elements of $\mathcal{M}$. The subset of those $(d-1)$ sub-simplices of $\mathcal{M}$ that are included in exactly one element is called the boundary mesh *of $\mathcal{M}$.*

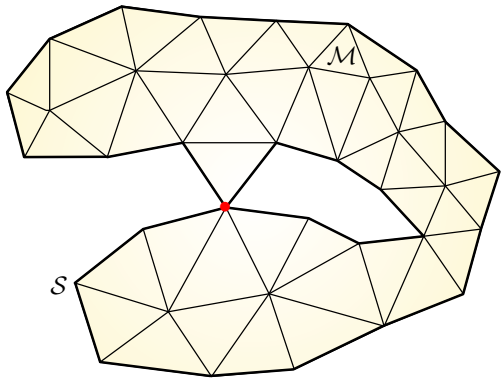
(ii) The closed set $|\mathcal{M}|$ coincides with the closure of its interior: $|\mathcal{M}| = \overline{\text{int}(|\mathcal{M}|)}$.

(iii) The set formed by the faces of the boundary mesh of $\mathcal{M}$ coincides with the topological boundary of $|\mathcal{M}|$.

Proof. (i). We argue by contradiction. Assume that there exists a $(d-1)$ sub-simplex $S = \{V_1, \dots, V_d\}$ of $\mathcal{M}$ which is included in at least three different elements $K_i = \{V_1, \dots, V_d, W_i\}$, $i = 1, 2, 3$. Since K_1 is non degenerate, the vertex W_1 lies in one of the two strict half-spaces delimited by the affine hyperplane $H := \text{span}\{V_1, \dots, V_d\}$. Since $\mathcal{M}$ is conforming, K_1 and K_2 have disjoint interiors, and so W_2 belongs to the other strict half-space delimited by H. Eventually, since the interior of K_3 is disjoint from those of K_1 and K_2,



(A) A 2d simplicial mesh $\mathcal{M}$ with vertices $\{V_i\}_{1\leq i\leq 8}$ and elements $\{K_i\}_{1\leq i\leq 7}$. The star $\mathrm{st}_{\mathcal{M}}(V_7)$ of the vertex V_7 is made of the pink elements.



(B) A 2d mesh $\mathcal{M}$ whose boundary mesh $\mathcal{S}$ is the bold line: $\mathcal{M}$ is non regular, since the region of the red boundary vertex is not homeomorphic to a half-plane.

FIGURE 2. Illustrations of the main notions about "classical" meshes.

the vertex W_3 cannot belong to any of these half spaces, and since K_3 is non degenerate, it cannot belong to H either. This is absurd, and we have thus proved (i).

(ii). Since $|\mathcal{M}|$ is closed, the inclusion $\overline{\mathrm{int}(|\mathcal{M}|)} \subset |\mathcal{M}|$ is immediate. On the other hand, let $x \in |\mathcal{M}|$. Then x belongs to one simplex $K \in \mathcal{M}$, and since K is non degenerate, there exists a sequence $x^n \in \mathrm{int}(K)$ such that $x^n \to x$. Since the sequence x^n obviously belongs to $\mathrm{int}(|\mathcal{M}|)$, the inclusion $|\mathcal{M}| \subset \overline{\mathrm{int}(|\mathcal{M}|)}$ follows.

(iii). Let $\mathcal{S}$ denote the boundary mesh of $\mathcal{M}$; we first show that $\mathcal{S} \subset \partial|\mathcal{M}|$. Let S be a face in $\mathcal{S}$; since $\partial|\mathcal{M}|$ is closed, it is enough to show that it contains the relative interior of S, i.e. its interior within the affine hyperplane spanned by S. To achieve this, let x be a point in S which does not belong to any $(d-2)$-sub-simplex of S, and let us assume for a contradiction that x lies in the interior of $|\mathcal{M}|$. Then, there exists $r > 0$ such that the open ball $B(x, r)$ with center x and radius r in $\mathbb{R}^d$ is included in $|\mathcal{M}|$. Introducing a unit normal vector n to S, $x \pm tn$ then belong to $|\mathcal{M}|$ for $t > 0$ small enough, and since x is in the relative interior of S, S is shared by two disjoint elements of $\mathcal{M}$. This contradicts the definition of S: $x \in |\mathcal{M}|$ cannot belong to the interior of $|\mathcal{M}|$. Hence, $x \in \partial|\mathcal{M}|$, and we have proved the inclusion $\mathcal{S} \subset \partial|\mathcal{M}|$.

Conversely, we show that $\partial|\mathcal{M}| \subset \mathcal{S}$. Let $x \in \partial|\mathcal{M}|$. Since $\mathcal{S}$ is a closed set, it is enough to prove that there exists a sequence $x^n \in \mathcal{S}$ that converges to x. To achieve this, by definition of the topological boundary and because of (ii), there exist sequences $t^n \to 0$, $y^n \in \mathrm{int}(|\mathcal{M}|)$ and $z^n \in \mathbb{R}^d \backslash |\mathcal{M}|$, such that

$$|x - y^n| = t^n \text{ and } |x - z^n| = t^n.$$

Let now $\gamma^n(s)$, $s \in [0, 1]$ be a circular arc with center x and endpoints $\gamma^n(0) = y^n$, $\gamma^n(1) = z^n$. We define:

$$s^n = \inf \left\{ s \in (0,1) \;\; \gamma^n(s) \in \mathbb{R}^d \backslash |\mathcal{M}| \right\} \in (0,1), \text{ and } x^n = \gamma^n(s^n).$$

The point x^n necessarily belongs to a face S of $\mathcal{M}$, and up to making a small perturbation of the sequences y^n and z^n (which belong to the open sets $\mathrm{int}(|\mathcal{M}|)$ and $\mathbb{R}^d \backslash |\mathcal{M}|$), we may assume that it actually belongs to the relative interior of S. Hence, S belongs to only one simplex of $\mathcal{M}$ and $x^n \in \mathcal{S}$. We have thus constructed the desired sequence $x^n \in \mathcal{S}$, which converges to x since $|x - x^n| = t^n$. □

Definition 2.4. The mesh $\mathcal{M}$ is *regular* if the set $|\mathcal{M}|$ is a manifold with boundary, *i.e.*, each point $x \in |\mathcal{M}|$ has a neighborhood in $|\mathcal{M}|$ which is homeomorphic to either $\mathbb{R}^d$ or $\mathbb{R}^{d-1} \times \mathbb{R}_+$.

Remark 2.1. A mesh $\mathcal{M}$ typically fails to be regular when one of the sub-simplices of the boundary mesh $\mathcal{S}$ is shared by elements from "distant" regions of $\mathcal{S}$, as exemplified in Figure 2 (b).

Definition 2.5 (Star of a sub-simplex)**.** The *star* of a sub-simplex $S \in \sigma(\mathcal{M})$ in the mesh $\mathcal{M}$ is the collection of all d-simplices of $\mathcal{M}$ containing S:

$$\mathrm{st}_{\mathcal{M}}(S) := \{K \in \mathcal{M} \,|\, S \subset K\}.$$

The next result shows that the stars of all the sub-simplices of a mesh can be inferred from the stars of its vertices.

Lemma 2.1. *Let $S_1, S_2 \in \sigma(\mathcal{M})$ be two sub-simplices of a conforming mesh $\mathcal{M}$. Then,*

(i) Either $\mathrm{st}_{\mathcal{M}}(S_1) \cap \mathrm{st}_{\mathcal{M}}(S_2) = \emptyset$ *or* $S_1 \cup S_2 \in \sigma(\mathcal{M})$*, and in this case, it holds:*

$$\mathrm{st}_{\mathcal{M}}(S_1 \cup S_2) = \mathrm{st}_{\mathcal{M}}(S_1) \cap \mathrm{st}_{\mathcal{M}}(S_2). \tag{2.1}$$

(ii) The following inclusion holds true:

$$\mathrm{st}_{\mathcal{M}}(S_1) \cup \mathrm{st}_{\mathcal{M}}(S_2) \subset \mathrm{st}_{\mathcal{M}}(S_1 \cap S_2). \tag{2.2}$$

Proof. (i). Assuming that $\mathrm{st}_{\mathcal{M}}(S_1) \cap \mathrm{st}_{\mathcal{M}}(S_2) \neq \emptyset$, there exists an element $K \in \mathcal{M}$ such that $S_1 \subset K$ and $S_2 \subset K$. In particular, all the vertices of S_1 and S_2 belong to K, and so $S_1 \cup S_2 \in \sigma(\mathcal{M})$. Now, for any element K of $\mathcal{M}$, we have:

$$\begin{aligned} K \in \mathrm{st}_{\mathcal{M}}(S_1 \cup S_2) &\Leftrightarrow S_1 \cup S_2 \subset K \\ &\Leftrightarrow S_1 \subset K \text{ and } S_2 \subset K \\ &\Leftrightarrow K \in \mathrm{st}_{\mathcal{M}}(S_1) \cap \mathrm{st}_{\mathcal{M}}(S_2), \end{aligned},$$

which proves (i).

(ii). Any element $K \in \mathcal{M}$ that belongs to $\mathrm{st}_{\mathcal{M}}(S_1) \cup \mathrm{st}_{\mathcal{M}}(S_2)$ contains either S_1 or S_2, and in any case, it contains the intersection $S_1 \cap S_2 \subset K$. This proves that $K \in \mathrm{st}_{\mathcal{M}}(S_1 \cap S_2)$, as desired. □

We extend the above definition of star to any point $x \in |\mathcal{M}|$, that may not be a vertex of the mesh $\mathcal{M}$:

$$\forall\, x \in |\mathcal{M}|, \quad \mathrm{st}_{\mathcal{M}}(x) := \{K \in \mathcal{M} \,|\, x \in |K|\}.$$

The next lemma shows that $\mathrm{st}_{\mathcal{M}}(x)$ is the star of the smallest sub-simplex containing x.

Lemma 2.2. *For any* $x \in |\mathcal{M}|$*, it holds that* $\mathrm{st}_{\mathcal{M}}(x) = \mathrm{st}_{\mathcal{M}}(S)$ *where* S *is the sub-simplex*

$$S := \bigcap_{\substack{S' \in \sigma(\mathcal{M}) \\ x \in |S'|}} S'.$$

Proof. Let S be as in the above statement, and let K be an arbitrary element in $\mathrm{st}_{\mathcal{M}}(S)$. By definition, K contains the sub-simplex S, and so $x \in |K|$. This implies that $K \in \mathrm{st}_{\mathcal{M}}(x)$, whence the inclusion $\mathrm{st}_{\mathcal{M}}(S) \subset \mathrm{st}_{\mathcal{M}}(x)$.

Conversely, let K be an element in the star $\mathrm{st}_{\mathcal{M}}(x)$, i.e. $x \in |K|$. Then, (2.2) implies that

$$K \in \bigcup_{\substack{S' \in \sigma(\mathcal{M}) \\ x \in |S'|}} \mathrm{st}_{\mathcal{M}}(S') \subset \mathrm{st}_{\mathcal{M}}(S),$$

which proves that $\mathrm{st}_{\mathcal{M}}(x) \subset \mathrm{st}_{\mathcal{M}}(S)$. □

2.2. **Fractured meshes**

We now arrive at the notion of fractured mesh. Let $\mathcal{M}$ be a regular simplicial mesh and let $\Gamma \subset \sigma_{d-1}(\mathcal{M})$ be a "fracture", i.e. a subset of the faces of $\mathcal{M}$. The need to consider entities of Γ differently, depending on which neighboring region they are observed from, motivates the notion of *generalized sub-simplices*, which is illustrated in 2d on Figs. 2a and 3 and in 3d on Figure 4.

Definition 2.6 (Generalized sub-simplices, [7, Def. 2.11])**.** Let $S \in \sigma(\mathcal{M})$ and let $\mathcal{G}(S)$ be the graph formed by the elements of $\mathrm{st}_{\mathcal{M}}(S)$, in which $K, K' \in \mathrm{st}_{\mathcal{M}}(S)$ are connected by an edge if they share a face that does not belong to Γ, i.e. $\#(K \cap K') = d-1$ and $K \cap K' \notin \Gamma$. The n_S connected components of $\mathcal{G}(S)$ are called the *generalized sub-simplices* of S and they are denoted by

$$\mathrm{st}_{\mathcal{M}}(S;1), \ldots, \mathrm{st}_{\mathcal{M}}(S;n_S).$$

This gives rise to the following disjoint partition:

$$\mathrm{st}_{\mathcal{M}}(S) = \bigcup_{1 \leq \alpha \leq n_S} \mathrm{st}_{\mathcal{M}}(S;\alpha).$$

Having extended the notion of star to any point $x \in |\mathcal{M}|$, we likewise define the generalized points associated to x, denoted by:

$$\mathrm{st}_{\mathcal{M}}(x;1), \ldots, \mathrm{st}_{\mathcal{M}}(x;n_x).$$

Intuitively, these generalized points represent the various replications of x attached to the different regions delimited by the fracture Γ.

Notation. Throughout the rest of the article, $\mathcal{M}$ stands for a regular simplicial mesh in $\mathbb{R}^d$, and we denote by Ω the bounded Lipschitz domain which is the interior of $|\mathcal{M}|$. We consider a fracture $\Gamma \subset \sigma_{d-1}(\mathcal{M})$ inside Ω, and with a slight abuse of notation, we indifferently denote by $\Gamma \equiv |\Gamma|$ the fracture and its discretization as a collection of $(d-1)$-subsimplices of $\mathcal{M}$.

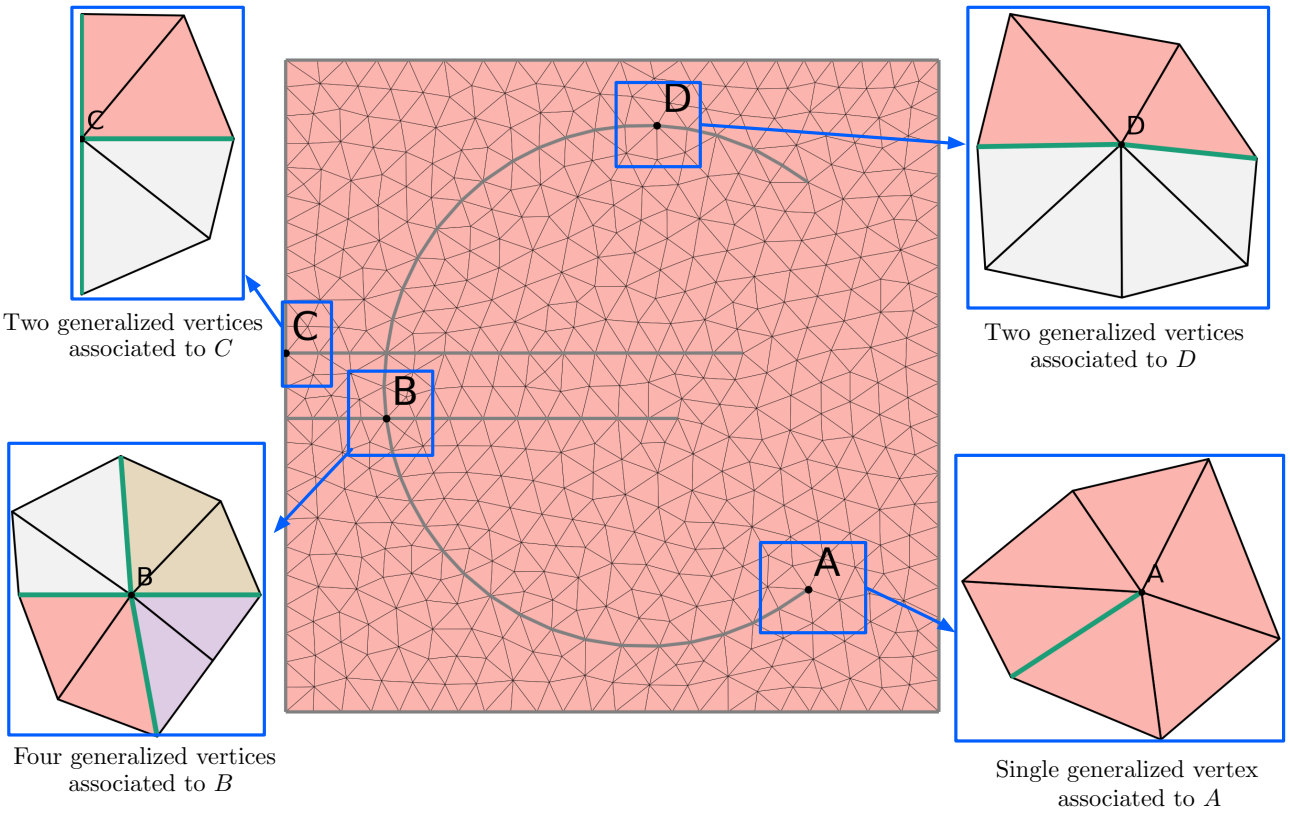


FIGURE 3. Mesh of the fractured €-shaped 2d domain with four particular vertices A, B, C, D at different locations on the fracture.

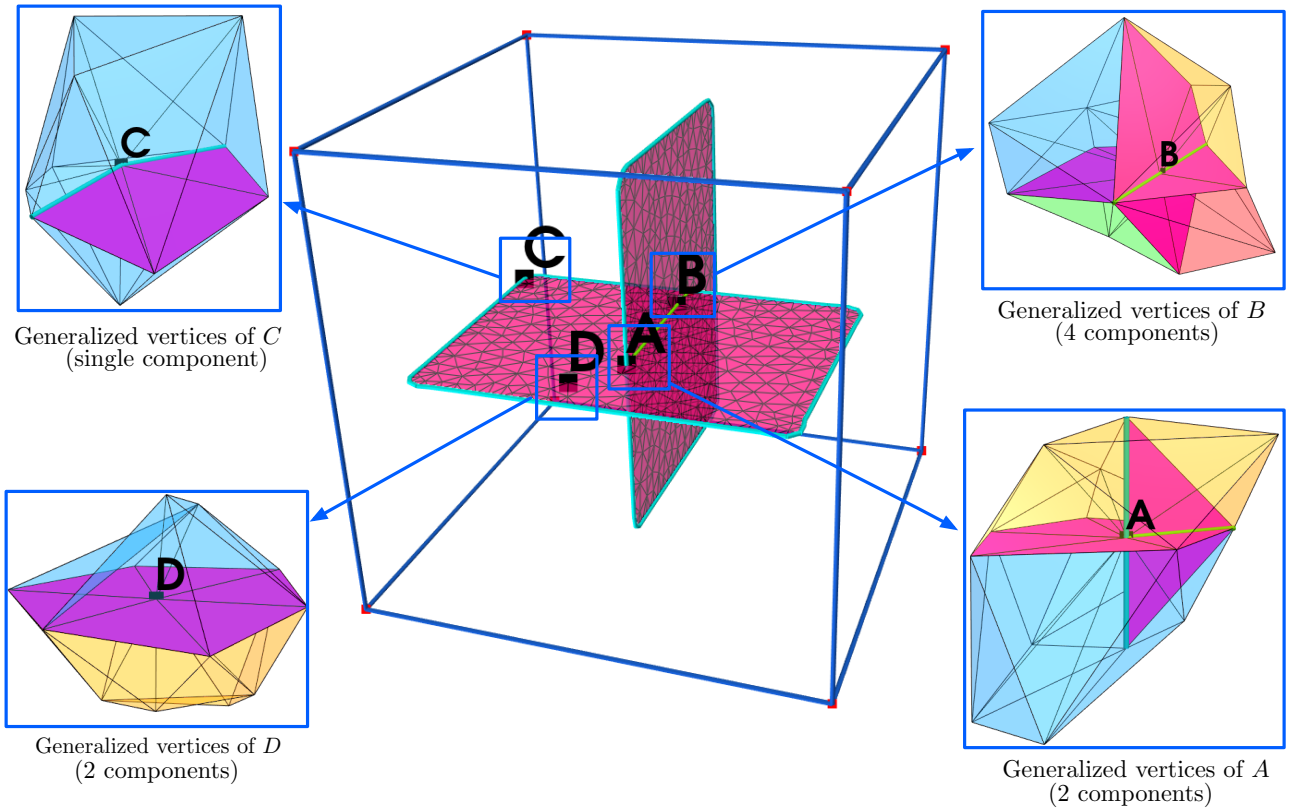


FIGURE 4. Fractured mesh of a multi-screen 3d domain with four particular generalized vertices A, B, C, D. The blue line represents the boundary of the fracture. The green line represents the intersection between the two screens. The vertices on the fracture have two generalized vertices (as in A or D), except those at the boundary of the fracture, that have only one (as in C), and those at the intersection between the two screens, that have four (as in B).

3. Finite element matrix assembly using a domain decomposition method

This section details the construction of the finite element matrix of a boundary value problem involving a fracture from the local finite element matrices associated to suitable restrictions of the latter on regular, non-cracked subdomains.

3.1. The Finite Element method on a cracked domain

Let us consider an elliptic boundary value problem posed on Ω, given by its variational form:

$$\text{Find } u \in V \text{ such that } a(u,v) = b(v), \quad \forall v \in V, \tag{3.1}$$

where

- The functional space $V := H^1(\Omega\backslash\Gamma)$ is the closure of $\mathcal{C}^\infty(\Omega\backslash\Gamma)$ for the norm
$$u \mapsto \left(\int_\Omega u^2 \,\mathrm{d}x + \int_{\Omega\backslash\Gamma} |\nabla u|^2 \,\mathrm{d}x \right)^{1/2};$$
it contains functions with possibly different traces from different sides of Γ, see Chap. 3 in [78];
- $a : V \times V \to \mathbb{R}$ is a coercive, continuous bilinear form on V;
- $b : V \to \mathbb{R}$ is a continuous linear form on V.

We further assume that a and b are *additive*, i.e. for any decomposition $\overline{\Omega} = \cup_{0\le p\le n}\overline{\Omega_p}$ featuring pairwise disjoint open subsets Ω_p of Ω, there exist continuous bilinear forms a_{Ω_p} and continuous linear forms b_{Ω_p} on V such that

$$\forall u, v \in V, \quad a(u,v) = a_{\Omega_1}(u_{|\Omega_1}, v_{|\Omega_1}) + \ldots + a_{\Omega_p}(u_{|\Omega_p}, v_{|\Omega_p}), \tag{3.2}$$

$$\forall v \in V, \quad b(v) = b_{\Omega_1}(v_{|\Omega_1}) + \ldots + b_{\Omega_p}(v_{|\Omega_p}). \tag{3.3}$$

For instance, this property is satisfied as soon as $a(u,v)$ and $b(v)$ are made of domain and surface integrals involving local differential operators applied to u and v.

Let us now introduce the "classical" Lagrange finite element space $V^k(\mathcal{M})$ of order $k \ge 0$ on the mesh $\mathcal{M}$:

$$V^k(\mathcal{M}) := \left\{ u \in H^1(\Omega) \;|\; u_{|K} \in \mathbb{P}_k \quad \forall K \in \mathcal{M} \right\}, \tag{3.4}$$

where $\mathbb{P}_k$ is the set of polynomials of degree k. We define the "fractured" finite element space $V^k(\mathcal{M};\Gamma)$ by:

$$V^k(\mathcal{M};\Gamma) := \{ u \in H^1(\Omega\backslash\Gamma) \;|\; u_{|K} \in \mathbb{P}_k \quad \forall K \in \mathcal{M} \}.$$

For the classical approximation properties of $V^k(\mathcal{M})$ in Sobolev spaces, see *e.g.* [21]. Similar approximation properties hold for $V^k(\mathcal{M};\Gamma)$ in $H^1(\Omega\backslash\Gamma)$, see [6, Theorem 1.2].

The finite element approximation of the variational problem (3.1) with $\mathbb{P}_k$ Lagrange finite elements $V^k(\mathcal{M};\Gamma)$ reads:

$$\text{Find } u_h \in V^k(\mathcal{M};\Gamma) \text{ such that } a(u_h, v_h) = b(v_h) \quad \text{for all } v_h \in V^k(\mathcal{M};\Gamma). \tag{3.5}$$

Let us now discuss the construction of a finite element basis for the space $V^k(\mathcal{M};\Gamma)$ and the algebraic system thus resulting from the discrete variational formulation (3.5). Let $\{x_1, \ldots, x_N\}$ be the usual set of $\mathbb{P}_k$ Lagrange degrees of freedom – hereafter called nodes – associated to $V^k(\mathcal{M})$, and let $\{\phi_1, \ldots, \phi_N\}$ be the corresponding basis of functions; in particular the functions ϕ_i are continuous across Γ. Every node x_i has $n_{x_i} \ge 1$ associated generalized nodes $\mathrm{st}_{\mathcal{M}}(x_i;1), \ldots, \mathrm{st}_{\mathcal{M}}(x_i;n_{x_i})$, see Section 2.2. Let us denote by $\mathcal{I}$ the index set of generalized degrees of freedom:

$$\mathcal{I} := \{ (i,\alpha) \;|\; 1 \le i \le N\,, 1 \le \alpha \le n_{x_i} \}\,,$$

and let $\mathcal{Q} := \#(\mathcal{I})$ be its cardinal. For $(i,\alpha) \in \mathcal{I}$, let $\phi_{i,\alpha}$ be the function which coincides with ϕ_i on the elements of the generalized component α of x_i, and which equals 0 on the other elements:

$$\forall K \in \mathcal{M}, \quad \forall x \in K\,, \quad \phi_{i,\alpha}(x) := \begin{cases} \phi_i(x) & \text{if } K \in \mathrm{st}_{\mathcal{M}}(x_i;\alpha), \\ 0 & \text{otherwise.} \end{cases} \tag{3.6}$$

According to [6, Lemma 6.2], the collection $\{\phi_{i,\alpha}\}_{(i,\alpha)\in\mathcal{I}}$ is a basis of $V^k(\mathcal{M};\Gamma)$. We finally introduce the operator $\tau : \mathbb{R}^{\mathcal{Q}} \to V^k(\mathcal{M};\Gamma)$ mapping coordinate vectors to finite element functions:

$$\tau[(U_{i,\alpha})_{(i,\alpha)\in\mathcal{I}}] := \sum_{(i,\alpha)\in\mathcal{I}} U_{i,\alpha}\phi_{i,\alpha}.$$

The stiffness matrix $\mathbf{A} \in \mathbb{R}^{\mathcal{Q}\times\mathcal{Q}}$ and the right-hand side $\mathbf{b} \in \mathbb{R}^{\mathcal{Q}}$ of the problem (3.5) in the basis $\{\phi_{i,\alpha}\}_{(i,\alpha)\in\mathcal{I}}$ of $V^k(\mathcal{M};\Gamma)$ are then defined by their entries:

$$\mathbf{A}_{(i,\alpha),(j,\beta)} = a(\phi_{i,\alpha}, \phi_{j,\beta}), \qquad \mathbf{b}_{i,\alpha} = b(\phi_{i,\alpha}), \qquad \forall (i,\alpha),(j,\beta) \in \mathcal{I}.$$

Equivalently, these objects are characterized by the following relations:

$$\mathbf{V}^T\mathbf{A}\mathbf{U} = a(\tau[\mathbf{U}], \tau[\mathbf{V}]), \quad \mathbf{V}^T\mathbf{b} = b(\tau[\mathbf{V}]), \qquad \forall \mathbf{U}, \mathbf{V} \in \mathbb{R}^{\mathcal{Q}}. \tag{3.7}$$

3.2. Construction of the finite element matrix from a partition of the mesh

The assembly of the global stiffness matrix $\mathbf{A}$ is tedious in numerical practice. It indeed raises the need to identify the generalized degrees of freedom of the space $V^k(\mathcal{M};\Gamma)$, that is, to determine how many copies of each degree of freedom of the "classical" space $V^k(\mathcal{M})$ appear in $V^k(\mathcal{M};\Gamma)$, and to endow them with a suitable label. This task is not only cumbersome, but it is also highly intrusive in the considered finite element solver. This is undesirable as one ideally wishes to use a third-party library without any information of its internal operation (which is usually not accessible to the user). The purpose of this section is to describe a calculation procedure of $\mathbf{A}$ from local finite element matrices obtained by a black-box application of an existing software on complementary regular subdomains of Ω which individually do not present cracks. The systematic construction of such a partition is the subject of Section 4.

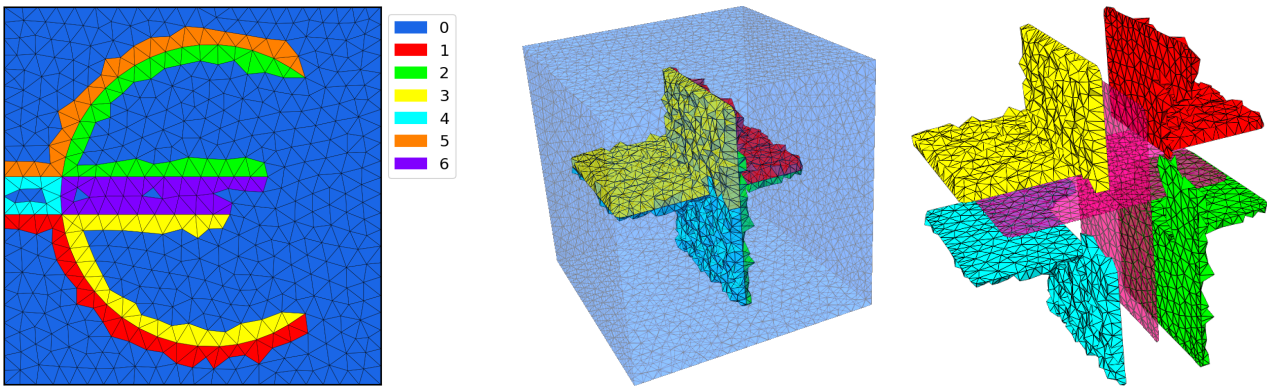
Let us assume that the mesh $\mathcal{M}$ is partitioned as:

$$\mathcal{M} = \bigcup_{p=0}^{n} \mathcal{M}_p, \tag{3.8}$$

where $\mathcal{M}_0, \dots, \mathcal{M}_n$ are $(n+1)$ disjoint sub-meshes on which the local assemblies will take place. We assume that whenever two elements K, K' are incident "across the fracture", they belong to distinct components of the partition. More precisely, we assume that the following property holds:

$$\forall S \in \sigma(\mathcal{M}),\ \forall 1 \le \alpha \ne \beta \le n_S,\ \forall (K,K') \in \mathrm{st}_{\mathcal{M}}(S;\alpha) \times \mathrm{st}_{\mathcal{M}}(S;\beta),$$
$$\forall 0 \le p,q \le n, \quad K \in \mathcal{M}_p \text{ and } K' \in \mathcal{M}_q \Rightarrow p \ne q. \tag{P}$$

Such a partition always exists; for instance, the trivial partition $\mathcal{M} = \bigcup_{K\in\mathcal{M}}\{K\}$, obtained by singling out all the elements of $\mathcal{M}$, obviously satisfies (P): then, $n = \#(\sigma_d(\mathcal{M})) - 1$ and for any $p = 0, \dots, n$, $\mathcal{M}_p = \{K_{p+1}\}$. Note however that, in practice, in order to limit the number of calls to a third-party software for assembling the stiffness matrix $\mathbf{A}$, it is desirable that the number n of components of the partition be as small as possible, see Figure 5 for an illustration.



(A) Partitioning obtained with the graph partitioning algorithm of Section 4 for the 2d example.

(B) Partition obtained with the algorithm of Section 4 for the 3d multi-screen. The right panel shows the four components adjacent to the fracture.

FIGURE 5. Partitions of the 2d and 3d meshes associated to the self intersecting crack and multi-screen geometries of Figure 1, satisfying the property (P).

For $0 \le p \le n$, we denote by Ω_p the interior of the subdomain $|\mathcal{M}_p|$ spanned by the triangulation $\mathcal{M}_p$. The property (P) guarantees that the Ω_p are "regular" domains, that are situated locally on one side of the fracture Γ, as formalized by the following proposition.

Proposition 3.1. *Any mesh decomposition (3.8) satisfying (P) also complies with the following properties:*

(i) The fracture Γ is contained in the boundaries of the subdomains: $\Gamma \subset \bigcup_{0\le p\le n} \partial\Omega_p$;

(ii) For each $0 \le p \le n$, the star of any sub-simplex S in $\mathcal{M}_p$ is included in exactly one of the generalized sub-simplices of S in $\mathcal{M}$:

$$\forall 0 \le p \le n,\ \forall S \in \sigma(\mathcal{M}_p),\ \exists! 1 \le \alpha \le n_S,\ \text{s.t. } \mathrm{st}_{\mathcal{M}_p}(S) \subset \mathrm{st}_{\mathcal{M}}(S;\alpha);$$

(iii) The previous property also holds true for the star of any point $x \in \overline{\Omega_p}$:

$$\forall 0 \le p \le n,\ \forall x \in \overline{\Omega_p},\ \exists! 1 \le \alpha \le n_x,\ \text{s.t. } \mathrm{st}_{\mathcal{M}_p}(x) \subset \mathrm{st}_{\mathcal{M}}(x;\alpha).$$

Proof. (i). We show that:

$$\text{For all } \; 0 \le p \le n, \quad S \in \sigma_{d-1}(\mathcal{M}_p) \cap \Gamma \Rightarrow S \in \partial\Omega_p. \tag{3.9}$$

In fact, let $S \in \sigma_{d-1}(\mathcal{M}_p) \cap \Gamma$, and suppose, for a contradiction, that $S \notin \partial\Omega_p$. Then Proposition 2.1 implies that there exist $K, K' \in \mathcal{M}_p$ with $K \neq K'$ such that $S \subset K$ and $S \subset K'$. However, because $\mathcal{M}$ is a simplicial mesh, S has exactly two associated generalized $(d-1)$-sub-simplices, K and K', that are separated by $S \subset \Gamma$. Therefore, K and K' cannot both belong to $\mathcal{M}_p$ due to (P), which is the desired contraction, proving (3.9).

(ii). Let $S \in \sigma(\mathcal{M}_p)$, and assume that, for any $1 \le j \le n_S$, $\mathrm{st}_{\mathcal{M}_p}(S) \not\subset \mathrm{st}_{\mathcal{M}}(S; j)$. Then, there exist two elements $K, K' \in \mathrm{st}_{\mathcal{M}_p}(S)$ and $1 \le j \ne k \le n_S$ such that $K \in \mathrm{st}_{\mathcal{M}}(S; j)$ and $K' \in \mathrm{st}_{\mathcal{M}}(S; k)$. This contradicts (P).

(iii). This is an immediate consequence of Lemma 2.2 and the previous point. □

3.3. **Finite Element assembly of the cracked problem from local matrices**

Let us denote by $a_p := a_{\Omega_p}$ and $b_p := b_{\Omega_p}$ the bilinear and linear forms induced by a and b on the sub-domains Ω_p, $p = 0, \dots, n$, see Assumptions (3.2) and (3.3). These satisfy:

$$\forall u, v \in H^1(\Omega \backslash \Gamma), \qquad a(u,v) = \sum_{p=0}^{n} a_p(u|_{\Omega_p}, v|_{\Omega_p}), \qquad b(v) = \sum_{p=0}^{n} b_p(v|_{\Omega_p}). \tag{3.10}$$

For $0 \le p \le n$, we denote by $\mathbf{A}_p$ and $\mathbf{b}_p$ the finite element matrices and vectors of the sub-problem

$$\text{Find } u_h, v_h \in V^k(\mathcal{M}_p) \text{ such that } a_p(u_h, v_h) = b_p(v_h) \text{ for all } v_h \in V^k(\mathcal{M}_p). \tag{3.11}$$

Since the latter involves the "classical" Lagrange finite element space $V^k(\mathcal{M}_p)$ defined by (3.4), $\mathbf{A}_p$ and $\mathbf{b}_p$ can be computed using any available third-party software.

For $0 \le p \le n$, let $\mathcal{I}_p$ be the set of degrees of freedom of $V^k(\mathcal{M}_p)$, or equivalently, the subset of indices of the Lagrange degrees of freedom of $\mathcal{M}$ that belong to $\overline{\Omega_p}$:

$$\mathcal{I}_p := \{1 \le i \le N \,|\, x_i \in |\mathcal{M}_p|\}, \text{ and } \; \widehat{N}_p := \#(\mathcal{I}_p).$$

A basis of $V^k(\mathcal{M}_p)$ is made of the restrictions $(\phi_i|_{\Omega_p})_{i \in \mathcal{I}_p}$ of the basis functions of the complete finite element space $V^k(\mathcal{M})$. Introducing the mappings $\tau_p : \mathbb{R}^{\widehat{N}_p} \to H^1(\Omega_p)$ between coordinate vectors and finite element functions in $V^k(\mathcal{M}_p)$, i.e.

$$\forall\, \mathbf{U} = (U_i)_{i \in \mathcal{I}_p} \in \mathbb{R}^{\widehat{N}_p}, \quad \tau_p[\mathbf{U}] := \sum_{i \in \mathcal{I}_p} U_i \phi_i|_{\Omega_p},$$

the matrices and vectors $\mathbf{A}_p \in \mathbb{R}^{\widehat{N}_p \times \widehat{N}_p}$ and $\mathbf{b}_p \in \mathbb{R}^{\widehat{N}_p}$ satisfy:

$$\forall\, \mathbf{U}, \mathbf{V} \in \mathbb{R}^{\widehat{N}_p}, \qquad \mathbf{V}^T \mathbf{A}_p \mathbf{U} = a_p(\tau_p[\mathbf{U}], \tau_p[\mathbf{V}]), \quad \mathbf{V}^T \mathbf{b}_p = b_p(\tau_p[\mathbf{V}]).$$

Explicitly, their entries read:

$$(\mathbf{A}_p)_{ij} = a_p(\phi_i|_{\Omega_p}, \phi_j|_{\Omega_p}), \qquad (\mathbf{b}_p)_j = b_p(\phi_i|_{\Omega_p}), \qquad i, j \in \mathcal{I}_p.$$

Let us now explain how the total finite element matrix $\mathbf{A}$ and right-hand side $\mathbf{b}$ of the fractured problem (3.5) can be constructed from their local counterparts $\mathbf{A}_p$, $\mathbf{b}_p$. We define the mappings $\alpha_p : \mathcal{I}_p \to \mathbb{N}$ indicating the number of the generalized node in $\mathcal{M}$ associated to an input node in $\mathcal{M}_p$:

$$\forall\, 0 \le p \le n, \;\; \forall i \in \mathcal{I}_p, \;\; \alpha_p(i) \in \{1, \dots, n_{x_i}\} \text{ is the unique integer s.t. } \mathrm{st}_{\mathcal{M}_p}(x_i) \subset \mathrm{st}_{\mathcal{M}}(x_i; \alpha_p(i)); \tag{3.12}$$

see Proposition 3.1. These mappings induce the following decomposition of the index set $\mathcal{I}$ of generalized nodes:

$$\mathcal{I} = \bigcup_{p=0}^{n} \{(i, \alpha_p(i)) \,|\, i \in \mathcal{I}_p\}. \tag{3.13}$$

Note that this reunion may feature redundant elements. For instance, a node x_i lying "far" from the fracture set Γ – and thus having one single generalized vertex $\mathrm{st}_{\mathcal{M}}(x_i; 1)$ – may belong to two elements K, K' affected to different submeshes $\mathcal{M}_p$ and $\mathcal{M}_q$; in this case, $\alpha_p(i) = \alpha_q(i) = 1$. Drawing inspiration from domain decomposition methods [45], we now define, for $p = 0, \dots, n$, the restriction operator $\mathcal{R}_p : \mathbb{R}^{\mathcal{Q}} \to \mathbb{R}^{\widehat{N}_p}$ by

$$\mathcal{R}_p[(U_{i,\alpha})_{(i,\alpha) \in \mathcal{I}}] := (U_{i,\alpha_p(i)})_{i \in \mathcal{I}_p}. \tag{3.14}$$

A straightforward verification shows that the transpose operator $\mathcal{R}_p^T : \mathbb{R}^{\widehat{N}_p} \to \mathbb{R}^{\mathcal{Q}}$ realizes the extension by zero of a vector $(U_i)_{i \in \mathcal{I}_p}$ to the indices $\mathcal{I} \backslash \{(i, \alpha_p(i)) \,|\, i \in \mathcal{I}_p\}$. In practice, the operator $\mathcal{R}_p$ is implemented as a boolean matrix with size $\widehat{N}_p \times \mathcal{Q}$, which is also denoted by $\mathcal{R}_p$ with a small abuse of notation.

The main ingredient in the assembly of $\mathbf{A}$ and $\mathbf{b}$ on the basis of their contributions from the sub-meshes $\mathcal{M}_p$ is the next Proposition 3.2. It crucially relies on the fact that the basis functions $\phi_{i,\alpha}$ of the cracked problem (3.5) are the restrictions of the "classical" Lagrange basis functions ϕ_i of $\mathcal{M}$ to the submeshes $(\mathcal{M}_p)_{0\leq p\leq n}$. We start with a lemma.

Lemma 3.1. *Let $\mathcal{M} = \bigcup_{p=0}^n \mathcal{M}_p$ be a partition satisfying the property (P). The restriction of a finite element function $u \in V^k(\mathcal{M};\Gamma)$ to Ω_p is given by the restricted component vector:*

$$\text{For any } \mathbf{U} = (U_{i,\alpha})_{(i,\alpha)\in\mathcal{I}} \in \mathbb{R}^{\mathcal{Q}}, \quad \tau[\mathbf{U}]|_{\Omega_p} = \tau_p[\mathcal{R}_p\mathbf{U}].$$

Proof. This follows from an elementary calculation:

$$\begin{aligned} \tau[\mathbf{U}]|_{\Omega_p} &= \sum_{(i,\alpha)\in\mathcal{I}} U_{i,\alpha}\phi_{i,\alpha}|_{\Omega_p} \\ &= \sum_{i\in\mathcal{I}_p} U_{i,\alpha_p(i)}\phi_i|_{\Omega_p} \\ &= \tau_p[\mathcal{R}_p\mathbf{U}], \end{aligned} \tag{3.15}$$

where the second line follows from the definition (3.6) of the basis functions $\{\phi_{i,\alpha}\}_{(i,\alpha)\in\mathcal{I}_p}$. □

Proposition 3.2. *Let $\mathcal{M} = \bigcup_{p=0}^n \mathcal{M}_p$ be a mesh partition satisfying the property (P). The finite element matrix $\mathbf{A}$ and right-hand side $\mathbf{b}$ of the "fractured" problem (3.5) are related to the finite element matrices $(\mathbf{A}_p)_{0\leq p\leq n}$ and vectors $(\mathbf{b}_p)_{0\leq p\leq n}$ of the local problems (3.11) through the following "domain decomposition"-like formulas:*

$$\mathbf{A} = \sum_{p=0}^n \mathcal{R}_p^T\mathbf{A}_p\mathcal{R}_p, \qquad \mathbf{b} = \sum_{p=0}^n \mathcal{R}_p^T\mathbf{b}_p. \tag{3.16}$$

Proof. Let $\mathbf{U},\mathbf{V}$ be arbitrary vectors in $\mathbb{R}^{\mathcal{Q}}$. From (3.10) and (3.15), we have:

$$\begin{aligned} \mathbf{V}^T\mathbf{A}\mathbf{U} &= a(\tau[\mathbf{U}],\tau[\mathbf{V}]) = \sum_{p=0}^n a_p(\tau[\mathbf{U}]|_{\Omega_p},\tau[\mathbf{V}]|_{\Omega_p}) = \sum_{p=0}^n a_p(\tau_p[\mathcal{R}_p\mathbf{U}],\tau_p[\mathcal{R}_p\mathbf{V}]) \\ &= \sum_{p=0}^n \mathbf{V}^T\mathcal{R}_p^T\mathbf{A}_p\mathcal{R}_p\mathbf{U}. \end{aligned}$$

Likewise, for any vector $\mathbf{V}\in\mathbb{R}^{\mathcal{Q}}$, we have:

$$\begin{aligned} \mathbf{V}^T\mathbf{b} &= b(\tau[\mathbf{V}]) = \sum_{p=0}^n b_p(\tau[\mathbf{V}]|_{\Omega_p}) = \sum_{p=0}^n b_p(\tau_p[\mathcal{R}_p\mathbf{V}]) \\ &= \sum_{p=0}^n \mathbf{V}^T\mathcal{R}_p^T\mathbf{b}_p. \end{aligned}$$

These result in the desired formulas. □

4. Mesh decomposition using a greedy algorithm for graph partitioning with cannot-link constraints

This section describes an efficient numerical algorithm for computing a suitable partition of the total mesh $\mathcal{M}$ into several components satisfying the property (P).

The proposed method relies on graph theory, and it operates on the undirected graph $\mathcal{G}(\mathcal{M})$ defined from $\mathcal{M}$ as follows:

- The nodes of $\mathcal{G}(\mathcal{M})$ are the elements K of $\mathcal{M}$,
- Two elements $K,K'\in\mathcal{M}$ are connected by an edge $\{K,K'\}$ if they are adjacent through a face.

We additionally introduce a subset $\mathcal{CL}$ of "cannot-link" edges within $\mathcal{G}(\mathcal{M})$. The latter gather the pairs $\{K,K'\}$ made of two elements separated by the fracture Γ; *i.e.*,

$$\mathcal{CL} := \Big\{\{K,K'\}\,|\,\exists S\in\sigma(\mathcal{M}),\ n_S\geq 1,\ 1\leq\alpha\neq\beta\leq n_S,\ K\in \mathrm{st}_{\mathcal{M}}(S;\alpha) \text{ and } K'\in \mathrm{st}_{\mathcal{M}}(S;\beta)\Big\}. \tag{4.1}$$

In words, the cannot-link edges $\{K,K'\}\in\mathcal{CL}$ are those made of elements K and K' that belong to two different generalized sub-simplices of a single sub-simplex $S\in\sigma(\mathcal{M})$. Note that $\mathcal{CL}$ contains not only the pairs of elements that are adjacent through a face of the fracture set Γ, but all the pairs that share entities from the fracture Γ and must be "seen" from different sides of the latter. An algorithm to generate the set of constraints $\mathcal{CL}$ starting from the mesh $\mathcal{M}$ and the fracture Γ is presented in Section 6.1.

From this perspective, the search for a partition $\mathcal{M} = \bigcup_{p=0}^n \mathcal{M}_p$ of $\mathcal{M}$ satisfying the property (P) boils down to finding a partition of the graph $\mathcal{G}(\mathcal{M})$ into $(n+1)$ disjoint connected components $\mathcal{G}_0,\dots\mathcal{G}_n$ subject to

the "cannot-link" constraint that any two nodes K, K' that are connected by an edge $\{K, K'\} \in \mathcal{CL}$ must be affected to different components.

The setting and notations from graph theory needed to address this general problem are introduced in Section 4.1 and our algorithm is described in Section 4.2.

4.1. Definitions and notations about graphs and partitions

Let us start by recalling a few classical definitions about the objects at stake in this section, see e.g. [16, 105] for more exhaustive introductions to graph theory.

Definition 4.1. A *graph* $\mathcal{G}$ is given by a non empty and finite set $\mathcal{V}$ of elements called *vertices*, and a collection $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ of unoriented pairs $\{v, w\}$ of vertices $v, w \in \mathcal{V}$ called *edges*. Moreover,

- The *degree* $\deg(v)$ of a vertex $v \in \mathcal{V}$ is the number of distinct edges $e \in \mathcal{E}$ having v as one of their endpoints.
- The graph $\mathcal{G}$ is *connected* if for any two vertices $v, w \in \mathcal{V}$, there exist edges $(v^0, v^1), \dots (v^{k-1}, v^k) \in \mathcal{E}$ with $v^0 = v$ and $v^k = w$.

The graphs considered in this article are assumed to be undirected, that is $\{v, w\} \in \mathcal{E}$ if and only if $\{w, v\} \in \mathcal{E}$. However, all that follows would also hold true in the case of oriented graphs.

For a given collection of vertices $S \subset \mathcal{V}$, we denote by $\mathcal{G}[S]$ the sub-graph induced by S:

$$\mathcal{G}[S] := (S, \mathcal{E}_S) \text{ with } \mathcal{E}_S := \{(v, w) \in S \times S \,|\, (v, w) \in \mathcal{E}\}.$$

Definition 4.2 (Partition of a graph)**.** A *partition* of the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is a decomposition of the set $\mathcal{V}$ of vertices into $(n+1)$ components:

$$\mathcal{V} = \bigcup_{0 \le p \le n} \mathcal{V}_p \qquad \text{with } \mathcal{V}_p \cap \mathcal{V}_q = \emptyset \quad \forall 0 \le p \ne q \le n,$$

and such that for any $0 \le p \le n$, $\mathcal{G}[\mathcal{V}_p]$ is connected.

Definition 4.3 (Cannot-link constraints)**.** Let $\mathcal{CL}$ be a subset of $\mathcal{V} \times \mathcal{V}$, gathering the "cannot-link" pairs of vertices. A partition $(\mathcal{G}[V_p])_{0 \le p \le n}$ of $\mathcal{G}$ *complies with the cannot-link constraints* $\mathcal{CL}$ if any two vertices forming a pair in $\mathcal{CL}$ are affected to distinct components of the partition:

$$\forall\ \{v, w\} \in \mathcal{CL}, \forall\, 0 \le p, q \le n, \quad v \in V_p \text{ and } w \in V_q \Rightarrow p \ne q.$$

Obviously, for any graph $\mathcal{G}$ and cannot-link set $\mathcal{CL}$, the trivial partition of $\mathcal{G}$ induced by the decomposition of $\mathcal{V}$ into singletons, i.e.

$$V = \bigcup_{v \in \mathcal{V}} \{v\} \tag{4.2}$$

obviously satisfies the cannot-link constraints $\mathcal{CL}$. However, in applications, it is often desirable for a partition of $\mathcal{G}$ to have a moderate number n of components.

More generally, constrained graph partitioning—with cardinality, membership, cannot-link, or must-link conditions—is classical in applications such as mesh partitioning [69], network design [85] and data clustering [8]; see [24] for a review. In the cannot-link setting considered here, the problem is often formulated as the search for the minimum number of edges to remove to satisfy the constraints [4, 86, 11]:

$$\min n \quad \text{s.t. } (\mathcal{G}[\mathcal{V}_p])_{0 \le p \le n} \text{ is a partition of } \mathcal{G} \text{ satisfying the constraints } \mathcal{CL} \tag{4.3}$$

Unfortunately, this combinatorial problem is notoriously difficult, probably NP-hard. In this work, we rather aim to find a partition which is optimal in some Pareto sense with a complexity which is nearly linear in the number of vertices in $\mathcal{V}$ and the number of cannot-link constraints.

Definition 4.4 (Minimal partitioning)**.** A graph partitioning $(\mathcal{G}[\mathcal{V}_p])_{0 \le p \le n}$ is called *minimal* for the problem (4.3) if it is not possible to merge two of its components without violating the cannot-link constraints $\mathcal{CL}$ defined by (4.1):

$$\forall 0 \le p \ne q \le n, \quad \mathcal{G}[\mathcal{V}_p \cup \mathcal{V}_q] \text{ is connected } \Rightarrow \exists (v, w) \in \mathcal{CL} \text{ with } v \in \mathcal{V}_p \text{ and } w \in \mathcal{V}_q.$$

The resulting partition may not be "optimal", strictly speaking. However, it usually has a moderate number of components, which is sufficient for our purpose; it is moreover minimal in the sense that no two of these can be merged without violating the constraints.

4.2. **A greedy algorithm for graph partitioning with cannot-link constraints**

This section describes a greedy algorithm that computes a minimal partition of a graph $\mathcal{G}$ that complies with a given set $\mathcal{CL}$ of cannot-link constraints, in the sense of Definition 4.4. Algorithm 1 operates as follows. It starts from the trivial partition (4.2) of $\mathcal{G}$, featuring as many components as vertices in this graph, and then travels the edges $\{v, w\} \in \mathcal{E}$ of $\mathcal{G}$ to try and merge the components containing v and w. In order to manage the connected components efficiently while ensuring that cannot-link constraints are not violated, the algorithm relies on a Union-Find data structure [62] alongside an auxiliary graph $\textsc{Forbidden}=(\mathcal{U}, \mathcal{F})$, that are both updated dynamically:

- The Union-Find structure handles the maintenance of the partition. It consists of:
  - A function $\textsc{Find} : \mathcal{V} \to \mathcal{V}$ that maps each vertex v to a representative $\textsc{Find}(v)$ of its component in the current partition. This mapping completely characterizes the partition as:
$$\mathcal{V} = \bigcup_{i \in \textsc{Find}(V)} \{v \in \mathcal{V} \mid \textsc{Find}(v) = i\}. \tag{4.4}$$
    At the initial stage of the algorithm, one sets: $\textsc{Find}(v) = v$ for all $v \in \mathcal{V}$.
  - The $\textsc{Union}$ procedure allows to merge two components: subsequent calls to $\textsc{Find}$ return a single shared representative for vertices in both sets. An efficient tree-based implementation of this operation with path compression results in a complexity of $\mathcal{O}(\alpha^*(\#(\mathcal{V})))$ per operation, where α^* denotes the inverse Ackermann function [35, 103].
- The graph $\textsc{Forbidden}$ $(\mathcal{U}, \mathcal{F})$ encodes and dynamically updates the constraints between the evolving components of $\mathcal{V}$. It enables to identify the components that cannot be merged without violating the cannot-link constraints $\mathcal{CL}$. This graph is made of the following objects:
  (i) Its set of vertices $\mathcal{U} = \{\textsc{Find}(v) \mid v \in V\}$ consists of the representatives of the current partition.
  (ii) Its set of edges $\mathcal{F} = \{(\textsc{Find}(v), \textsc{Find}(w)) \in \mathcal{U} \times \mathcal{U} \mid \{v, w\} \in \mathcal{CL}\}$ accounts for the pairs of components that should not be merged.

Algorithm 1 Greedy graph partitioning algorithm with cannot-link constraints.

Require: Graph $\mathcal{G}$ with vertices $\mathcal{V}$ and edges $\mathcal{E}$.
Require: Cannot-link pairs $\mathcal{CL} \subset \mathcal{V} \times \mathcal{V}$.
1: **Initialize** Union-Find data structure with $\#(\mathcal{V})$ elements.
2: **Initialize** graph structure $\textsc{Forbidden}= (\mathcal{U}, \mathcal{F})$ with $\mathcal{U} = \mathcal{V}$ and $\mathcal{F} = \mathcal{CL}$.
3: **for all** edge $\{v, w\} \in \mathcal{E}$ **do**
4: **if** $\{\textsc{Find}(v), \textsc{Find}(w)\} \notin \mathcal{F}$ **then**
5: $\textsc{Union}(v, w)$
6: $\textsc{Merge}(\textsc{Find}(v),\textsc{Find}(w))$ into $\textsc{Find}(v)$.
7: **end if**
8: **end for**
9: **return** $\textsc{Find}$.

At any stage of the algorithm, the two components under scrutiny, represented by $\textsc{Find}(v)$ and $\textsc{Find}(w)$, are merged provided they are not connected by an edge in the graph $\textsc{Forbidden}$. After this happens, the $\textsc{Merge}$ procedure is used to update the Union-Find structure and the graph $\textsc{Forbidden}$, see line 6: the nodes $\textsc{Find}(v)$ and $\textsc{Find}(w)$ in $\textsc{Forbidden}$ are merged, and all edges incident to either of the two original representatives are redirected to the new common representative. By merging the node with smallest degree into that with larger degree, the cost of the merger of two nodes A, $B \in \mathcal{U}$ is of the order of the minimum of the degrees of A and B in the graph $\textsc{Forbidden}$.

Proposition 4.1. *Algorithm 1 computes a minimal (in the sense of Definition 4.4) partition of $\mathcal{G}$ in at most*
$$\mathcal{O}\Big(\#(\mathcal{E})\alpha^*(\#(V)) + \#(\mathcal{CL})\log(\#(V))\Big)$$
operations, where α^* is the inverse Ackerman function.

Proof. The complexity of Algorithm 1 is mainly determined by two factors: the management of the vertex partition using the Union-Find structure and the upkeep of the $\textsc{Forbidden}$ graph accounting for the cannot-link constraints. The algorithm considers each edge $\{v, w\} \in \mathcal{E}$ exactly once. For each edge, it performs a constant number of $\textsc{Find}$ operations, and if a merger is possible, a $\textsc{Union}$ operation. As previously mentioned, using a Union-Find data structure with path compression, the total time spent on these operations is at most $\mathcal{O}(\#(\mathcal{E})\alpha^*(\#(\mathcal{V})))$ [35, 103].

The most significant overhead stems from the update of the graph $\textsc{Forbidden}$ associated to the merging operations, which can be estimated thanks to a classical doubling argument. Let e be an arbitrary cannot-link pair of vertices in the set $\mathcal{CL}$, that will be updated as the algorithm proceeds. The merging operations induce a

cost of one operation for e every time one of its nodes is merged with another node with higher degree. Let us denote by (u_k) and $(v_{k'})$ the sequences of nodes of e, where the indices k and k' refer to the number of merging operations that have led to u_k and $v_{k'}$. By construction, each merging operation related to either vertex causes its degree to double (at least), up to the deletion of one edge. Hence, the degree sequences satisfy

$$\deg(u_{k+1}) \geq 2\deg(u_k) - 1 \text{ and } \deg(v_{k'+1}) \geq 2\deg(v_{k'}) - 1.$$

Since the degree of both nodes must remain inferior to $|\mathcal{CL}|$, the total cost of the merging operations affecting a single edge e in $\mathcal{CL}$ is of the order $\mathcal{O}(\log(|\mathcal{CL}|))$. Consequently, the total cost of merging nodes in the FORBIDDEN graph is bounded by $\mathcal{O}(|\mathcal{CL}|\log(|\mathcal{CL}|))$, which is $\mathcal{O}(|\mathcal{CL}|\log|\mathcal{V}|)$ since $|\mathcal{CL}| \leq |\mathcal{V}|^2$.

Finally, the output partitioning is necessarily minimal, because Algorithm 1 has considered uniting all possible edges of $\mathcal{E}$. □

Remark 4.1. The complexity of Algorithm 1 can be further reduced by partitioning only the subgraph of $\mathcal{G}$ made of the nodes that pertain to at least one edge in the cannot-link constraint set $\mathcal{CL}$. Indeed, let

$$\mathcal{V}_0 := \{u \in \mathcal{V} \,|\, \forall (v,w) \in \mathcal{CL}, \quad u \notin (v,w)\}$$

be the set of nodes in $\mathcal{V}$ that do not appear in the cannot-link set $\mathcal{CL}$. It is sufficient to compute a partition $\mathcal{V}\backslash\mathcal{V}_0 = \bigcup_{1\leq p\leq n} \mathcal{V}_i$ of $\mathcal{G}[\mathcal{V}\backslash\mathcal{V}_0]$ into n components by application of Algorithm 1 and return the partition $\mathcal{V} = \bigcup_{p=0}^{n} \mathcal{V}_p$. In the motivating application of this article to constrained mesh partitioning, the set $\mathcal{V}_0$ corresponds to the submesh $\mathcal{M}_0$ of $\mathcal{M}$ given by:

$$\mathcal{M}_0 := \mathcal{M}\backslash\mathcal{M}^c, \text{ where the envelope of } \Gamma \text{ is } \mathcal{M}^c := \bigcup_{\substack{S\in\sigma(\mathcal{M}),\\ n_S>1}} \mathrm{st}_{\mathcal{M}}(S),$$

i.e. $\mathcal{M}^c$ is the mesh whose elements are those that belong to the stars of simplices $S \in \sigma(\mathcal{M})$ having more than one single generalized component, see Figure 5b for a 3d illustration. Partitioning $\mathcal{M}^c$ instead of $\mathcal{M}$ is computationally advantageous since it only contains the elements adjacent to Γ, and so $\mathcal{M}^c$ is a significantly smaller mesh than $\mathcal{M}$. The partitionings displayed on Figure 5 rely on this trick. A partitioning of the same 2d and 3d cracked geometries obtained by direct applications of Algorithm 1 to the full mesh $\mathcal{M}$ are shown on Figure 6 and Figure 7, respectively. Partitioning the complete mesh $\mathcal{M}$ instead of $\mathcal{M}^c$ yields slightly fewer components in the 2d situation (5 instead of 6), but it results in a larger number of components in 3d (10 instead of 5), where the larger space dimension leaves more opportunities for this number to increase with the number of mesh elements.

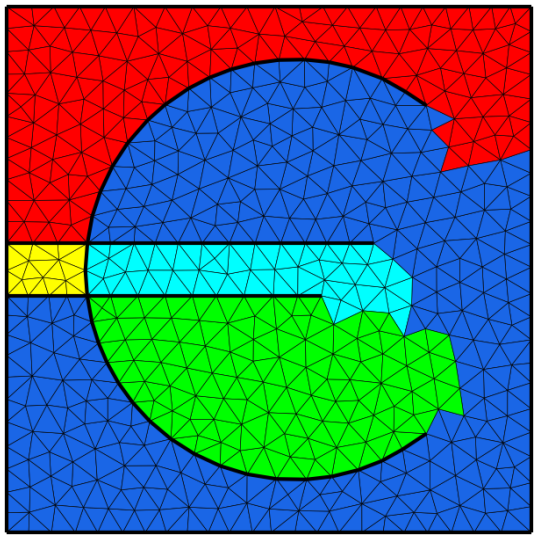


FIGURE 6. Partition of the self-intersecting crack mesh obtained by direct application of Algorithm 1 to $\mathcal{M}$ and not to its envelope $\mathcal{M}^c = \mathcal{M}\backslash\mathcal{M}_0$.

5. CONSTRUCTION OF FRACTURED MESHES AND REMESHING

This section deals with the construction of a conforming mesh of the domain Ω containing a discretization of the fracture Γ as a collection of sub-simplices, see Figure 1. This operation is useful in itself: for instance, it is used in geophysics to insert fault lines in a mesh of the underground [12, 71]. It is also one of the basic ingredients of the body-fitted method developed in [36, 37] to track the evolution of an open curve in 2d, and an open surface in 3d. Since the aforementioned articles are specifically devoted to the construction method of a conforming mesh of a fractured geometry used in this article, we limit ourselves with a brief presentation

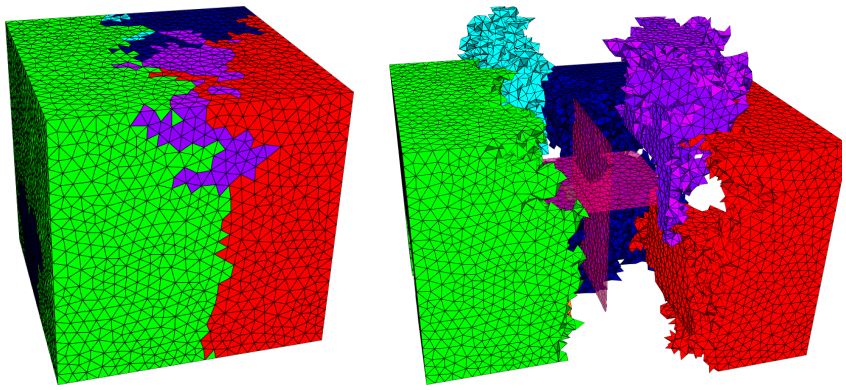


FIGURE 7. Partition of the cubic mesh containing the multi-screen geometry by partitioning directly $\mathcal{M}$ with Algorithm 1, rather than its envelope $\mathcal{M}^c$.

of the main ideas. The open surface discretization and remeshing functionalities used to generate all meshes considered in this article are implemented in the `-lsopen` command-line option of the open-source software `mmg` [81].

In this section again, let Ω be a Lipschitz polygonal domain in $\mathbb{R}^d$, equipped with a conforming simplicial mesh $\mathcal{M}$, and let $\Gamma \Subset \Omega$ be a codimension 1 submanifold of $\mathbb{R}^d$ with non-empty boundary accounting for the fracture. We aim to create a new mesh $\widetilde{\mathcal{M}}$ of Ω featuring an explicit discretization of Γ, as a collection $\mathcal{S}$ of $(d-1)$ sub-simplices of $\mathcal{M}$. Although this basic operation only applies to the case where Γ is a manifold (possibly with boundary), repeated applications of this basic operation allow to handle general, non-manifold fractures Γ (*e.g.*, with multiple junctions), as illustrated in Figure 10. Note that, for simplicity but without loss of generality, we solely consider the situation where Γ is a manifold (with non empty boundary), as successive applications of this basic operation allow to deal with complex, non-manifold shapes of Γ (e.g. presenting multiple junctions), see Figure 10.

Under this assumption, as suggested in e.g. [10, 25, 56, 97], Γ can be accounted for by two "level set" functions $\phi, \psi : \Omega \to \mathbb{R}^d$, in the sense that:

$$\Gamma = \Big\{ x \in \Omega, \;\; \phi(x) = 0 \text{ and } \psi(x) < 0 \Big\}. \tag{5.1}$$

Intuitively, Γ is the intersection of the extending (closed) hypersurface $\widetilde{\Gamma} := \{x \in \Omega, \;\; \phi(x) = 0\}$ with the negative subregion $\{x \in \Omega, \;\; \psi(x) < 0\}$ of ψ in Ω, see Figure 8. In practice, ϕ and ψ are defined at the vertices of the mesh $\mathcal{M}$, and their values at other points of Ω are obtained by linear interpolation. Such a representation of the fracture Γ is easily constructed when it has a "simple" shape, e.g. when it is a surface disk in 3d. In the general situation, for instance when Γ is supplied as a mesh, it can be generated by a variant of the Fast Marching Method [96] which is detailed in the forthcoming work [37].

From this two level set representation of Γ, our method operates in two stages, illustrated in Figure 9 and described in more details in the next subsections:

(i) We discretize Γ explicitly into the mesh $\mathcal{M}$. This task is purely logical, and it results in an intermediate conforming but low-quality mesh $\mathcal{M}_{\text{temp}}$ of Ω, bearing a discretization of Γ via a collection of sub-simplices $\mathcal{S}_{\text{temp}}$.
(ii) Remeshing is applied to improve the quality of $\mathcal{M}_{\text{temp}}$.

5.1. **Rough insertion of the fracture into the mesh**

This stage is a variation of the marching tetrahedra algorithm [26, 44] for isosurface extraction, which is itself a simplicial counterpart of the celebrated marching cubes algorithm [75]. It starts from an initial mesh $\mathcal{M}$ of the computational domain Ω, in which the fracture Γ, supplied by the two level set representation in (5.1), is not yet discretized. There, we identify the set $\mathcal{K}$ of the elements $K \in \mathcal{M}$ that satisfy the following two conditions:

- ϕ takes a positive value in at least one of the vertices of K and a negative value in at lease another vertex of K, i.e. K intersects the 0 level set of ϕ;
- ψ takes a negative value in at least one of the vertices of K, i.e. K lies inside or on the boundary of the negative subdomain of ψ.

We then travel the set $\mathcal{K}$ and for each $K \in \mathcal{K}$, we detect the intersection of the isosurface $\{\phi = 0\}$ with the edges of K from linear interpolation of the values of ϕ at its vertices. We then use a pattern to split the simplices sharing at least one of the identified edges by this process in such a way that Γ appears explicitly in the resulting mesh. Note that this operation concerns not only the simplices of $\mathcal{K}$, but also those elements in

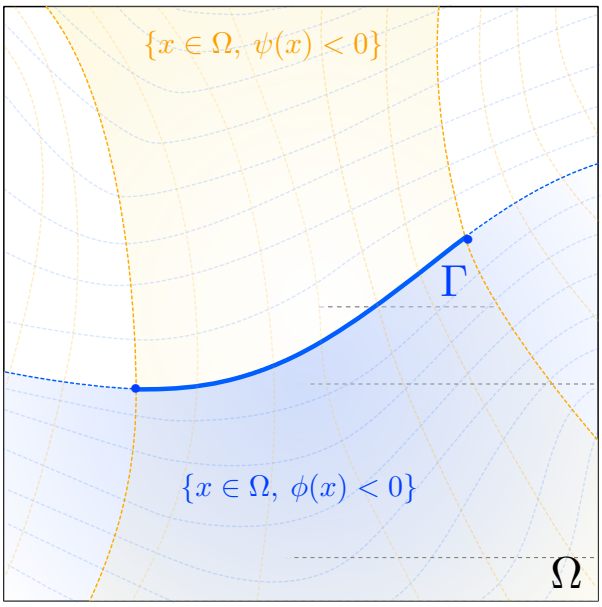


FIGURE 8. Description of a 2d fracture $\Gamma \subset \Omega$ by means of two level set functions $\phi, \psi : \Omega \to \mathbb{R}$ in the sense of (5.1).

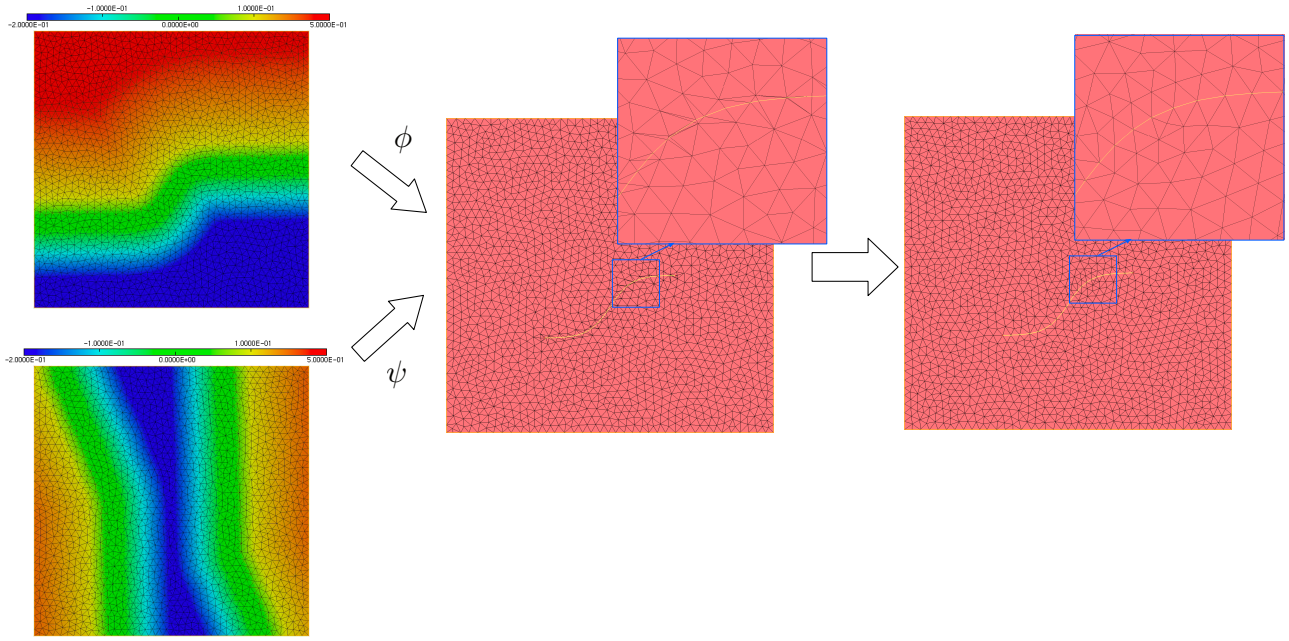


FIGURE 9. Illustration of the body-fitted fracture meshing based on a two level set function representation described in Section 5; (left) Two level set representation of a fracture $\Gamma \subset \Omega$; (middle) Poor-quality mesh $\mathcal{M}_{\text{temp}}$ resulting from the rough discretization stage described in Section 5.1; (right) High-quality mesh $\mathcal{M}$ resulting from the remeshing procedure described in Section 5.2.

$\mathcal{M}$ that are adjacent to those in $\mathcal{K}$, whose splitting is needed to ensure conformity of the resulting mesh in spite of the fact that they do not intersect the fracture Γ.

This stage results in a mesh $\mathcal{M}_{\text{temp}}$ of Ω, which is valid, conforming, and contains an explicit discretization of Γ as a collection $\mathcal{S}_{\text{temp}}$ of sub-simplices. Unfortunately, $\mathcal{M}_{\text{temp}}$ will likely be of poor quality – *i.e.*, contain very flat, nearly degenerate elements – since the relative positions of Γ and the vertices of the elements of $\mathcal{K}$ are arbitrary, see Figure 9 (b).

5.2. **Remeshing of the resulting situation**

In this second stage, we repeatedly apply four local mesh modification operators, in order to gradually improve the quality of $\mathcal{M}_{\text{temp}}$.

- *Edge split:* A "long" edge in the mesh is split into two, after addition of a new vertex in the mesh. All the simplices sharing this edge are split accordingly.
- *Edge collapse:* One of the two endpoints of a "short" edge is merged onto the other, and the attached simplices are updated accordingly.
- *Edge swap:* An edge is suppressed from the mesh and the whole "shell" of elements sharing this edge (that is, two triangles in 2d, and a more complex collection of tetrahedra in 3d) is retriangulated.
- *Vertex relocation:* One vertex is moved, while all the connectivities in the mesh remain untouched.

We refer to classical textbook about meshing such as [17, 53] for a more detailed presentation of these operations. The output of this stage is the desired high-quality mesh $\mathcal{T}$ of Ω, in which the fracture Γ is explicitly discretized, as a collection $\mathcal{S}$ of sub-simplices, see Figure 9 (c).

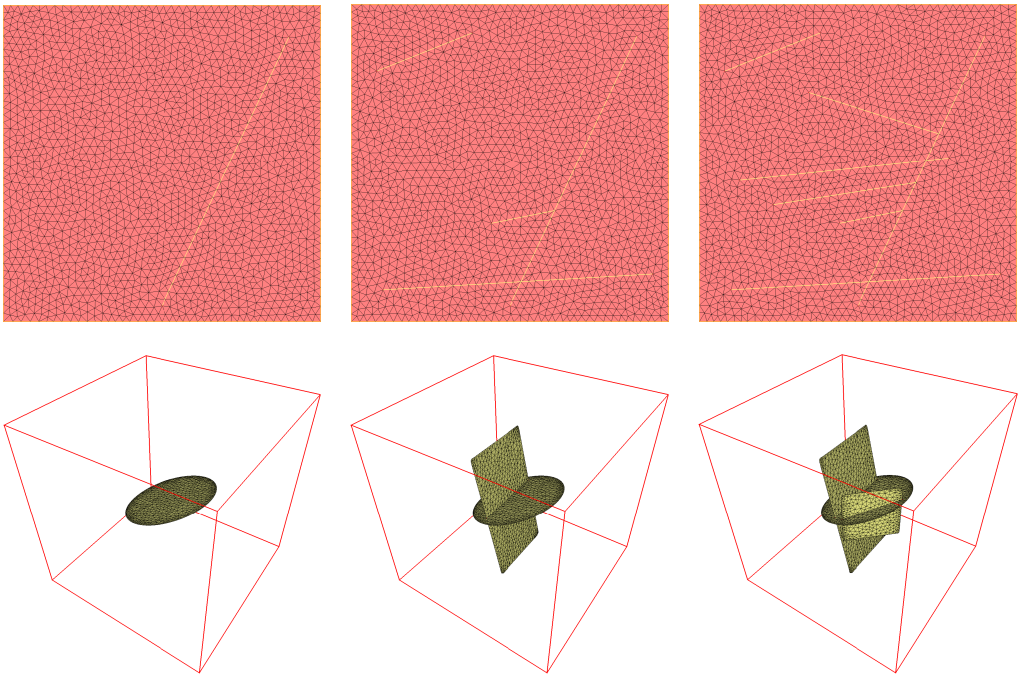


FIGURE 10. (Upper row) Repeated insertion of manifold curves into an ambient mesh of Ω, resulting in a non-manifold fractured configuration; (lower row) Successive insertion of three open surfaces into an ambient mesh of a three-dimensional domain Ω (for vizualization purpose, only the sub-simplices of the fracture set are displayed).

6. Details about the numerical implementation

This section discusses a few implementation details of the previously introduced methods dedicated to the resolution of the variational problem (3.1). In Section 6.1, we describe the practical construction of the cannot-link constraints attached to a fractured geometry defined in (4.1). Then, in Section 6.2, we illustrate the execution of our strategy with the third-party finite element solver `FreeFem`.

6.1. Numerical computation of the cannot-link constraints for the mesh partitioning

6.1.1. *Two-dimensional case*

By definition, the cannot-link pairs $\{K, K'\} \in \mathcal{CL}$ of a 2d mesh $\mathcal{M}$ enclosing a fracture Γ are made of elements K, K' that belong to different generalized sub-simplices of a common sub-simplex $S \in \sigma(\mathcal{M})$. These pairs are thus of two types:

- S is an edge of Γ, and K and K' are the two elements that contain this edge.
- S is a vertex of Γ, and K and K' belong to two different components of the star $\mathrm{st}_{\mathcal{M}}(S)$.

With a small abuse of notation, let $\partial\Gamma$ denote the set of vertices on the boundary of the fracture Γ, i.e. those vertices that only belong to one edge of Γ. For any edge $e \in \Gamma$, the generalized components of $\mathrm{st}_{\mathcal{M}}(e)$ are included in those of $\mathrm{st}_{\mathcal{M}}(V)$ for either of its vertices V that is not on $\partial\Gamma$. Thus, the edges e of Γ generating cannot-link pairs that are not already accounted for by generalized vertices are exactly the "isolated edges", *i.e.*, edges with both ends in $\partial\Gamma$, see Figure 11.

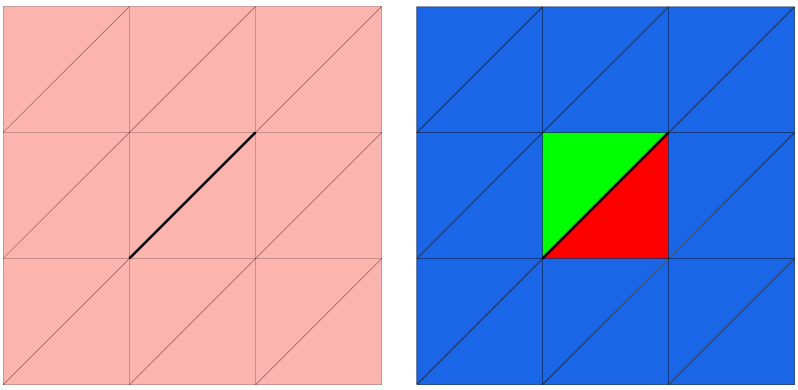


FIGURE 11. (Left) Mesh $\mathcal{M}$ with a fracture Γ consisting of a single internal edge; (right) Partition of this mesh fulfilling the property (P).

As a result, in two space dimensions, the identification of the cannot-link pairs of elements of a fractured mesh $\mathcal{M}$ is straightforward: $\mathcal{CL}$ is the set of all pairs $\{K, K'\}$ of elements belonging to different generalized components of the vertices V on Γ, augmented with the pairs of triangles adjacent to the isolated edges of Γ.

6.1.2. *Three-dimensional case*

Let $\mathcal{M}$ be a three-dimensional mesh and, like in the previous Section 6.1.1, let $\partial\Gamma$ be the set of edges that lie in only one triangle of the fracture Γ. The cannot-link pairs of tetrahedra $\{K, K'\}$, induced by sub-simplices $S \in \sigma(\mathcal{M})$, are of three different types:

- S is a triangle in the fracture Γ, and K and K' are the two elements of $\mathcal{M}$ sharing this face;
- S is an edge of Γ and K and K' belong to two different components of the star $\mathrm{st}_{\mathcal{M}}(S)$;
- S is a vertex of Γ and K and K' belong to two different components of the star $\mathrm{st}_{\mathcal{M}}(S)$.

The computation of the cannot-link pairs associated to generalized vertices or triangles is straightforward. To reduce the computational complexity, one may try to reduce to a minimum the number of edges considered for generating cannot-link constraints. Although not strictly needed in the following, the result of this proposition conveniently allows to eliminate the contributions from generalized edges which do not induce cannot-link constraints other than those already induced by the generalized vertices.

Proposition 6.1. *Let $\mathcal{M}$ be a regular mesh and let $e = \{V, W\} \in \sigma_1(\mathcal{M}) \subset \Gamma$ be an edge of $\mathcal{M}$ that belongs to the fracture. Then,*

(i) The star $\mathrm{st}_{\mathcal{M}}(e)$ admits at least two generalized components, that are included in those of $\mathrm{st}_{\mathcal{M}}(V)$ provided one of the following conditions is satisfied:
  (1) V is not part of the boundary of the mesh or that of the fracture: $V \notin \partial\mathcal{M}$ and $V \notin \partial\Gamma$.
  (2) $V \in \partial\mathcal{M}$ but all the edges e of $\mathcal{M}$ emanating from V that belong to $\partial\Gamma$ also belong to $\partial\mathcal{M}$, i.e., *for any edge $e' \in \sigma_1(\mathcal{M})$ with endpoints V and W', $e' \subset \partial\Gamma \Rightarrow W' \in \partial\mathcal{M}$.*

(ii) The star $\mathrm{st}_{\mathcal{M}}(e)$ has only one component if $e \subset \partial\Gamma$ with $V \notin \partial\mathcal{M}$ but $V \in \partial\Gamma$, or if $V \in \partial\mathcal{M}$ but $W \notin \partial\mathcal{M}$.

Proof. We present an argument based on graph theory, which is illustrated in Figure 12.

(i). Since the mesh $\mathcal{M}$ is regular, the star $\mathrm{st}_{\mathcal{M}}(V)$ is topologically equivalent to a ball if $V \notin \partial\mathcal{M}$ and to a half-ball if $V \in \partial\mathcal{M}$ [94, p. 24]. Consider a small sphere centered around v intersecting the faces adjacent to v.

Let us assume that (1) holds true. We consider the "shell" of V obtained as the intersection of the tetrahedra having V as vertex with a sufficiently small sphere centered at V. Since $V \notin \partial\mathcal{M}$, this intersection generates a planar graph with the topology of the sphere, whose edges and nodes are respectively the faces and the edges of these tetrahedra that contain V. Moreover, the graph of the tetrahedra containing V is precisely the dual of this graph. The edge e is represented on this graph as a single vertex on the sphere, and the fracture is a set of edges. Since V does not belong to $\partial\Gamma$, then the set of edges associated to Γ on the sphere has no boundary, see Figure 12 (top). Consequently, the connected components of the sub-dual graph generated by $\mathrm{st}_{\mathcal{M}}(e)$, which are in correspondance with the edges emanating from e in this sphere, must be included in those of $\mathrm{st}_{\mathcal{M}}(V)$. If instead, $V \in \partial\Gamma$ and $e \subset \partial\Gamma$, it is clear that the star of e has only one component because e is then incident to only one "edge" of Γ on the shell.

Let us now assume that (2) holds true. If $V \in \partial\mathcal{M}$ and $\mathcal{M}$ is regular, then the "shell" of V defined as above has the topology of a half sphere. Since the condition (2) is satisfied, the boundary of Γ intersects only the flat part of the boundary of the half-sphere. Then, it is clear that $\mathrm{st}_{\mathcal{M}}(e)$ admit at least two components that are subsets of the components of $\mathrm{st}_{\mathcal{M}}(V)$ because there is no path between tetrahedra containing e and pertaining to different components of $\mathrm{st}_{\mathcal{M}}(e)$ that do not intersect Γ.

(ii). The star of e has only one component if $e \subset \partial\Gamma$ with $W \notin \partial\mathcal{M}$. □

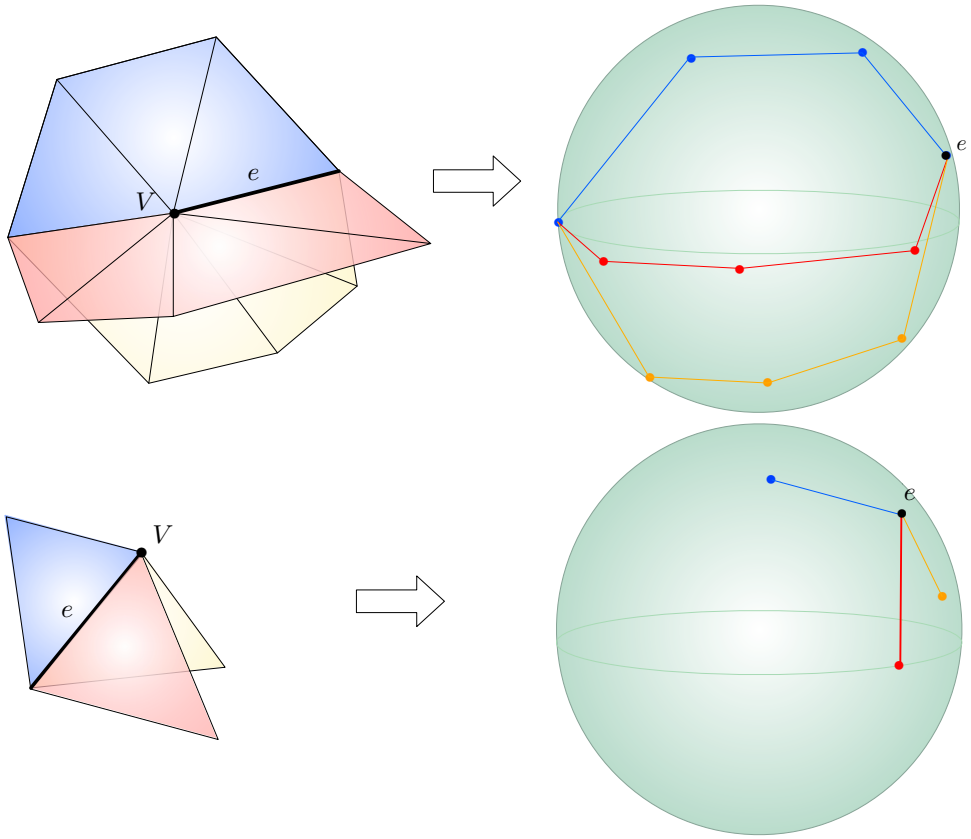


FIGURE 12. Graph interpretation of the generalized components of edges and vertices used in the proof of Proposition 6.1: the components of $\mathrm{st}_{\mathcal{M}}(V)$ are accounted for by the disjoint regions delimited by the paths on the sphere, and those of $\mathrm{st}_{\mathcal{M}}(e)$ correspond to the edges connected to e. (Upper situation) $V \notin \partial\Gamma$: the components of $\mathrm{st}_{\mathcal{M}}(e)$ are accounted for by $\mathrm{st}_{\mathcal{M}}(V)$; (Lower situation) $V \in \partial\Gamma$ and $e \not\subset \partial\Gamma$: $\mathrm{st}_{\mathcal{M}}(V)$ has only one component, while $\mathrm{st}_{\mathcal{M}}(e)$ has three.

This proposition implies that the edges e of $\mathcal{M}$, with endpoints V, W, that generate cannot-link constraints not already accounted for by generalized vertices, are the following

(i) The edges with both endpoints in $\partial\Gamma$, but not lying in $\partial\Gamma$: $V \in \partial\Gamma$ and $W \in \partial\Gamma$ are both internal vertices of $\mathcal{M}$ (i.e. $V, W \notin \partial\mathcal{M}$) and $e \not\subset \partial\Gamma$, see Figure 13.
(ii) The edges $e \not\subset \partial\Gamma$ with vertices $V \in \partial\mathcal{M}$, $W \notin \partial\mathcal{M}$ but $W \in \partial\Gamma$ (or, symmetrically $W \in \partial\mathcal{M}$, $V \notin \partial\mathcal{M}$ and $V \in \partial\Gamma$);
(iii) The edges with vertices $V \in \partial\mathcal{M}$, $W \in \partial\mathcal{M}$ where both V and W do not satisfy the condition (2) in Proposition 6.1.

All three situations are exceptional. They only occur when the fracture has thin parts wide of only one edge that may or may not touch the boundary $\partial\mathcal{M}$ of $\mathcal{M}$: case (i), (ii) and (iii) correspond respectively to the cases where $V, W \notin \partial\mathcal{M}$, $V \in \partial\mathcal{M}$ but $W \notin \partial\mathcal{M}$ (or symmetrically $W \in \partial\mathcal{M}$ but $V \notin \partial\mathcal{M}$) , and $v, w \in \partial\mathcal{M}$. The situation (i) is represented on Figure 13. In our implementation, we compute the stars of all edges having one vertex on either the boundary of the fracture or of the mesh $\partial\mathcal{M}$, which is slightly suboptimal, but robust to cases where the mesh would not be regular while still involving only a moderate number of edges (proportional to the number of vertices on the boundary of Γ). We note, however, that the computation of *all* generalized edges is still required when constructing the mapping α_p and associated restriction operators in (3.13) and (3.14) required for the assembly step of the Finite Element matrix as described in the next subsection. Indeed, these are needed to properly assign duplicated degrees of freedom on edges to their relevant components on the partition $\mathcal{M} = \cup_{p=0}^{n} \mathcal{M}_p$.

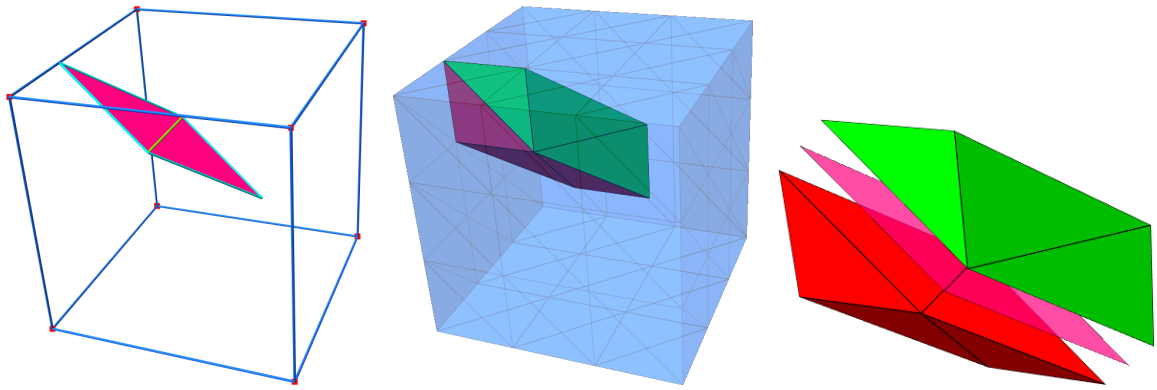


FIGURE 13. "Degenerate" fracture featuring an internal edge (in green on the left plot) e that is not part of the boundary $\partial\Gamma$ although connecting two vertices $V, W \in \partial\Gamma$. The middle plot shows the partitioning obtained with our method and the third plot displays an "exploded" view of the situation.

6.2. Assembly of restriction operators and practical implementation using `FreeFem` as third party software

The procedure for assembling the finite element matrix and right-hand side of our variational problem (3.1) is summarized in Algorithm 2. Our numerical implementation relies on the software `FreeFem` for assembling the finite element matrices of the local subproblems (3.11). In this environment:

- The positions of the $\mathbb{P}_k$ Lagrange nodes $\{x_1, \dots, x_N\}$ of the finite element space $V^k(\mathcal{M})$ required in step 3 are identified by computing the decomposition of the identity mapping on the basis $\{\phi_1, \dots, \phi_N\}$ of this space; indeed for $i = 1, \dots, d$, this decomposition reads:

$$\forall x \in |\mathcal{M}|, \quad x = \sum_{k=1}^{N} \alpha_k \phi_k(x), \text{ for some coefficients } \alpha_1, \dots, \alpha_N \in \mathbb{R}^d.$$

  Hence, because $\phi_k(x_i) = 1$ if $i = k$ and 0 otherwise, it holds that $x_i = \alpha_i$.
- The computation of the mappings α_p at step 5, associating to a Lagrange node x_i in the local mesh $\mathcal{M}_p$ the unique component number $\alpha_p(i)$ such that $\mathrm{st}_{\mathcal{M}_p}(x_i) \subset \mathrm{st}_{\mathcal{M}}(x_i; \alpha_p(i))$, requires restriction operators for passing from the numbering of the total finite element space $V^k(\mathcal{M})$ to the local ones $V^k(\mathcal{M}_p)$, $p = 0, \dots, n$. These restriction operators are obtained with the command `restrict` in `FreeFem`.
- The local matrices $\mathbf{A}_p$ and $\mathbf{b}_p$ are easily assembled with `FreeFem`, by implementing their associated variational formulation as for a problem without fracture. The knowledge of the partition enables to impose easily different boundary conditions on opposite sides of the fracture, for instance a Dirichlet boundary condition on one side and a Neumann one on the other side, see Section 7.1 below for an example.
- The global linear system $\mathbf{A}x = \mathbf{b}$ is solved with the sparse solver `umfpack` [41] for the 2d examples and `mumps` [3] for the 3d cases.

Algorithm 2 Construction of the global finite element matrix on a mesh $\mathcal{M}$ with fracture Γ.

Require: Mesh $\mathcal{M}$ with a fracture $\Gamma \subset \sigma_{d-1}(\mathcal{M})$ given as a set of faces.

1: Compute generalized vertices and generalized edges (in 3d)
2: Partition the mesh using Algorithm 1.
3: Identify the degrees of freedom x_i of the space $V^k(\mathcal{M})$ of order k conforming Lagrange finite elements. Identify those nodes that are located at vertices, edges or faces (in 3d) of the fracture.
4: Create an index numbering $(i_1, \alpha_1), \dots (i_{\mathcal{Q}}, \alpha_{\mathcal{Q}})$ for the generalized degrees of freedom $(i, \alpha) \in \mathcal{I}$.
5: Compute the mappings α_p in (3.12), and the restriction matrices $\mathcal{R}_p$ in (3.12), $0 \le p \le n$.
6: Use a third-party software for assembling the local matrices $\mathbf{A}_p$ and right-hand side $\mathbf{b}_p$.
7: Assemble the global finite element matrix $\mathbf{A}$ and right-hand side $\mathbf{b}$ using (3.16).

Let us eventually comment about the vizualization of the final fractured solution. To achieve this, we construct the mesh $\mathcal{M}_{\text{cut}}$ where all the faces along the fracture are "cut", by replicating each vertex on Γ according to the number of components of the associated generalized vertex. More precisely, the set of vertices of $\mathcal{M}_{\text{cut}}$ is:

$$\sigma_0(\mathcal{M}_{\text{cut}}) = \{(V, \alpha) \,|\, V \in \sigma_0(\mathcal{M}), \quad 1 \le \alpha \le n_{V_i}\},$$

where for a given V, the vertices (V, α) are physically located at V although topologically distinct. The set of elements of $\mathcal{M}_{\text{cut}}$ is:

$$\sigma_d(\mathcal{M}_{\text{cut}}) = \left\{ \left\{ (V_1, \alpha_1), \dots, (V_{d+1}, \alpha_{d+1}) \right\} \,\middle|\, K = \{V_1, \dots, V_{d+1}\} \in \sigma_d(\mathcal{M}), \right.$$
$$\left. \alpha_i \in \{1, \dots, n_{V_i}\} \text{ is the unique index s.t. } K \in \text{st}_{\mathcal{M}}(V_i; \alpha_i) \text{ for } i = 1, \dots, d+1 \right\},$$

i.e., elements are disconnected along the fracture. We then assign the values of the solution at the vertices of $\mathcal{M}_{\text{cut}}$, which enables to plot it as a $\mathbb{P}_1$ function.

Remark 6.1. As mentioned in the introduction, one simple idea to construct $\mathbf{A}$ and $\mathbf{b}$ would consist in calling the third-party software to assemble them on the mesh $\mathcal{M}_{\text{cut}}$. However, common third-party softwares often assume that the mesh should be regular. This is not a requirement, however, from visualization libraries such as `matplotlib` [64] in 2d, and `pyvista` [101] in 3d, which allow to plot discontinuous $\mathbb{P}_1$ fields.

7. Numerical examples

This section presents a few 2d and 3d numerical experiments, illustrating the main features and the versatility of the proposed method for solving boundary value problems in fractured geometries. In Section 7.1, a two-dimensional Laplace equation is solved on a domain with a complex fracture to assess the accuracy of high-order finite elements. The second example, proposed in Section 7.2, considers a steady-state advection equation involving discontinuities across the skeleton of an ellipse to illustrate the ability of the method to handle non-standard variational problems. Finally, Section 7.3 deals with relatively large-scale 2d and 3d acoustic scattering problems featuring multi-screen geometries.

7.1. Two-dimensional Laplace equation

This first example features the Laplace equation:

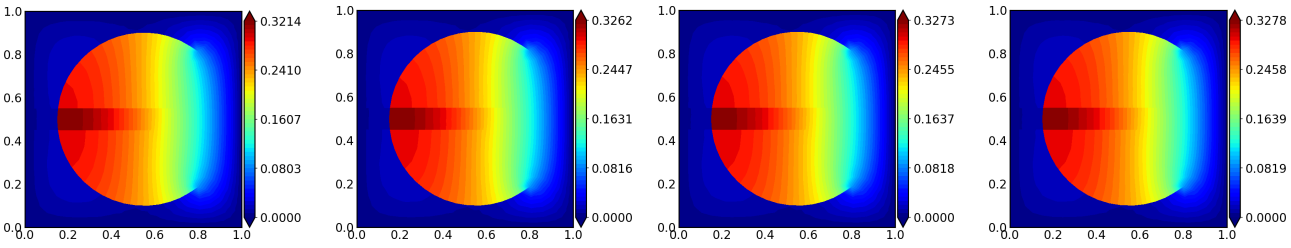
$$\begin{cases} -\Delta u = 1 \text{ in } \Omega \backslash \Gamma, \\ u = 0 \text{ on } \partial\Omega, \\ \dfrac{\partial u}{\partial n} = 0 \text{ on } \Gamma, \end{cases} \tag{7.1}$$

where Ω is the unit square $(0,1)^2$ in $\mathbb{R}^2$ and the fracture Γ is the €-shaped collection of curves depicted in Figure 1a. The variational form of this problem reads:

$$\text{Find } u \in H^1_0(\Omega \backslash \Gamma) \text{ such that } \int_{\Omega \backslash \Gamma} \nabla u \cdot \nabla v \, \mathrm{d}x = \int_{\Omega \backslash \Gamma} f v \, \mathrm{d}x \quad \forall v \in H^1_0(\Omega \backslash \Gamma), \tag{7.2}$$

where $H^1_0(\Omega \backslash \Gamma) := \{ v \in H^1(\Omega \backslash \Gamma) \,|\, v = 0 \text{ on } \partial\Omega \}$.

In order to solve (7.1) or (7.2), we use the mesh $\mathcal{M}$ of Figure 5, conforming to this fractured geometry, together with the partitioning algorithm of Section 3 to assemble the finite element matrices $\mathbf{A}$ and $\mathbf{b}$ associated with the $\mathbb{P}_k$ Lagrange finite element method, for $k \in \{1, 2, 3, 4\}$. The projections of the solutions onto the space of $\mathbb{P}_1$ functions on the cut mesh $\mathcal{M}_{\text{cut}}$ are displayed in Figure 14; a 3d surface plot of the solution obtained with $\mathbb{P}_4$ elements is represented in Figure 15, see Section 6.2 and Remark 6.1 for details about the vizualization of functions on fractured geometries. The impact of the finite element order is illustrated in Figure 16: the numerical error between these different functions and a reference $\mathbb{P}_4$ solution computed on a very fine mesh decreases significantly as the finite element order k increases.



(A) $\mathbb{P}_1$ (B) $\mathbb{P}_2$ (C) $\mathbb{P}_3$ (D) $\mathbb{P}_4$

Figure 14. Solutions to the Laplace problem (7.1) on the €-shaped fractured geometry of Figure 5, obtained with Lagrange finite element methods of various orders.

Finally, we solve the variant of (7.1) where the inner sides of the two central bars of the € symbol (pertaining to the component labeled 6 on Figure 5a) bear homogeneous Dirichlet conditions while the other sides of these

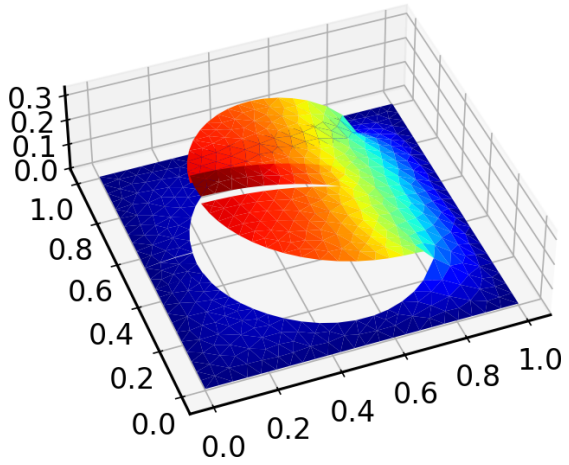
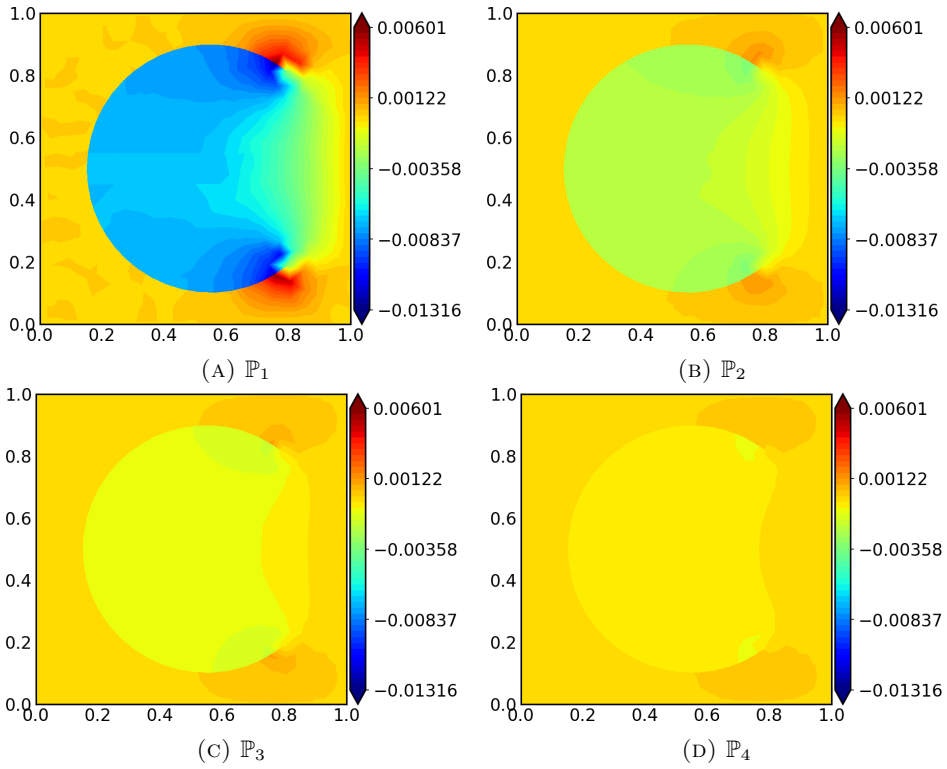


FIGURE 15. Graph of the $\mathbb{P}_4$ solution of the Laplace problem (7.1) on the ∈-shaped fractured geometry of Figure 5.



(A) $\mathbb{P}_1$ (B) $\mathbb{P}_2$

(C) $\mathbb{P}_3$ (D) $\mathbb{P}_4$

FIGURE 16. Plots of the error between the $\mathbb{P}_k$ solution and a reference solution computed on a finer mesh for the Laplace problem (7.1) on the ∈-shaped fractured 2d domain of Figure 5.

bars and the other regions of Γ bear homogeneous Neumann conditions; the solution is displayed on Figure 17. Our domain decomposition strategy makes it easy to implement this change in boundary conditions, by simply modifying the assembly of the local matrix of the component labeled 6 with the chosen third-party software.

7.2. **Steady-state advection along the normal rays to an ellipse**

The example of the previous Section 7.1 features a "standard" Laplace equation, where the coefficients of the local finite element matrices could relatively easily be implemented "by hand". In order to illustrate how our approach adapts to the assembly of more complex variational problems, we consider a steady-state advection equation which was originally motivated by the topic of shape optimization in our previous work [49].

Let D be the 2d square $D = (-1/2, 1/2)^2$ enclosing the ellipse

$$\Omega = \left\{ (x_1, x_2) \in \mathbb{R}^2 \,|\, \frac{x_1^2}{a^2} + \frac{x_2^2}{b^2} < 1 \right\}, \text{ where } a = 0.4 \text{ and } b = 0.2. \tag{7.3}$$

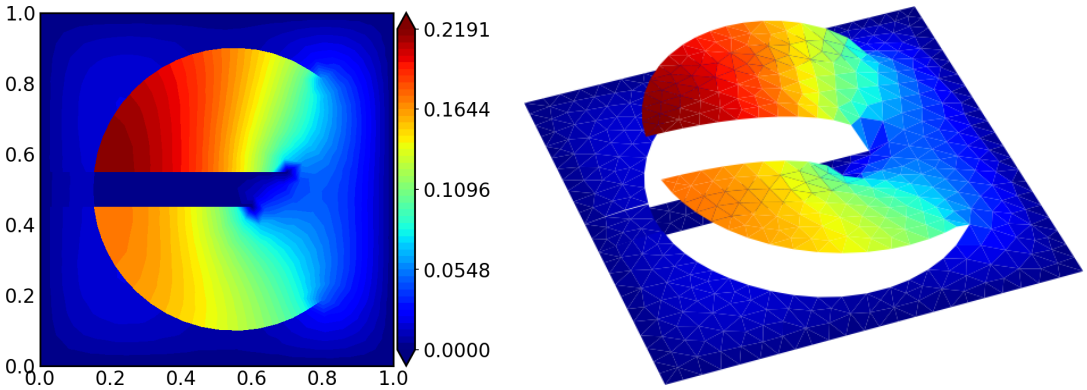


FIGURE 17. Numerical solution of problem (7.1), subject to a homogeneous Dirichlet boundary condition imposed on the portion of Γ lying in the component 6 in Figure 5a.

In this example, the fracture Γ is the "skeleton", or medial axis of Ω, i.e., the closure of the set of points $x \in D$ that are equidistant to at least two distinct points on $\partial\Omega$ (see e.g.[42]). Here, Γ reads explicitly:

$$\Gamma = \{(x_1, 0) \,|\, |x_1| \leq a - b^2/a\}.$$

The domain D is equipped with a mesh $\mathcal{M}$ enclosing a discretization of the boundary $\partial\Omega$ of the ellipse and its skeleton Γ as edges (see Figure 18).

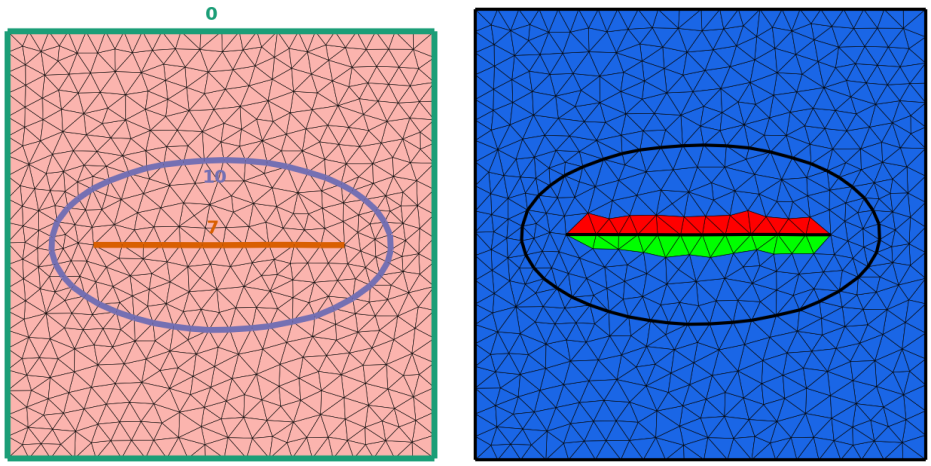


FIGURE 18. (Left) Mesh $\mathcal{M}$ of the domain D enclosing the boundary of the ellipse Ω in (7.3) (labeled "10") and its skeleton Γ (labeled "7") as subsets of edges; (Right) Partition of $\mathcal{M}$ obtained by Algorithm 1 considering Γ as the fracture.

Let us recall a few useful definitions and facts for this section, referring to [42] for further details:

- The signed distance function $d_\Omega : D \to \mathbb{R}$ to the domain Ω is defined by:

$$d_\Omega(x) = \begin{cases} -d(x, \partial\Omega) \text{ if } x \in \Omega, \\ d(x, \partial\Omega) \text{ if } x \in D\backslash\Omega, \end{cases} \quad \text{where} \quad d(x, \partial\Omega) := \inf_{y \in \partial\Omega} |x - y| \text{ is the distance from } x \text{ to } \partial\Omega.$$

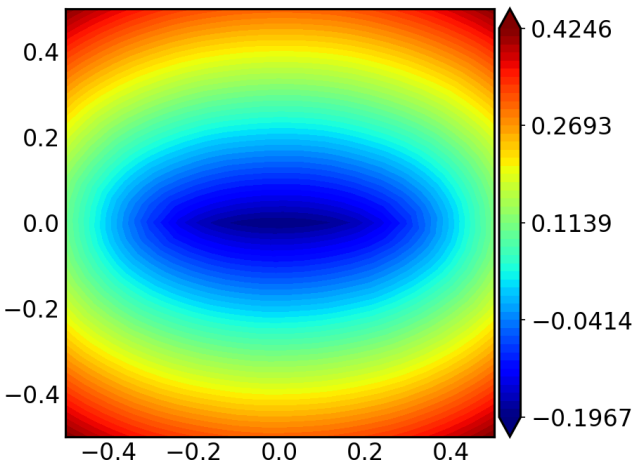
- For any point $x \in D$ outside the skeleton Γ, there exists a unique point $y = p_{\partial\Omega}(x) \in \partial\Omega$ realizing the minimum in the above definition of $d(x, \partial\Omega)$, which is called the projection of x onto $\partial\Omega$.
- The signed distance function d_Ω is differentiable on $D\backslash\Gamma$ and its gradient ∇d_Ω is the extension of the unit normal vector n to $\partial\Omega$ (pointing outward) along the normal rays to $\partial\Omega$:

$$\forall x \in D\backslash\Gamma, \quad \nabla d_\Omega(x) = n(p_{\partial\Omega}(x)),$$

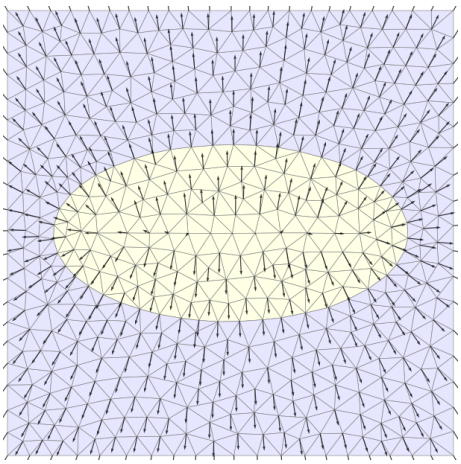
see Figure 19.

Let us now consider the following boundary-value problem

$$\begin{cases} \nabla d_\Omega \cdot \nabla u = 0 & \text{in } D\backslash\Gamma, \\ u = u_0 & \text{on } \partial\Omega, \end{cases} \tag{7.4}$$



(A) d_Ω



(B) ∇d_Ω

FIGURE 19. (Left) Values of the signed distance function d_Ω to the ellipse $\Omega \subset D$ in Figure 18; (right) Representation of its gradient vector field.

for the boundary datum $u_0(x_1, x_2) = x_2$. This problem is a steady-state advection equation whose solution is explicitly given by the formula:

$$u(x) = u_0(p_{\partial\Omega}(x)),$$

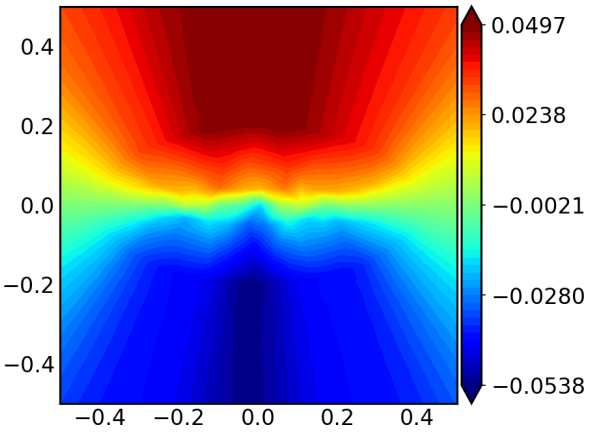
i.e. u is obtained by extending the values of u_0 constantly along the normal rays to $\partial\Omega$. It is discontinuous across the skeleton Γ, where the projection $p_{\partial\Omega}$ is not defined. In other words, different characteristic curves of the vector field ∇d_Ω intersect at Γ, which causes the solution u to be discontinuous.

The problem (7.4) has the following weak formulation:

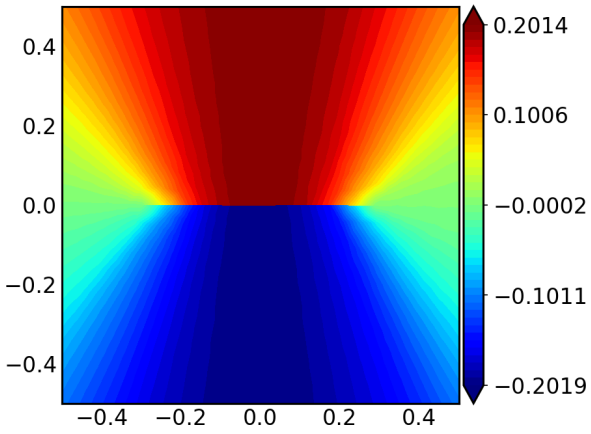
$$\text{Find } u \in V \text{ such that } \int_{\partial\Omega} uv \,\mathrm{d}s + \int_{D\setminus\Gamma} (\nabla d_\Omega \cdot \nabla u)(\nabla d_\Omega \cdot \nabla v) \,\mathrm{d}x = \int_{\partial\Omega} u_0 v \,\mathrm{d}s, \text{ for all } v \in V, \tag{7.5}$$

with the functional space $V := \{v \in L^2(D\setminus\Gamma) \,|\, \nabla d_\Omega \cdot \nabla v \in L^2(D\setminus\Gamma)\}$, see [49]. A "naive" finite element resolution of (7.5) by discretization with $\mathbb{P}_1$ finite elements on the mesh $\mathcal{M}$, without duplication of the degrees of freedom lying on Γ, is plotted in Figure 20a. This solution fails to capture the discontinuity of u on Γ, resulting in large errors throughout the computational domain D. In our previous work [49], this issue was alleviated either by extruding a thin rectangular hole enclosing Γ from D or by adding a weight to the vector field ∇d_Ω that vanishes near Γ in the variational formulation (7.5).

The solution obtained by assembling the finite element matrix on the fractured mesh with Algorithm 2 is shown in Figure 20b; the partition of the mesh produced by Algorithm 1 is plotted in Figure 18. Our approach enables the solution to exhibit the expected discontinuous behavior at Γ and to be accurate in the whole computational domain D, even in the immediate neighborhood of Γ.



(A) "Naive" solution.



(B) "Fractured" solution.

FIGURE 20. Numerical solutions to (7.4) obtained with the finite element method. (Left) "Naive" solution to (7.5) obtained by using regular $\mathbb{P}_1$ finite elements functions (that are continuous on Γ) on the mesh $\mathcal{M}$ in Figure 18; (right) "Fractured" solution obtained by duplicating the degrees of freedom along Γ.

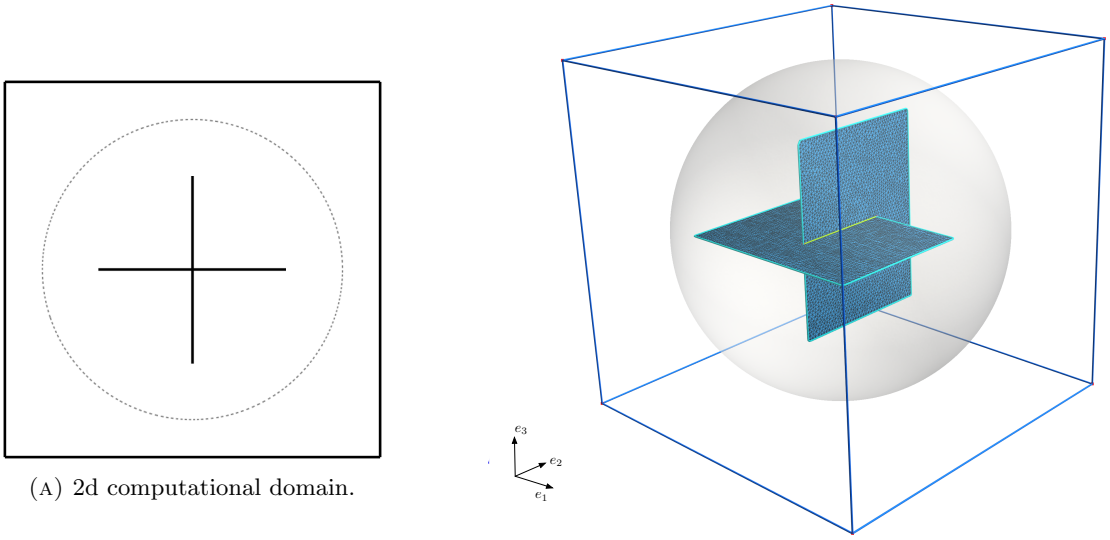
The assembly of problem (7.5) is relatively straightforward with our partition strategy: we first compute the signed distance function d_Ω as a $\mathbb{P}_1$ function on the mesh $\mathcal{M}$ thanks to the open-source library `mshdist` [39]. Then, the values of d_Ω are projected onto each submesh $\mathcal{M}_p$, $0 \leq p \leq n$ and the gradient ∇d_Ω is evaluated on every mesh component to assemble (7.5).

7.3. 2d and 3d acoustic scattering by a multi-screen

Our third application deals with acoustic scattering in the presence of a singular geometry. An incident plane wave $u_{\text{in}}(x)$ is scattered by a "multi-screen" structure $\Gamma \subset \mathbb{R}^d$, which is the reunion of two $(d-1)$-dimensional panels. We consider one two-dimensional and one three-dimensional situation:

- for $d = 2$, $\Gamma = \Gamma_1 \cup \Gamma_2$ with $\Gamma_1 := \{0.5\} \times (0.25, 0.75)$ and $\Gamma_2 := (0.25, 0.75) \times \{0.5\}$;
- for $d = 3$, $\Gamma = \Gamma_1 \cup \Gamma_2$ with $\Gamma_1 := \{0.5\} \times (0.3, 0.9) \times (0.1, 0.9)$ and $\Gamma_2 := (0.1, 0.9) \times (0.1, 0.7) \times \{0.5\}$.

These multi-screens are embedded in square and cubic computational domains, respectively $D = (0,1)^2$ in 2d and $D = (0, 1.4)^3$ in 3d, see Figure 21.



(A) 2d computational domain.

(B) 3d computational domain.

FIGURE 21. Computational domains enclosing two- and three-dimensional multi-screens for the scattering problem of Section 7.3. The boundary of the absorbing PML region is represented by a gray dotted line in the 2d case and a transparent gray sphere in the 3d case.

We denote by $\Sigma = \Gamma_1 \cap \Gamma_2$ the intersection between both panels: Σ is a point when $d = 2$, and a segment when $d = 3$. Each panel Γ_i $(i = 1, 2)$ is an open Lipschitz surface equipped with a unit normal vector $n_i : \Gamma_i \to \mathbb{R}^d$ defining its orientation. In the cases under scrutiny, denoting by $\{e_i\}_{i=1,\dots,d}$ the canonical basis of $\mathbb{R}^d$, we set:

- $n_1 = e_1$ and $n_2 = e_2$ if $d = 2$,
- $n_1 = e_1$ and $n_2 = e_3$ if $d = 3$.

For a function $v : \mathbb{R}^d \backslash \Gamma \to \mathbb{R}$ which is sufficiently smooth from each side of Γ_1 and Γ_2, we define the two-sided traces on each panel $\Gamma_i \backslash \Sigma$ by:

$$v_\pm(x) = \lim_{t \to 0^+} v(x \pm t n_i), \qquad x \in \Gamma_i \backslash \Sigma,$$

and likewise, we define the two-sided normal derivatives on $\Gamma_i \backslash \Sigma$ by:

$$\left.\frac{\partial v}{\partial n_i}\right|_\pm (x) := \lim_{t \to 0^+} n_i(x) \cdot \nabla v(x \pm t n_i), \qquad x \in \Gamma_i \backslash \Sigma. \tag{7.6}$$

The multi-screen Γ receives an incident wave u_{in} which solves the Helmholtz equation in $\mathbb{R}^d$:

$$\Delta u_{\text{in}} + k^2 u_{\text{in}} = 0 \text{ in } \mathbb{R}^d,$$

where $k = \frac{2\pi}{\lambda} > 0$ is the wave number, depending on the wave length λ. Specifically, $u_{\text{in}}(x) := e^{\mathrm{i}\frac{2\pi}{\lambda}\theta \cdot x}$ where θ is the direction of the wave; in our applications, θ and λ are set to:

- $\theta = \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right)$ and $\lambda = 0.05$ in 2d;
- $\theta = \left(\frac{1}{2}, \frac{1}{2}, \frac{1}{\sqrt{2}}\right)$ and $\lambda = 0.07$ in 3d.

The acoustic field u_s scattered by Γ is the solution to the following equation:

$$\begin{cases} \Delta u_s + k^2 u_s = 0 \text{ in } \mathbb{R}^d \backslash \Gamma, \\ \left.\dfrac{\partial u_s}{\partial n_i}\right|_+ = \left.\dfrac{\partial u_s}{\partial n_i}\right|_- = -\dfrac{\partial u_{\text{in}}}{\partial n_i} \quad \text{on } \Gamma_i \backslash \Sigma, \quad i = 1, 2, \\ \dfrac{\partial}{\partial |x|} u_s(x) - \mathrm{i}k u_s(x) = \mathrm{o}(|x|^{-(d-1)/2}) \text{ as } |x| \to +\infty. \end{cases} \tag{7.7}$$

Here, the Neumann boundary condition on $\Gamma_i \backslash \Sigma$ accounts for the hard reflection of the scattered wave: both one-sided normal derivatives are prescribed on each screen Γ_i, or equivalently, the total field $u = u_s + u_{\text{in}}$ satisfies the hard-wall condition $\left.\frac{\partial u}{\partial n_i}\right|_\pm = 0$ on $\Gamma_i \backslash \Sigma$ for $i = 1, 2$. The third equation in (7.7) is the Sommerfeld radiation condition, which imposes the outgoing behavior of the field far from the multi-screen. In this setting, the solution u_s to (7.7) has different values on different sides of the multi-screen Γ [32], which motivates the use of our decomposition method.

The numerical resolution of (7.7) is a priori intricate as it is posed in the infinite domain $\mathbb{R}^d$. To alleviate this issue, we reduce it to a bounded domain D (typically, a large enough cube) thanks to the Perfectly Matched Layer (PML) method, introduced in [13], following the implementation in spherical coordinates as in e.g. [33]. This method yields a cheap and accurate approximation of the outgoing solution to (7.7) [20]. For the convenience of the reader, a short derivation of the latter is recalled in Appendix A.

Let $\sigma : \mathbb{R}_+ \to \mathbb{R}$ be a smooth increasing function such that $\sigma(r) = 0$ for $0 \le r \le r_{\text{PML}}$, where r_{PML} is large enough so that Γ is included in the open ball $B(0, r_{\text{PML}})$ with center 0 and radius r_{PML}; we set $r_{\text{PML}} := 0.4$ in 2d and $r_{\text{PML}} = 0.65$ in 3d. The scattered field u_s in (7.7) is approximated by the complex-valued solution u_s^{PML} to:

$$\begin{cases} \operatorname{div}(A(x)\nabla u_s^{\text{PML}}) + \alpha^2(|x|)\beta(|x|)k^2 u_s^{\text{PML}} = 0 & \text{in } D \backslash \Gamma, \\ \left.\dfrac{\partial u_s^{\text{PML}}}{\partial n_i}\right|_+ = \left.\dfrac{\partial u_s^{\text{PML}}}{\partial n_i}\right|_- = -\dfrac{\partial u_{\text{in}}}{\partial n_i} & \text{on } \Gamma_i \backslash \Sigma, \quad i = 1, 2, \\ u_s^{\text{PML}} = 0 & \text{on } \partial D, \end{cases} \tag{7.8}$$

where the functions α and β are defined by:

$$\alpha(r) = 1 + \mathrm{i}\sigma(r), \text{ and } \beta(r) := 1 + \mathrm{i}(\sigma(r) + r\sigma'(r)),$$

and the PML tensor $A(x)$ is given by:

$$A(x) = \alpha(|x|)^{d-3}\beta(|x|)I + \left(\frac{\alpha(|x|)^{d-1}}{\beta(|x|)} - \alpha(|x|)^{d-3}\beta(|x|)\right)\frac{xx^T}{|x|^2}. \tag{7.9}$$

We use a polynomial profile for σ:

$$\sigma(r) = \frac{\sigma_0}{p}\left(\frac{r - r_{\text{PML}}}{R_d - r_{\text{PML}}}\right)^p \mathbf{1}_{r > r_{\text{PML}}}, \tag{7.10}$$

where $\sigma_0 := 10$, $p = 3$, and $R_d = \frac{1}{2}\operatorname{diam}(D)$. By construction, (7.8) reduces to the original Helmholtz operator in the inner region $B(0, r_{\text{PML}})$.

Introducing the functional space $V = \{v \in H^1(D\backslash\Gamma; \mathbb{C}) \mid v = 0 \text{ on } \partial D\}$, the weak formulation of (7.8) consists in finding $u_s^{\text{PML}} \in V$ such that for all $v \in V$:

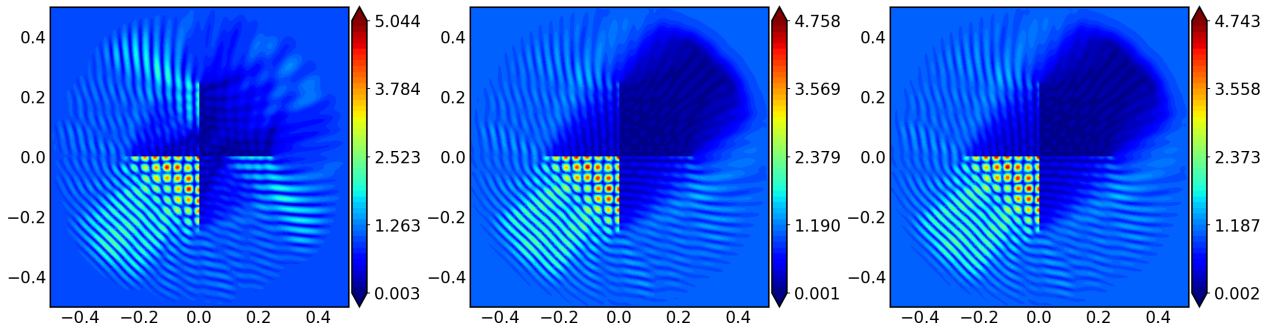
$$\int_D \left(A(x)\nabla u_s^{\text{PML}} \cdot \nabla \overline{v} - k^2\alpha^2(|x|)\beta(|x|)u_s^{\text{PML}}\overline{v}\right) \mathrm{d}x = \sum_{i=1,2}\int_{\Gamma_i} \frac{\partial u_{\text{in}}}{\partial n_i}(\overline{v}_+ - \overline{v}_-)\,\mathrm{d}s, \tag{7.11}$$

where the notation $\overline{v}$ stands for the conjugate of a complex-valued function v.

This problem is solved using "fractured" $\mathbb{P}_k$ elements according to our partitioning strategy. The assembly of the right-hand side is straightforward: for each subdomain $\mathcal{M}_p$, a unit normal n to $\Gamma \cap \partial\mathcal{M}_p$ pointing outward is computed. This normal has the opposite sign for the adjacent subdomain $\mathcal{M}_q$. The local contribution for $\mathcal{M}_p$ is thus the antilinear form $v \mapsto -\int_{\Gamma\cap\partial\mathcal{M}_p} \frac{\partial u_{\text{in}}}{\partial n}\overline{v}\,\mathrm{d}s$.

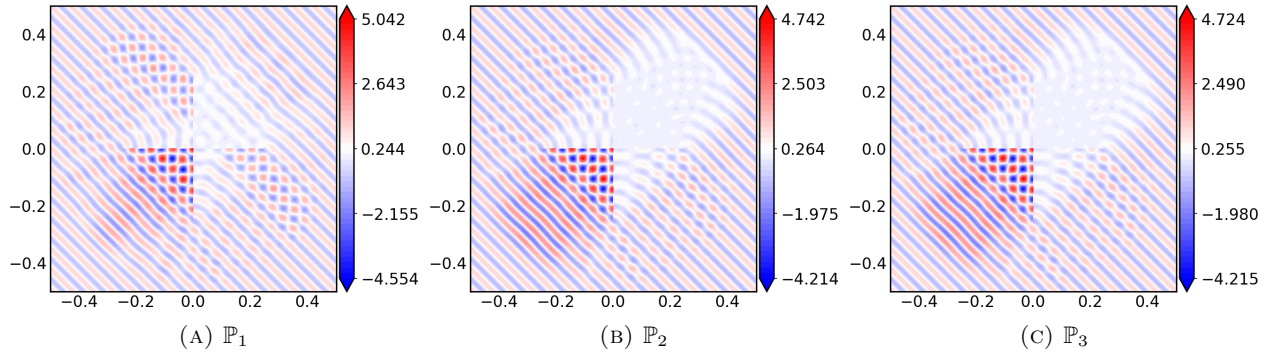
Let us first consider the treatment of the 2d situation described above with this strategy. The computational domain is discretized using Mmg [9, 38] with the values `hmin` $= 0.005$ and `hmax` $= 0.007$ for the minimum and maximum lengths of an edge in the mesh. The corresponding numerical solutions are plotted in Figs. 22 and 23, representing respectively the modulus and the real part of the total field $u_{\text{in}} + u_s^{\text{PML}}$. Since u_{in} is a plane wave coming from the bottom left, the wave is almost entirely reflected by the bottom left region adjacent to the screen. This generates a "shadow" region in the upper right part of the computational domain, where the total field is very small. As is visible in the figures, the use of $\mathbb{P}_2$ or $\mathbb{P}_3$ finite elements significantly enhances the resolution of this shadow region with respect to the cheaper $\mathbb{P}_1$ finite element method. This observation is related to the 'pollution effect', which can explain relatively large numerical errors on the $\mathbb{P}_1$ solution despite

the use of a fine mesh discretization. This effect is reduced by using higher order polynomial approximation [**?**]. The absorption of the outgoing field by the PML layer in the outer disk region $D\backslash B(0, R_{\mathrm{PML}})$ is evidenced in Figure 22, where the magnitude of the total field is constant and equal to one.



(A) $\mathbb{P}_1$ (B) $\mathbb{P}_2$ (C) $\mathbb{P}_3$

FIGURE 22. Magnitude of the total field $u_{\mathrm{in}} + u_s^{\mathrm{PML}}$ made from the numerical solution u_s^{PML} to the Helmholtz problem (7.8) for the 2d multi-screen geometry of Figure 21a, respectively using $\mathbb{P}_1$ to $\mathbb{P}_3$ elements.



(A) $\mathbb{P}_1$ (B) $\mathbb{P}_2$ (C) $\mathbb{P}_3$

FIGURE 23. Real part of the total field $u_{\mathrm{in}} + u_s^{\mathrm{PML}}$ made from the numerical solution u_s^{PML} to the Helmholtz problem (7.8) for the 2d multi-screen geometry of Figure 21a, respectively using $\mathbb{P}_1$ to $\mathbb{P}_3$ elements.

Let us eventually turn to the 3d test-case. We solve the Helmholtz problem (7.8) for the multi-screen geometry of Figure 21b using $\mathbb{P}_1$ and $\mathbb{P}_2$ finite elements. The computational mesh contains 9,856,862 tetrahedra. The finite element linear systems are solved with the direct solver `mumps` on an Intel Xeon Platinum 8360Y CPU @ 2.4 GHz (Ice Lake) with 36 physical cores, leveraging `OpenMp` parallelism. The $\mathbb{P}_1$ and $\mathbb{P}_2$ finite element problems feature 1,696,294 and 13,344,279 degrees of freedom, requiring approximately 30 GB and 600 GB of Random Access Memory (RAM) respectively. The real part and the modulus of the total field are plotted in Figure 24 to Figure 26 using two different 3d visualisations. Again, we observe substantial differences between the $\mathbb{P}_1$ and $\mathbb{P}_2$ solutions, the latter capturing better the shadow behind the obstacle in the positive coordinate regions.

Acknowledgements. The work of C.D is partially supported by the projects ANR-24-CE40-2216 STOIQUES and ANR-22-CE46-0006 StableProxies. The resources and services used in this work involving three-dimensional computations were provided by the VSC (Flemish Supercomputer Center), funded by the Research Foundation Flanders (FWO) and the Flemish Government.

APPENDIX A. DERIVATION OF THE SPHERICAL PML TENSOR IN ARBITRARY DIMENSION

This appendix provides a brief justification of the expression (7.9) of the PML tensor $A(x)$ (see also, *e.g.*, [33, 20, 55].)

Let $d \geq 2$ and let Ω be a bounded domain of $\mathbb{R}^d$; we set $\Omega_- := \Omega$ and $\Omega_+ := \mathbb{R}^d\backslash\overline{\Omega}$. The Helmholtz equation reads:

$$\begin{cases} \Delta u + k^2 u = 0 \text{ in } \Omega^+ \\ \qquad u = g \text{ on } \Omega_-, \\ \dfrac{\partial u}{\partial |x|} - \mathrm{i}ku = \mathrm{o}(|x|^{-\frac{d-1}{2}}) \text{ as } |x| \to +\infty. \end{cases} \tag{A.1}$$

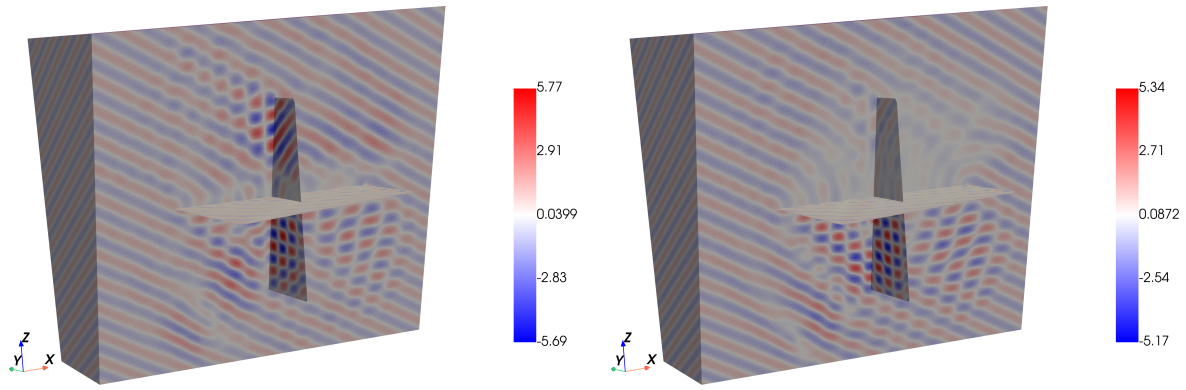


FIGURE 24. Clipped representation of the real part of the total field $u_{\text{in}} + u_s^{\text{PML}}$ made from the numerical solution u_s^{PML} to the Helmholtz problem (7.8) for the 3d multi-screen geometry of Figure 21b, using $\mathbb{P}_1$ and $\mathbb{P}_2$ elements.

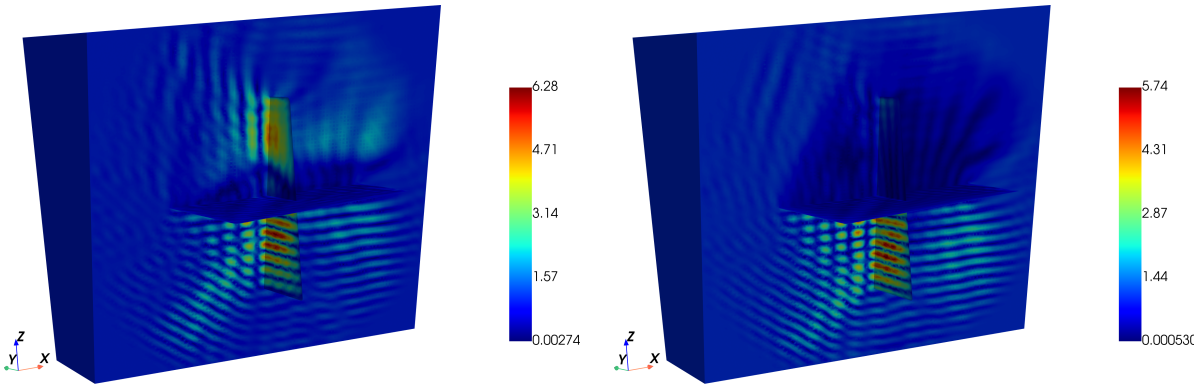


FIGURE 25. Clipped representation of the magnitude of the total field $u_{\text{in}} + u_s^{\text{PML}}$ made from the numerical solution u_s^{PML} to the Helmholtz problem (7.8) for the 3d multi-screen geometry of Figure 21b, using $\mathbb{P}_1$ and $\mathbb{P}_2$ elements.

in this appendix, we indifferently indicate by $u(x)$ and $u(r,\theta)$ the dependence of a function u on the Cartesian coordinates, and on the spherical coordinates (r,θ), where $r \in \mathbb{R}_+$ and θ belongs to the unit sphere $\mathbb{S}_{d-1} \subset \mathbb{R}^d$. Let $R > 0$ be large enough so that $\overline{\Omega} \subset B(0,R)$.

According to [29], the Perfectly Matched Layer method can be understood as a complex stretching of the radial coordinate. For a given direction $\theta \in \mathbb{S}^{d-1}$, the function $r \mapsto u(r,\theta)$ has an analytic continuation (still denoted by u) as a holomorphic function on some complex neighborhood U of the set $\{x \in \mathbb{R} : |x| > R\}$; for instance, ony may take $U = B(0,R)^c \backslash \mathbb{R}_-$ if $d = 2$ and $U = B(0,R)^c$ if $d = 3$ (this follows, for instance, from the Atkinson-Wilcox expansion of outgoing waves in dimension 3 [5, 106], [34, Theorem 3.6], and from the Karp expansion of outgoing waves in dimension 2, see [68]). Besides, for a given argument $\varphi_0 \in (0, \pi/2)$, the function $r \mapsto u(re^{i\varphi_0}, \theta)$ has exponential decay as $r \to \infty$, which follows from the properties of the outgoing fundamental solution $r \mapsto \frac{e^{ikr}}{r}$. Let us then introduce the function $r \mapsto \widetilde{u}(r,\theta)$ accounting for a smooth transition from the oscillatory behavior of $r \mapsto u(r,\theta)$ to the evanescent behaviour of $r \mapsto u(e^{i\varphi_0} r, \theta)$, by setting

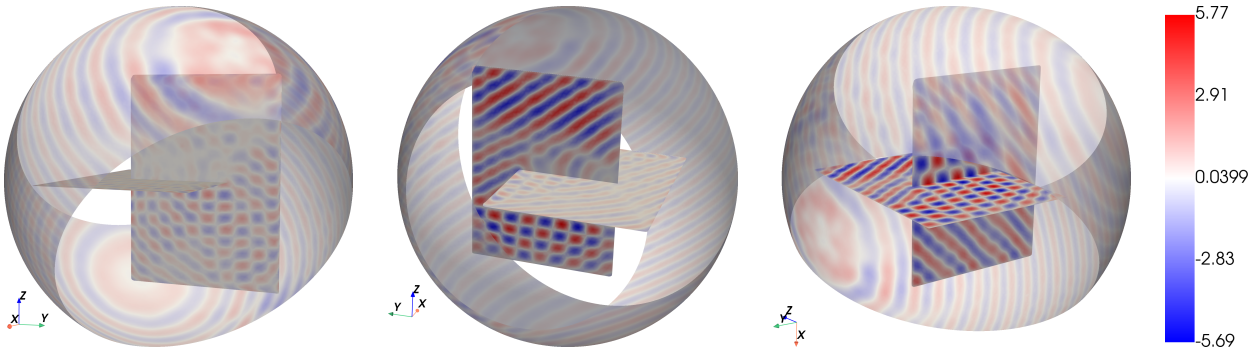
$$\widetilde{u}(r,\theta) := u(\widetilde{r}(r),\theta)\,, \quad \widetilde{r}(r) := r(1 + i\sigma(r)) =: r\alpha(r)\,,$$

in which the imaginary part $\sigma(r)$ satisfies $\sigma(r) \equiv 0$ on $\{r \le r_{\text{PML}}\}$, smoothly increases from 0 to some fixed value $\sigma_0 > 0$ in a transition layer $\{r_{\text{PML},1} < r \le r_{\text{PML},2}\}$, and then remains constant equal to σ_0 in $\{r > r_{\text{PML},2}\}$. The function $\widetilde{u}$ then coincides with u in the region $\{|x| \le r_{\text{PML},1}\}$, and it has exponential decay when $r \to \infty$ by construction. Moreover, using the relation

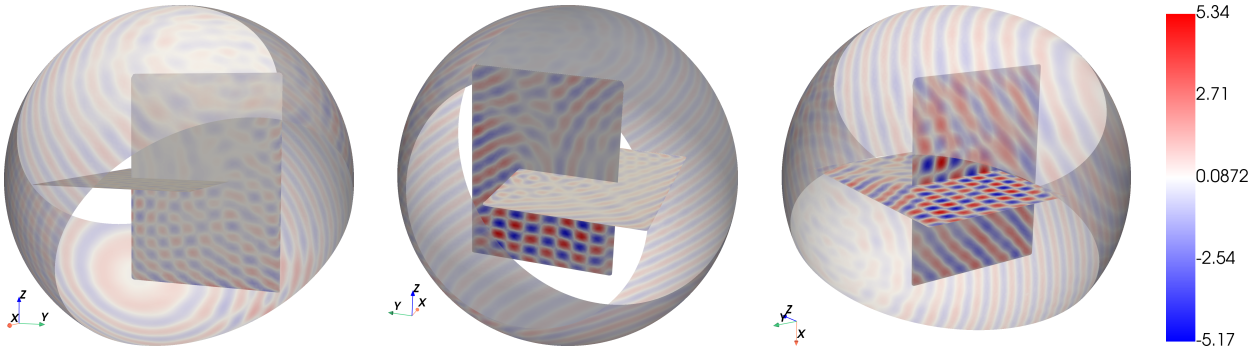
$$\frac{d\widetilde{r}}{dr} = r\beta(r)\,, \quad \beta(r) := 1 + i(\sigma + r\sigma'(r)).$$

it can be verified that $\widetilde{u}$ satisfies the following boundary value problem:

$$\begin{cases} \widetilde{\Delta}\widetilde{u} + k^2\widetilde{u} = 0 \text{ in } \Omega^+ \\ \qquad \widetilde{u} = g \text{ on } \Omega_-, \\ \qquad \widetilde{u}(x) = \mathcal{O}(1) \text{ as } |x| \to \infty \end{cases} \tag{A.2}$$

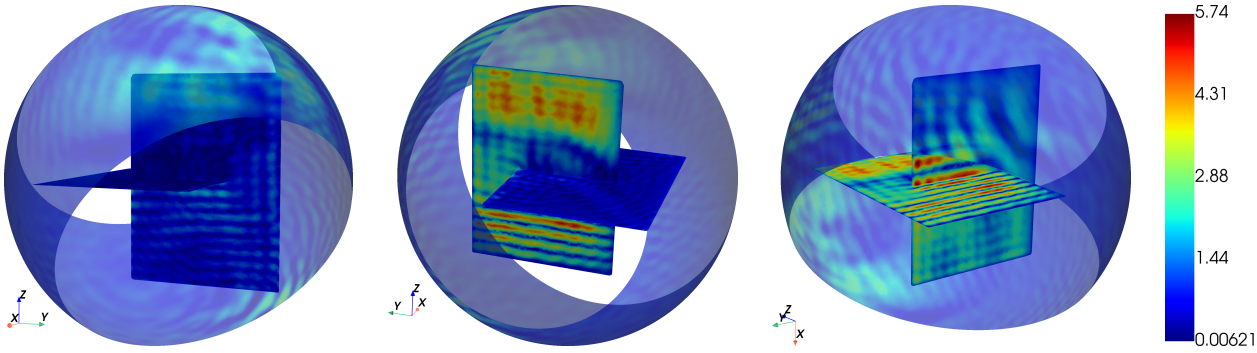


(A) $\mathbb{P}_1$

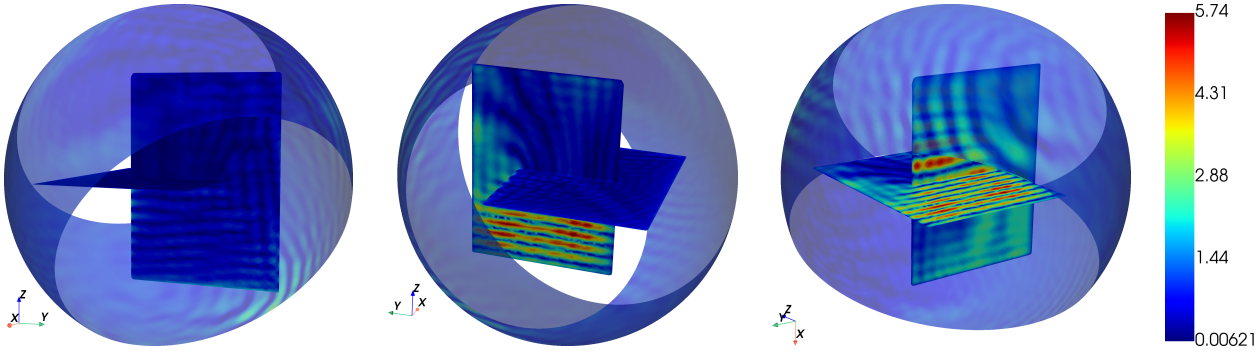


(B) $\mathbb{P}_2$

FIGURE 26. Plot of the real part of the total field $\Re(u_{\text{in}}+u_s^{\text{PML}})$ with u_s^{PML} being the numerical solution to the Helmholtz problem (7.8) for the 3d multi-screen geometry of Figure 21b, using $\mathbb{P}_1$ and $\mathbb{P}_2$ elements. The trace of the solution on a portion of the boundary of the PML region is represented to better visualise the orientation of the incoming wave.



(A) $\mathbb{P}_1$



(B) $\mathbb{P}_2$

FIGURE 27. Plot of the amplitude of the total field $u_{\text{in}} + u_s^{\text{PML}}$ induced by the numerical solution u_s^{PML} to the Helmholtz problem (7.8) for the 3d multi-screen geometry of Figure 21b, using $\mathbb{P}_1$ and $\mathbb{P}_2$ elements. The trace of the solution on a portion of the boundary of the PML region is represented for comparison with Figure 26.

where the operator $\widetilde{\Delta}$ has the following expression in spherical coordinates:

$$\widetilde{\Delta} v := \frac{1}{\alpha(r)^{d-1} r^{d-1} \beta(r)} \frac{\partial}{\partial r} \left(\frac{r^{d-1} \alpha(r)^{d-1}}{\beta(r)} \frac{\partial v}{\partial r} \right) + \frac{1}{\alpha(r)^2 r^2} \Delta_{\mathbb{S}^{d-1}} v, \tag{A.3}$$

involving the Laplace-Beltrami operator Δ_θ acting on the partial mapping $\theta \mapsto u(r,\theta)$ defined on the unit sphere. It can be shown that (A.2) admits a unique solution, which coincides with u on $\Omega^+ \cap B(0,R)$. The PML method consists in solving (A.2) numerically, instead of the original scattering problem (A.1). The point is that a simple Dirichlet truncation applied to (A.2) will result in a very small error, provided this operation is carried out on a region of space which is contained in the zone $\mathbb{R}^d \backslash \overline{B(0, r_{\mathrm{PML},2})}$ where the unique solution of (A.2) decays exponentially [20, 55].

We end up this appendix by deriving a convenient expression of $\alpha(r)^{d-1}\beta(r)\widetilde{\Delta}$ in divergence form.

Proposition A.1. *The operator $\widetilde{\Delta}$ defined in spherical coordinates by (A.3), has the following alternative expression:*

$$\alpha(|x|)^{d-1}\beta(|x|)\widetilde{\Delta} v(x) = \operatorname{div}(A(x)\nabla v(x))$$

where

$$\begin{aligned} A(x) &= \frac{\alpha(|x|)^{d-1}}{\beta(|x|)} \frac{xx^T}{|x|^2} + \alpha(|x|)^{d-3}\beta(|x|) \left(\mathrm{I} - \frac{xx^T}{|x|^2} \right) \\ &= \alpha(|x|)^{d-3}\beta(|x|)\mathrm{I} + \left(\frac{\alpha(|x|)^{d-1}}{\beta(|x|)} - \alpha(|x|)^{d-3}\beta(|x|) \right) \frac{xx^T}{|x|^2}. \end{aligned} \tag{A.4}$$

Proof. Let w be a smooth function with compact support in $\mathbb{R}^d$, whose expression in spherical coordinates is denoted by $w(r,\theta)$. Multiplying $\widetilde{\Delta} v$ by $w(r,\theta) r^{d-1} \alpha(r)^{d-1} \beta(r)$ and integrating by parts on $\mathbb{R}^d$ in spherical coordinates, we find

$$-\int_{\mathbb{R}^d} \alpha(|x|)^{d-1}\beta(|x|)\widetilde{\Delta} v w \,\mathrm{d}x = \int_{\mathbb{R}_+} \int_{\mathbb{S}_{d-1}} \left(\frac{\alpha(r)^{d-1}}{\beta(r)} \partial_r v \partial_r w + \alpha(r)^{d-3} r^{-2} \beta(r) \nabla_T v \cdot \nabla_T w \right) r^{d-1} \,\mathrm{d}\theta \,\mathrm{d}r, \tag{A.5}$$

where $\nabla_T w$ is the tangential gradient of the partial mapping $\theta \mapsto w(r,\theta)$ defined on the unit sphere $\mathbb{S}^{d-1}$. Recalling now the following identities relating the spherical and Cartesian components of the gradient,

$$\frac{\partial v}{\partial r} = \frac{x}{|x|} \cdot \nabla v, \quad \nabla_T v = |x| \left(\mathrm{I} - \frac{xx^T}{|x|^2} \right) \nabla v,$$

substituting into (A.5) and performing the change of variables from spherical to Cartesian coordinates, we obtain

$$\begin{aligned} -\int_{\mathbb{R}^d} \alpha(|x|)^{d-1}\beta(|x|)\widetilde{\Delta} v w \,\mathrm{d}x &= \int_{\mathbb{R}^d} \left[\frac{\alpha(|x|)^{d-1}}{\beta(|x|)} \frac{xx^T}{|x|^2} \nabla v \cdot \nabla w + \alpha(|x|)^{d-3}\beta(|x|) \left(\mathrm{I} - \frac{xx^T}{|x|^2} \right) \nabla v \cdot \nabla w \right] \mathrm{d}x \\ &= \int_{\mathbb{R}^d} A(x) \nabla v \cdot \nabla w \,\mathrm{d}x = -\int_{\mathbb{R}^d} \operatorname{div}(A(x)\nabla v) w \,\mathrm{d}x. \end{aligned} \tag{A.6}$$

This proves the result. □

References

[1] Abdollahipour, A., Marji, M. F., Bafghi, A. Y., and Gholamnejad, J. DEM simulation of confining pressure effects on crack opening displacement in hydraulic fracturing. *International Journal of Mining Science and Technology 26*, 4 (2016), 557–561.

[2] Alboin, C., Jaffré, J., Roberts, J. E., and Serres, C. Modeling fractures as interfaces for flow and transport in porous media. In *Contemporary Mathematics*, Z. Chen and R. E. Ewing, Eds., vol. 295. American Mathematical Society, Providence, Rhode Island, 2002, pp. 13–24.

[3] Amestoy, P. R., Duff, I. S., L'Excellent, J.-Y., and Koster, J. Mumps: a general purpose distributed memory sparse solver. In *International Workshop on Applied Parallel Computing* (2000), Springer, pp. 121–130.

[4] Andreotti, E., Edelmann, D., Guglielmi, N., and Lubich, C. Constrained graph partitioning via matrix differential equations. *SIAM Journal on Matrix Analysis and Applications 40*, 1 (2019), 1–22.

[5] Atkinson, F. Lxi. on sommerfeld's "radiation condition.". *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science 40*, 305 (1949), 645–651.

[6] Averseng, M. Jump-preserving polynomial interpolation in non-manifold polyhedra. *Mathematics of computation*. Electronically published on March 26, 2026, DOI: https://doi.org/10.1090/mcom/4181 (to appear in print).

[7] Averseng, M., Claeys, X., and Hiptmair, R. Fractured meshes. *Finite Elements in Analysis and Design 220* (2023), 103907.

[8] Bader, D. A., Meyerhenke, H., Sanders, P., and Wagner, D. *Graph partitioning and graph clustering*, vol. 588. American Mathematical Society Providence, RI, 2013.

[9] Balarac, G., Basile, F., Bénard, P., Bordeu, F., Chapelier, J.-B., Cirrottola, L., Caumon, G., Dapogny, C., Frey, P., Froehly, A., et al. Tetrahedral remeshing in the context of large-scale numerical simulation and high performance computing. *MathematicS In Action 11*, 1 (2022), 129–164.

[10] Bar, L., Chan, T. F., Chung, G., Jung, M., Kiryati, N., Sochen, N., and Vese, L. A. Mumford and shah model and its applications to image segmentation and image restoration. *Handbook of mathematical methods in imaging* (2014), 1–52.

[11] Baumann, P., and Hochbaum, D. S. An algorithm for clustering with confidence-based must-link and cannot-link constraints. *INFORMS Journal on Computing* (2022). arXiv preprint.

[12] Belhachmi, A., Caumon, G., and Dapogny, C. Tetrahedral mesh updating for subsurface modeling: implicit finite surface insertion. In *Ring Meeting 2025* (2025).

[13] Berenger, J.-P. A perfectly matched layer for the absorption of electromagnetic waves. *Journal of computational physics 114*, 2 (1994), 185–200.

[14] Berre, I., Doster, F., and Keilegavlen, E. Flow in Fractured Porous Media: A Review of Conceptual Models and Discretization Approaches. *Transport in Porous Media 130*, 1 (Oct. 2019), 215–236.

[15] Boissonnat, J.-D., and Yvinec, M. *Algorithmic geometry. Translated from the 1995 French original by Hervé Brönnimann.* Cambridge: Cambridge University Press, 1998.

[16] Bondy, J. A., and Murty, U. S. R. *Graph theory*. Springer Publishing Company, Incorporated, 2008.

[17] Borouchaki, H., and George, P. L. *Meshing, Geometric Modeling and Numerical Simulation 1: Form Functions, Triangulations and Geometric Modeling*. John Wiley & Sons, 2017.

[18] Bostro¨m, A. Review of hypersingular integral equation method for crack scattering and application to modeling of ultrasonic nondestructive evaluation. *Applied Mechanics Reviews 56*, 4 (July 2003), 383–405.

[19] Bourdin, B., Francfort, G. A., and Marigo, J.-J. *The Variational Approach to Fracture*. Springer Netherlands, Dordrecht, 2008.

[20] Bramble, J., and Pasciak, J. Analysis of a finite pml approximation for the three dimensional time-harmonic maxwell and acoustic scattering problems. *Mathematics of Computation 76*, 258 (2007), 597–614.

[21] Brenner, S. C., and Scott, L. R. *The mathematical theory of finite element methods.*, 2nd ed. ed., vol. 15 of *Texts Appl. Math.* Berlin: Springer, 2002.

[22] Buffa, A., and Christiansen, S. H. The electric field integral equation on Lipschitz screens: definitions and numerical approximation. *Numer. Math. 94*, 2 (2003), 229–267.

[23] Buffa, A., Costabel, M., and Schwab, C. Boundary element methods for Maxwell's equations on non-smooth domains. *Numer. Math. 92*, 4 (2002), 679–710.

[24] Buluç, A., Meyerhenke, H., Safro, I., Sanders, P., and Schulz, C. Recent advances in graph partitioning. *Algorithm engineering: selected results and surveys* (2016), 117–158.

[25] Burchard, P., Cheng, L.-T., Merriman, B., and Osher, S. Motion of curves in three spatial dimensions using a level set approach. *Journal of Computational Physics 170*, 2 (2001), 720–741.

[26] Chan, S. L., and Purisima, E. O. A new tetrahedral tesselation scheme for isosurface generation. *Computers & Graphics 22*, 1 (1998), 83–90.

[27] Chandler-Wilde, S. N., Hewett, D. P., Moiola, A., and Besson, J. Boundary element methods for acoustic scattering by fractal screens. *Numerische Mathematik 147*, 4 (2021), 785–837.

[28] Chen, B., Sun, Y., Barboza, B. R., Barron, A. R., and Li, C. Phase-field simulation of hydraulic fracturing with a revised fluid model and hybrid solver. *Engineering Fracture Mechanics 229* (Apr. 2020), 106928.

[29] Chew, W. C., and Weedon, W. H. A 3d perfectly matched medium from modified maxwell's equations with stretched coordinates. *Microwave and optical technology letters 7*, 13 (1994), 599–604.

[30] Ciarlet, P. G. *The finite element method for elliptic problems*. SIAM, 2002.

[31] Claeys, X., Giacomel, L., Hiptmair, R., and Urzúa-Torres, C. Quotient-space boundary element methods for scattering at complex screens. *BIT Numerical Mathematics* (2021), 1–29.

[32] Claeys, X., and Hiptmair, R. Integral equations on multi-screens. *Integral equations and operator theory 77*, 2 (2013), 167–197.

[33] Collino, F., and Monk, P. The perfectly matched layer in curvilinear coordinates. *SIAM J. Sci. Comput. 19*, 6 (1998), 2061–2090.

[34] Colton, D., and Kress, R. *Integral equation methods in scattering theory*, reprint of the 1983 original published by Wiley ed., vol. 72 of *Class. Appl. Math.* Philadelphia, PA: Society for Industrial and Applied Mathematics (SIAM), 2013.

[35] Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. *Introduction to algorithms*. MIT press, 2022.

[36] Dapogny, C. Body-fitted tracking of 2d open curves with a level set based mesh evolution method. *in preparation* (2025).

[37] Dapogny, C. Level set tracking of the evolution of an open surface. *in preparation* (2025).

[38] Dapogny, C., Dobrzynski, C., and Frey, P. Three-dimensional adaptive domain remeshing, implicit domain meshing, and applications to free and moving boundary problems. *Journal of Computational Physics* (Apr. 2014).

[39] Dapogny, C., and Frey, P. Computation of the signed distance function to a discrete contour on adapted triangulation. *Calcolo 49*, 3 (2012), 193–219.

[40] Daux, C., Moës, N., Dolbow, J., Sukumar, N., and Belytschko, T. Arbitrary branched and intersecting cracks with the extended finite element method. *International Journal for Numerical Methods in Engineering 48*, 12 (2000), 1741–1760.

[41] Davis, T. A. Algorithm 832: Umfpack v4. 3—an unsymmetric-pattern multifrontal method. *ACM Transactions on Mathematical Software (TOMS) 30*, 2 (2004), 196–199.

[42] Delfour, M. C., and Zolésio, J.-P. *Shapes and geometries: metrics, analysis, differential calculus, and optimization*. SIAM, 2011.

[43] Diao, Y., Yang, J., Zhang, Y., Zhang, D., and Du, Y. Solving multi-material problems in solid mechanics using physics-informed neural networks based on domain decomposition technology. *Computer Methods in Applied Mechanics and Engineering 413* (Aug. 2023), 116120.

[44] Doi, A., and Koide, A. An efficient method of triangulating equi-valued surfaces by using tetrahedral cells. *IEICE TRANSACTIONS on Information and Systems 74*, 1 (1991), 214–224.

[45] Dolean, V., Jolivet, P., and Nataf, F. *An Introduction to Domain Decomposition Methods: Algorithms, Theory, and Parallel Implementation*. Society for Industrial and Applied Mathematics, Philadelphia, PA, Nov. 2015.

[46] Erhel, J., De Dreuzy, J.-R., and Poirriez, B. Flow Simulation in Three-Dimensional Discrete Fracture Networks. *SIAM Journal on Scientific Computing 31*, 4 (Jan. 2009), 2688–2705.

[47] Ern, A., and Guermond, J.-L. *Theory and practice of finite elements*, vol. 159. Springer, 2004.

[48] Feppon, F. PyMedit, 2024.

[49] Feppon, Florian, Allaire, Grégoire, and Dapogny, Charles. A variational formulation for computing shape derivatives of geometric constraints along rays. *ESAIM: M2AN 54*, 1 (2020), 181–228.

[50] Flemisch, B., Berre, I., Boon, W., Fumagalli, A., Schwenck, N., Scotti, A., Stefansson, I., and Tatomir, A. Benchmarks for single-phase flow in fractured porous media. *Advances in Water Resources 111* (Jan. 2018), 239–258.
[51] Formaggia, L., Fumagalli, A., and Scotti, A. Numerical methods for flow in fractured porous media. In *Encyclopedia of Solid Earth Geophysics*. Springer, 2021, pp. 1125–1130.
[52] Forti, T. L. D., Forti, N. C. S., Santos, F. L. G., and Carnio, M. A. The continuous-discontinuous Galerkin method applied to crack propagation. *Computers and Concrete 23*, 4 (Apr. 2019), 235–243.
[53] Frey, P. J., and George, P.-L. *Mesh generation: application to finite elements*. ISTE, 2007.
[54] Fries, T.-P., and Belytschko, T. The extended/generalized finite element method: An overview of the method and its applications. *International Journal for Numerical Methods in Engineering 84*, 3 (2010), 253–304.
[55] Galkowski, J., Lafontaine, D., and Spence, E. Perfectly-matched-layer truncation is exponentially accurate at high frequency. *SIAM J. Math. Anal. 55*, 4 (2023), 3344–3394.
[56] Gravouil, A., Moës, N., and Belytschko, T. Non-planar 3d crack growth by the extended finite element and level sets—part ii: Level set update. *International journal for numerical methods in engineering 53*, 11 (2002), 2569–2586.
[57] Gu, Y., Zhang, C., Zhang, P., Golub, M. V., and Yu, B. Enriched physics-informed neural networks for 2D in-plane crack analysis: Theory and MATLAB code. *International Journal of Solids and Structures 276* (Aug. 2023), 112321.
[58] Hansbo, A., and Hansbo, P. An unfitted finite element method, based on Nitsche's method, for elliptic interface problems. *Computer Methods in Applied Mechanics and Engineering 191*, 47 (Nov. 2002), 5537–5552.
[59] Hecht, F. Bamg: bidimensional anisotropic mesh generator. *User Guide. INRIA, Rocquencourt 17* (1998).
[60] Hecht, F. New development in FreeFem++. *Journal of numerical mathematics 20*, 3-4 (2012), 251–266.
[61] Hiptmair, R., and Urzúa-Torres, C. Preconditioning the EFIE on screens. *Mathematical Models and Methods in Applied Sciences 30*, 9 (2020), 1705–1726.
[62] Hopcroft, J. E., and Ullman, J. D. Set merging algorithms. *SIAM Journal on Computing 2*, 4 (1973), 294–303.
[63] Hoteit, H., and Firoozabadi, A. An efficient numerical model for incompressible two-phase flow in fractured media. *Advances in Water Resources 31*, 6 (June 2008), 891–905.
[64] Hunter, J. D. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering 9*, 3 (2007), 90–95.
[65] Iarve, E. V., Zhou, E., Ballard, M. K., Gao, Z., Adluru, H. K., and Mollenhauer, D. Regularized X-FEM Modeling of Arbitrary 3D Interacting Crack Networks. *International Journal for Numerical Methods in Engineering 126*, 1 (2025), e7653.
[66] Ingraffea, A. R. Computational Fracture Mechanics. In *Encyclopedia of Computational Mechanics*. John Wiley & Sons, Ltd, 2007, ch. 11.
[67] Jagtap, A. D., Kharazmi, E., and Karniadakis, G. E. Conservative physics-informed neural networks on discrete domains for conservation laws: Applications to forward and inverse problems. *Computer Methods in Applied Mechanics and Engineering 365* (June 2020), 113028.
[68] Karp, S. N. A convergent 'farfield' expansion for two-dimensional radiation functions. *Commun. Pure Appl. Math. 14* (1961), 427–434.
[69] Karypis, G., and Kumar, V. Metis: A software package for partitioning unstructured graphs, partitioning meshes, and computing fill-reducing orderings of sparse matrices. *University of Minnesota, Technical Report* (1997).
[70] Komori, K. Simulation of shearing by node separation method. *Computers & Structures 79*, 2 (Jan. 2001), 197–207.
[71] Legentil, C., Pellerin, J., Raguenel, M., and Caumon, G. Towards a workflow to evaluate geological layering uncertainty on co2 injection simulation. *Applied Computing and Geosciences 18* (2023), 100118.
[72] Ling, D., Bu, L., Tu, F., Yang, Q., and Chen, Y. A finite element method with mesh-separation-based approximation technique and its application in modeling crack propagation with adaptive mesh refinement. *International Journal for Numerical Methods in Engineering 99*, 7 (2014), 487–521.
[73] Llanas, B., Lantarón, S., and Sáinz, F. J. Constructive Approximation of Discontinuous Functions by Neural Networks. *Neural Processing Letters 27*, 3 (June 2008), 209–226.
[74] López, C. M., Carol, I., and Aguado, A. Meso-structural study of concrete fracture using interface elements. II: Compression, biaxial and Brazilian test. *Materials and Structures 41*, 3 (Apr. 2008), 601–620.
[75] Lorensen, W. E., and Cline, H. E. Marching cubes: A high resolution 3d surface construction algorithm. In *Seminal graphics: pioneering efforts that shaped the field*. 1998, pp. 347–353.
[76] Martin, P. A., and Rizzo, F. J. On boundary integral equations for crack problems. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences 421*, 1861 (Jan. 1997), 341–355.
[77] Martin, V., Jaffré, J., and Roberts, J. E. Modeling Fractures and Barriers as Interfaces for Flow in Porous Media. *SIAM Journal on Scientific Computing 26*, 5 (Jan. 2005), 1667–1691.
[78] McLean, W. C. H. *Strongly elliptic systems and boundary integral equations*. Cambridge university press, 2000.
[79] Melenk, J. M., and Babuška, I. The partition of unity finite element method: Basic theory and applications. *Computer Methods in Applied Mechanics and Engineering 139*, 1 (Dec. 1996), 289–314.
[80] Miehe, C., et al. A phase field model for fracture in elastic solids: Formulation and finite element analysis. *Computer Methods in Applied Mechanics and Engineering* (2010).
[81] MMG Team. MMG platform, 2024.
[82] Moës, N., Dolbow, J., and Belytschko, T. A finite element method for crack growth without remeshing. *International Journal for Numerical Methods in Engineering 46*, 1 (1999), 131–150.
[83] Mukhtar, F. Relative Performance of Three Mesh-Reduction Methods in Predicting Mode III Crack-Tip Singularity. *Latin American Journal of Solids and Structures 14* (Aug. 2017), 1226–1250.
[84] Nagaraja, S., Elhaddad, M., Ambati, M., Kollmannsberger, S., De Lorenzis, L., and Rank, E. Phase-field modeling of brittle fracture with multi-level hp-FEM and the finite cell method. *Computational mechanics 63*, 6 (2019), 1283–1300.
[85] Newman, M. *Networks*. Oxford university press, 2018.
[86] Nie, F., Zhang, H., Wang, R., and Li, X. Semi-supervised clustering via pairwise constrained optimal graph. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence* (2020), pp. 3160–3166.
[87] Paulino, G., and Zhang, Z. Cohesive modeling of propagating cracks in homogeneous and functionally graded composites. In *Proceedings of the 5th GRACM International Congress on Computational Mechanics, Limassol, Cyprus* (2005), vol. 29.
[88] Paulino, G., Zhang, Z., and Celes, W. Dynamic failure, branching and fragmentation using cohesive zone modeling. *XVIII Convegno Nazionale IGF* (2006).

[89] Pelfrene, J., van Dam, S., Sevenois, R., Gilabert, F., and van Paepegem, W. Fracture Simulation of Structural Glass by Element Deletion in Explicit FEM. *Challenging Glass Conference Proceedings 5* (June 2016), 439–454.
[90] Phung, B. R., and Spear, A. D. A voxel-based remeshing framework for the simulation of arbitrary three-dimensional crack growth in heterogeneous materials. *Engineering Fracture Mechanics 209* (Mar. 2019), 404–422.
[91] Portela, A., Aliabadi, M. H., and Rooke, D. P. The dual boundary element method: Effective implementation for crack problems. *International Journal for Numerical Methods in Engineering 33*, 6 (1992), 1269–1287.
[92] Radovitzky, R., Seagraves, A., Tupek, M., and Noels, L. A scalable 3D fracture and fragmentation algorithm based on a hybrid, discontinuous Galerkin, cohesive element method. *Computer Methods in Applied Mechanics and Engineering 200*, 1-4 (Jan. 2011), 326–344.
[93] Richardson, C. L., Hegemann, J., Sifakis, E., Hellrung, J., and Teran, J. M. An XFEM method for modeling geometrically elaborate crack propagation in brittle materials. *International Journal for Numerical Methods in Engineering 88*, 10 (2011), 1042–1065.
[94] Rourke, C. P., and Sanderson, B. J. *Introduction to piecewise-linear topology.*, revised reprint of the 1972 original ed., vol. 69 of *Ergeb. Math. Grenzgeb.* Springer-Verlag, Berlin, 1982.
[95] Rudoy, E. Domain decomposition method for crack problems with nonpenetration condition. *ESAIM: Mathematical Modelling and Numerical Analysis 50* (Sept. 2015).
[96] Sethian, J. A. A fast marching level set method for monotonically advancing fronts. *proceedings of the National Academy of Sciences 93*, 4 (1996), 1591–1595.
[97] Smereka, P. Spiral crystal growth. *Physica D: Nonlinear Phenomena 138*, 3-4 (2000), 282–301.
[98] Spatschek, R., Brener, E., and Karma, A. Phase field modeling of crack propagation. *Philosophical Magazine 91*, 1 (2011), 75–95.
[99] Spear, A. D., Hochhalter, J. D., Cerrone, A. R., Li, S. F., Lind, J. F., Suter, R. M., and Ingraffea, A. R. A method to generate conformal finite-element meshes from 3D measurements of microstructurally small fatigue-crack propagation. *Fatigue & Fracture of Engineering Materials & Structures 39*, 6 (2016), 737–751.
[100] Stephan, E. P. Boundary integral equations for screen problems in IR3. *Integral Equations and Operator Theory 10*, 2 (Mar. 1987), 236–257.
[101] Sullivan, B., and Kaszynski, A. PyVista: 3D plotting and mesh analysis through a streamlined interface for the Visualization Toolkit (VTK). *Journal of Open Source Software 4*, 37 (May 2019), 1450.
[102] Sun, Y., Chen, B., Edwards, M. G., and Li, C. Investigation of hydraulic fracture branching in porous media with a hybrid finite element and peridynamic approach. *Theoretical and Applied Fracture Mechanics 116* (Dec. 2021), 103133.
[103] Tarjan, R. E. Efficiency of a good but not linear set union algorithm. *Journal of the ACM (JACM) 22*, 2 (1975), 215–225.
[104] Wang, L.-X., Wen, L.-F., Tian, R., and Feng, C. Improved XFEM (IXFEM): Arbitrary multiple crack initiation, propagation and interaction analysis. *Computer Methods in Applied Mechanics and Engineering 421* (Mar. 2024), 116791.
[105] West, D. B., et al. *Introduction to graph theory*, vol. 2. Prentice hall Upper Saddle River, 2001.
[106] Wilcox, C. H. A generalization of theorems of rellich and atkinson. *Proceedings of the American Mathematical Society 7*, 2 (1956), 271–276.
[107] Xiang, G., Zhou, W., Yuan, W., Ji, X., and Chang, X. Pore pressure cohesive zone modelling of complex hydraulic fracture propagation in a permeable medium. *European Journal of Environmental and Civil Engineering 25*, 10 (Aug. 2021), 1733–1749.
[108] Yang, ZJ., and Chen, J. Finite element modelling of multiple cohesive discrete crack propagation in reinforced concrete beams. *Engineering Fracture Mechanics 72*, 14 (2005), 2280–2297.
[109] Zhao, L., and Shao, Q. Denns: Discontinuity-Embedded Neural Networks for Fracture Mechanics, Feb. 2025.