



HAL
open science

An integration technique for 3D curved cracks and branched discontinuities within the eXtended Finite Element Method

Bertrand Paul, Marcel Ndeffo, Patrick Massin, Nicolas Moes

► To cite this version:

Bertrand Paul, Marcel Ndeffo, Patrick Massin, Nicolas Moes. An integration technique for 3D curved cracks and branched discontinuities within the eXtended Finite Element Method. *Finite Elements in Analysis and Design*, 2017, 123, pp.19-50. 10.1016/j.finel.2016.09.002 . hal-01455394

HAL Id: hal-01455394

<https://hal.science/hal-01455394>

Submitted on 3 Feb 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

An integration technique for 3D curved cracks and branching discontinuities within the eXtended Finite Element Method

Paul B.^(1,2,3), Ndeffo M.⁽¹⁾, Massin P.⁽¹⁾, Moës N.⁽⁴⁾

⁽¹⁾ Laboratoire IMSIA, UMR EDF-CNRS-CEA-ENSTA ParisTech, 828 Boulevard des Maréchaux 91762 Palaiseau Cedex, France

⁽²⁾ IFPEN, 1 à 4 avenue de Bois-Préau, 92852 Rueil-Malmaison Cedex, France

⁽³⁾ GeoRessources UMR 7359 Université de Lorraine, ENSG ; CNRS ; CREGU ; 54518 Vandœuvre-lès-Nancy Cedex, France

⁽⁴⁾ GeM, École Centrale de Nantes/Université de Nantes/UMR 61831, rue de la Noé, 44321 Nantes Cedex 3

bertrand.paul@ifpen.fr

marcel.ndeffo@edf.fr

patrick.massin@edf.fr

nicolas.moes@ec-nantes.fr

Abstract

In this paper, we present a robust procedure for the integration of functions discontinuous across arbitrary curved interfaces defined by means of level set functions for an application to linear and quadratic eXtended Finite Elements. It includes the possibility to have branching discontinuities between the different sub-domains. For the volume integration, integration subcells are built from the approximation mesh, in order to obtain an accurate approximation of the sub-domains. The set of subcells we get constitutes the integration mesh, which can also be used by the visualization tools. Then, we extract the faces of these integration subcells that coincide with the sub-domain boundaries, allowing us to perform surface integrations on the sub-domain boundaries. When combined with the eXtended Finite Element Method (XFEM) optimal convergence rates are obtained with curved geometries for both linear and quadratic elements.

Keywords: XFEM, level-set-method, high order approximation, numerical integration, curved geometries

Introduction

The Extended Finite Element Method (X-FEM) is an extension to the classical finite element method introduced by Moës, Dolbow and Belytschko in [Dolbow**] which allows easy handling of problems with jumps or singularities. The X-FEM accounts for the presence of discontinuities by means of a local enrichment of the polynomial interpolation space with discontinuous functions, based on the partition of unity method [Melenk**]. The discontinuities are usually described implicitly by means of the level set method [Osher**], [Sethian**]. Some of the finite elements of the mesh are then crossed by the implicit discontinuities. This implies the necessity to account for the presence of discontinuities within the elements for the numerical integration. The finite elements are split into integration subcells approximating the subdomains formed by the arbitrary interfaces, so that the classical numerical integration techniques may be used on each continuous subdomain. One may also want to enforce loadings on the immersed discontinuities (contact efforts, fluid pressure in the case of hydraulic fractures). This requires a reconstructed approximation of the arbitrary discontinuities. It is clear then that the accuracy of the extended finite element depends on the quality of approximation of the subdomains formed by the immersed discontinuities. For this matter, appropriate error measures have been introduced by Ferté [Ferté**] for linear and quadratic extended finite elements. Corresponding theoretical orders of convergence were obtained by in 2

dimensions. In this paper, we propose an accurate integration technique for 3 dimensional quadratic models. The presence of discontinuities within the elements also prefigures major conditioning issues as the shape function supports associated to enriched degrees of freedom depends on the position of the arbitrary interfaces. This is even worse for 3 dimensional quadratic models. For this aim, we rely on the work of Ndeffo et al [Ndeffo**] who proposed a robust X-FEM formulation for quadratic models. In particular, we extend it to the case of branched interfaces.

The extension of the X-FEM to higher order elements has been widely considered. First of all, Stazi and Belytschko [Stazi**], based on the work of Chessa [Chessa**] on the partition of unity, proposed a quadratic interpolation for the classical part of the displacement but a linear one for the enriched Heaviside and crack-tip part. In order to better describe the crack curvature, the level set function was discretized using the same quadratic shape functions as those of the classical part of displacements and its iso-zero was approximated by means of a linear subdivision. But the authors highlighted the necessity to consider the curvature of the interface for the design of the integration subcells. Cheng and Fries [Cheng**], Dréau et al. [Dréau**], and Mounmassi and coworkers [Mounmassi**] observed suboptimal convergence rates for higher formulation with linear representations of curved interfaces. The description of the curved crack geometries over finer subgrids allowed to recover optimal error rates, provided that the straight subcells approximating the curved crack were small enough to compensate the interpolation error perpetrated by the linear subdivision [Legrain2**]. Finally, Legay, Wang and Belytschko [Legay**] and Huerta [Huerta**] used quadratic curved subcells in 2 dimensions to approximate curved interfaces and observed optimal convergence rates. The quadratic interpolation of the crack-tip part of the enrichment was considered in [Laborde**] but encountered major conditioning issues. In order to circumvent this difficulty, alternative enrichment strategies were proposed: enrichment with cutoff functions [Chahine**], vectorial enrichment [Chevaugéon**] or both combined [Nicaise**]. The use of a preconditioner over a fixed enriched area around the crack-tip [Béchet**] also offered a successful remedy. The numerical examples we present in this paper are limited to Heaviside enrichment, but our cutting procedure provides an accurate quadratic approximation of crack fronts which may be combined with the tools mentioned above for the integration over crack tip areas.

In this paper, the approach of Legay and coworkers [Legay**] is extended to 3 dimensional models. Starting from the discretized iso-zero of the level set, the elements are split into straight tetrahedral subcells in the linear case and quadratic curved tetrahedral subcells in the quadratic case. The resulting integration subcells are then considered as belonging to one or the other side of the discontinuity. The numerical results we obtain show the accuracy of the proposed integration procedure.

This paper is organized as follow. Some preliminaries are given in Section 1: we discuss the integration problem within the extended finite element method framework. Then Section 2 focuses on the partitioning of the approximation mesh with respect to the arbitrary discontinuities. The elements of the mesh that are crossed by the discontinuities are split into integration subcells fitting the discontinuities. In Section 3, we detail the recovery of the approximated discontinuity surfaces as the set of faces of integration subcells fitting the arbitrary discontinuities. In Section 4, we depict the X-FEM formulation we use for branched discontinuities. Then, convergence analyses are performed on the resolution of the approximated subdomains for 3D models with curved branched interfaces. Optimal convergence rates are achieved for both the linear and the quadratic case. Finally, we perform X-FEM mechanical test cases are conducted in 3 dimensions with a single curved interface. Convergence analyses are performed for both the displacement norm and the energy norm. Once again, we obtain optimal convergence rates for both the linear and the quadratic case.

Part 1: Preliminaries

1.1 Model problem

The formulation is written within the framework of linear elastostatic. We consider a crack Γ in a domain Ω with boundary $\partial\Omega$ and outward normal n , representing a linear elastic and isotropic solid. The crack is composed of two walls Γ_- and Γ_+ whose outward normal are respectively n_- and n_+ . The domain is submitted to imposed displacements \bar{u} on Γ_u and imposed tractions \bar{t} on Γ_t . The fracture walls are eventually subject to imposed tractions \bar{t}_- and \bar{t}_+ on Γ_- and Γ_+ (Figure 1). We study the case without volumic forces, in small strain. Stresses, strains and displacements are respectively denoted by σ , ε and u . The problem is to find the admissible displacement field which satisfies the following equations:

$$\nabla \cdot \sigma = 0 \text{ on } \Omega$$

$$\sigma \cdot n = \bar{t} \text{ on } \Gamma_t$$

$$\sigma \cdot n_- = \bar{t}_- \text{ on } \Gamma_- \text{ and } \sigma \cdot n_+ = \bar{t}_+ \text{ on } \Gamma_+$$

$$\varepsilon(u) = \frac{1}{2}(\nabla u + (\nabla u)^t) \text{ on } \Omega$$

$$u = \bar{u} \text{ on } \Gamma_u$$

$$\sigma = C : \varepsilon \text{ on } \Omega \text{ where } C \text{ is Hooke's tensor}$$

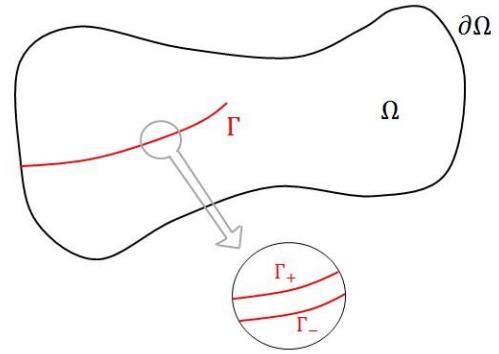


Figure 1: definition of the problem.

Let U be the space of admissible displacement field and U_0 the space of kinematically admissible test functions, vanishing on Γ_u . The weak form of equilibrium elastostatic and boundary conditions can be written as:

Find $u \in U$ such that:

$$\int_{\Omega} \varepsilon(u) : C : \varepsilon(v) d\Omega = \int_{\Gamma_t} \bar{t} \cdot v d\Omega + \int_{\Gamma_-} \bar{t}_- \cdot v d\Omega + \int_{\Gamma_+} \bar{t}_+ \cdot v d\Omega, \forall v \in U_0 \quad (1)$$

Within the finite element method, the domain Ω is meshed. The approximation of the displacement field u_h on the elements of the mesh is:

$$u_h(x) = \sum_{i \in N} \varphi_i(x) u_i$$

where φ_i are the finite element shape functions, u_i the nodal displacements and N the set of nodes of the mesh.

But the internal crack Γ does not necessarily conform the mesh. So in the framework of the extended finite element method (through the partition of unity [Melenk**]), the approximation of the displacement field is enriched by additional discontinuous functions to allow a displacement jump across the crack:

$$u_h(x) = \sum_{i \in N} \varphi_i(x) u_i + \sum_{j \in N_g} \varphi_j(x) F(x) a_j \quad (2)$$

where F is the enrichment function; a_j are the additional degrees of freedom for enriched nodes and N_g the set of enriched nodes. The function F is discontinuous across the fracture Γ , usually represented by means of level set functions.

1.2 Level set method

The arbitrary discontinuities are localized thanks to level set functions. An arbitrary interface is introduced thanks to a single level set function which represents the normal distance to the interface surface. The interface then corresponds to the iso-zero of this level set. In this article, this level set function is named the normal level set or lsn . When it comes to describing arbitrary cracks, two level sets are necessary. The first one still marks the normal distance to the crack surface, regardless of the presence of the crack front. The second one, the tangential level set (lst), is the tangent distance to the crack front with respect to the crack surface. The crack is defined as the set of points \mathbf{x} that satisfy: $\begin{cases} lsn(\mathbf{x}) = 0 \\ lst(\mathbf{x}) < 0 \end{cases}$. The level sets are often chosen as distance functions satisfying: $\begin{cases} \|\nabla lsn\| = 1 \\ \|\nabla lst\| = 1 \end{cases}$ and given their definition, they must also satisfy: $\nabla lsn \cdot \nabla lst = 0$

On Figure 2 (bottom), we observe the arbitrary crack described by a normal level set (top left) and a tangential level set (top right).

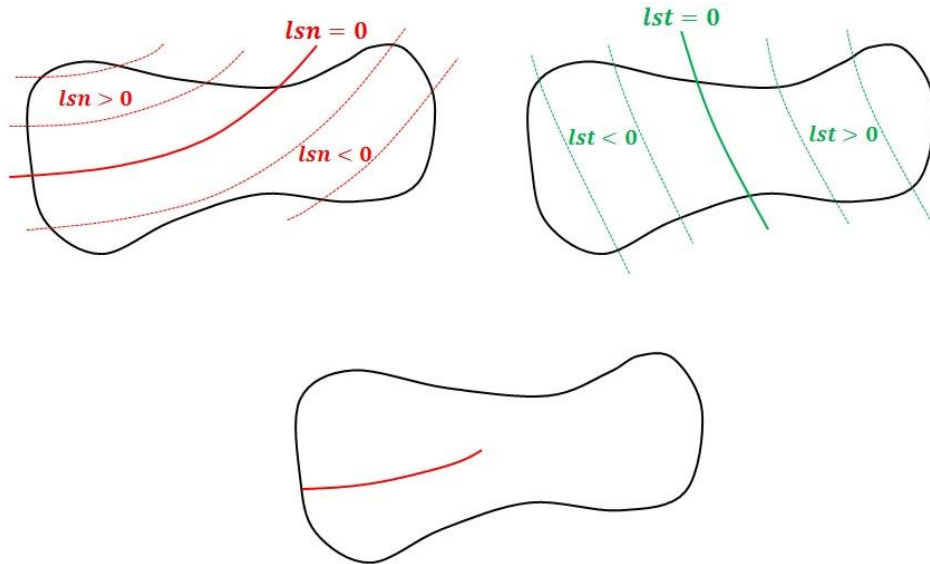


Figure 2: on the use of level set functions to describe arbitrary discontinuities.

In the implementation we have chosen and which is the most common, the level set functions are approximated by the same shape functions as the displacement field (see [Moës**]). This enables the crack shape to be described entirely in terms of nodal values. Of course this is not necessary nor always most convenient and one could chose to work with the functions themselves. However, in case of automatic propagation, the functions are not known explicitly and their numerical estimates at the nodes provide the only information to characterize them. So from now on, we abusively use the expressions “normal level set” and “tangential level set” to designate the discretized level sets.

The crack Γ depicted by the original level set functions is then discretized within the finite element mesh by means of the discretized level set functions. The resulting discretized discontinuity is denoted Γ_h . Finally, the elements crossed by Γ_h are split into integration subcells approximating the discontinuity (quadratic integration subcells in the quadratic case and linear ones in the linear case). The resulting approximation of the crack is denoted $\tilde{\Gamma}$ and does not necessarily coincide with the

discretized crack Γ_h [Ferté1**] (see Figure 4). In order to measure the error introduced in this two steps process, we compute the resolution ε defined as follow:

$$\varepsilon = \max_{x \in \Gamma} (\min_{\tilde{x} \in \tilde{\Gamma}} |x - \tilde{x}|)$$

For each point x of the theoretical interface Γ , we compute the distance to the approximated interface $\min_{\tilde{x} \in \tilde{\Gamma}} |x - \tilde{x}|$. Finally, we take the maximum over $x \in \Gamma$. The resolution ε is then the maximal distance between the analytical interface and the approximated interface. Ferté ([Ferté1**]) showed that the resolution is proportional to h^2 in the linear case and h^3 in the quadratic case, h denoting the representative size of the elements of the mesh, provided that the theoretical crack Γ has sufficient geometric continuity.

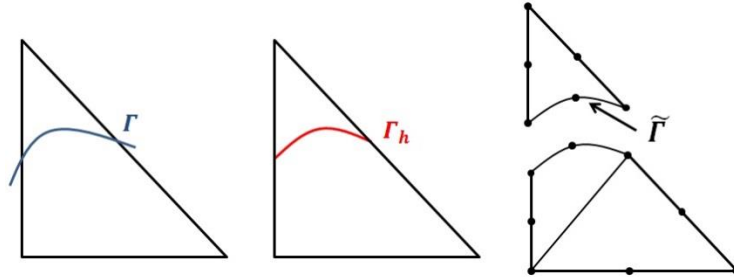


Figure 4: the analytical arbitrary interface Γ (left), the discretized iso-zero of the normal level set Γ_h (middle) and the approximated interface $\tilde{\Gamma}$ (right).

1.3 Domain integration

The approximation of the displacement field is discontinuous across the crack (equation (2)) so that the quantities we have to integrate over the domain Ω (equation(1)) are also discontinuous. In order to integrate discontinuous quantities on each side of an interface, a first step consists in the design of a physical support for each sub-domain so that we can use the classical integration techniques on each continuous domain. We aim at getting a quadratic accurate approximation of the sub-domains cut off by the arbitrary discontinuities.

In order to obtain an approximation of the sub-domains, we identify the elements of the mesh that are crossed by an arbitrary discontinuity and divide them into integration subcells in compliance with the discontinuity. Within an element that includes a crack front, we perform this cutting procedure regardless of the presence of the crack front (Figure 3). The presence of the crack front is then eventually taken into account with specific degrees of freedom associated to singular enrichment functions [Dolbow**] [Béchet**] or with internal variables within the framework of cohesive zone model [Ferté2**]. In the approach proposed by Minnebo et al. [Minnebo**], the elements of the mesh that include the crack front are cut with respect to the crack surface only: $\{l_{sn} = 0\} \cap \{l_{st} < 0\}$, so that the crack front position is topologically induced by the set of integration subcells. But the extension of this method to 3D models is laborious. Two types of elements are then concerned by the cutting procedure for a crack:

- The elements that are entirely crossed by a discontinuity. The edges of these elements are strictly intersected by the iso-zero of the normal level set and on each of these intersection we satisfy $l_{st} < 0$.
- The elements that are cut by the discontinuity and include a piece of the crack front. The edges of these elements are strictly intersected by the iso-zero of the normal level set and on the set of intersection points we satisfy $\max\{l_{st}\} * \min\{l_{st}\} < 0$.

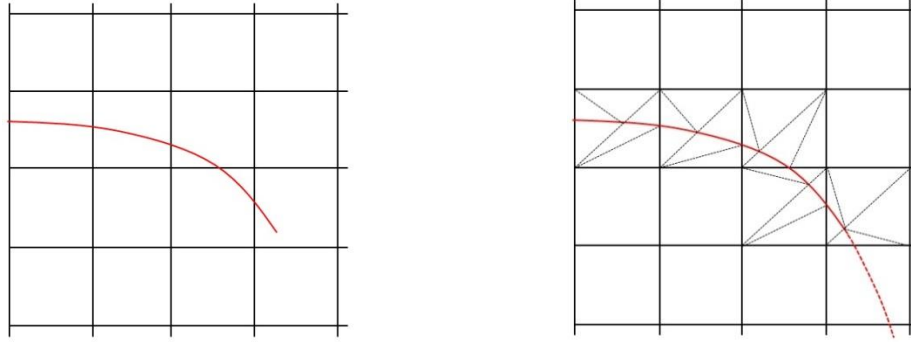


Figure 3: a crack on a regular mesh (left) and the triangular integration subcells generated in the elements crossed by the crack (right).

Once the cutting procedure is done, we can make the distinction between two meshes. The initial mesh whose nodes carry the degrees of freedom of the problem and the mesh resulting from the elements of the initial mesh that are not cut and the integration subcells designed to fit the arbitrary discontinuities. We denote this second mesh integration mesh, it is used for the domain integration but also for post-processing with visualization tools.

The difficulty of the problem lies in the construction of the integration subcells. We must have a systematic and robust procedure that manages to shape quadratic sub-elements fitting arbitrary discontinuities for 3D models with curved cracks. To our knowledge, there are few methods described in the literature [Fries**] for the cutting of 3D elements in the quadratic case. In the following, we propose a robust method to consistently create quadratic subcells fitting a crack surface in the 3D case.

Part 2: Partitioning 3D domains with arbitrary discontinuities

2.1 Overview of the cutting procedure

Our cutting procedure performs without refinement. We may then fail to identify small inclusions embedded within a single element, but we assume the user is aware of the relative size of the element of the mesh compared to the refinement of the level sets. On the contrary, the approach proposed by Fries et al [Fries**] uses a sample grid to detect eventual changes in the sign of the level set within the elements of the mesh. Adaptive remeshing with quadtree and octree meshes have also been studied in the extended finite method [Legrain**]. These methods enable a finer approximation of the immersed discontinuities and reduce the number of topologically distinct cutting configurations. Similarly to the approach of [Fries**], we build integration subcells fitting the arbitrary discontinuities with a quadratic accuracy. Then, we recover a reconstructed approximation of the immersed interfaces as the set of faces of the integration subcells coinciding with the discontinuities. Two noticeable features of our cutting procedure are the systematic preliminary reduction into primary elements and the level set adjustment procedure. This considerably reduces the number of topologically distinct cutting configurations we may encounter. The determination of the intersection points between the mesh and the immersed boundaries is systematically performed in the reference configuration of the elements so that the overall cutting procedure only relies on a one dimensional root-finding algorithm. We expect a quadratic convergence for this Newton-Raphson algorithm. Since the procedure depicted in this paper has been implemented in industrially

oriented finite element software, all the annoying cases have been identified and thoroughly treated. An overview of the overall cutting procedure, including the design of the integration subcells and the recovery of the contact faces, is summarized in Annex 1. Finally, the integration procedure detailed in sections 2 and 3 is entirely applicable for linear models. The difference lies in the fact that middle nodes are not necessary in the linear case for the voluminous integration subcells as well as for the contact faces.

2.2 Level set adjustments

In the linear case, the intersections between the iso-zero of the discretized normal level set with the mesh are found two ways:

- If at a node of the mesh $lsn(N) = 0$, then the node is located on the iso-zero,
- If at the edge linking node A and B $lsn(N_A) * lsn(N_B) < 0$, then the position of the intersection point I is given by: $I = N_A + \frac{lsn(N_A)}{lsn(N_A) - lsn(N_B)} \overrightarrow{N_A N_B}$ (see Figure 5), if the level set is a distance function.

The quadratic case requires more attention. The intersections between the iso-zero of the discretized normal level set and the mesh are localized:

- At the nodes of the mesh satisfying $lsn(N) = 0$
- On any edge strictly cut by the the iso-zero of the discretized normal level set.

But the condition for an edge to be cut is not similar to the linear case. Indeed, along the edge of a quadratic mesh, the discretized level sets are marked at three locations: the two end nodes and the middle node. This leads to potential double cancelation of the discretized level sets along an edge (see Figure 5).

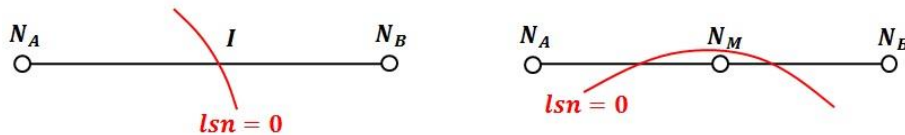


Figure 5: an intersection between the iso-zero of the normal level set and an edge in the linear case (left) and a case of double cancelation of the normal level set along an edge in the quadratic case (right).

In the end, the condition for an edge to be strictly cut by the iso-zero of the normal level set in the quadratic case is: $\max\{lsn(N_A), lsn(N_B), lsn(N_M)\} * \min\{lsn(N_A), lsn(N_B), lsn(N_M)\} < 0$. The choice is made to restrict the situations of double cancelation of the normal level set along an edge in order to reduce the number and the complexity of the cutting configurations. For this aim, we proceed to level set adjustments.

The only cases of multiple cancelations of the discretized normal level set along an edge we accept are depicted in Figure 6:



Figure 6: the two cases of multiple cancelations of the level sets along a three node edge which are authorized.

Either the normal level set is null for each node of the edge, either it is null for one vertex node and the middle node. We also forbid the case where the iso-zero of the normal level set brushes against

an edge: $\begin{cases} lsn(N_A) * lsn(N_B) > 0 \\ lsn(N_M) = 0 \end{cases}$. Four situations encountered with the discretized normal level set must then be adjusted. They are depicted on Figure 7.

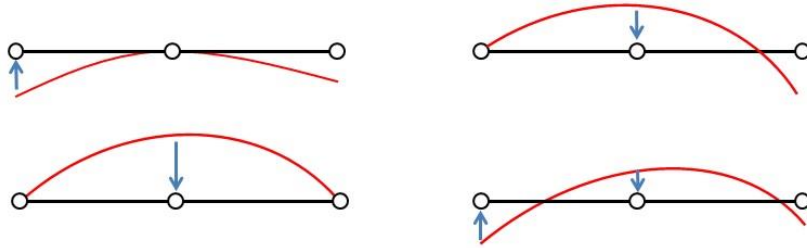


Figure 7: level set adjustments along an edge.

For each case, we modify the value of the discretized normal level set at some nodes to be reduced to one of the two configurations of Figure 6.

Each time an adjustment is performed we introduce an error because we modify the value of the discretized normal level set at one or two nodes. The approximated iso-zero of the normal level set Γ_h is then shifted (see Figure 4). It is not an optimal solution. As soon as we make adjustments, the convergence properties for the approximation of the sub-domains may not apply. But this choice is justified by the following arguments:

- the restriction of the double cancellation situations significantly reduces the number and the complexity of cutting configurations.
- the adjustments are likely to happen only when the iso-zero of the discretized level set is highly curved and close to an edge. The use of thinner meshes always ends up solving the problem.
- the code is able to return a message each time an adjustment is performed. And in order to provide an indication on the error introduced, we measure the shift realized relatively to the range of the level set values over the support of the shifted node.

Remarks:

- when an adjustment is performed on an edge, it can induce a situation that requires an adjustment on an adjacent edge. So the procedure is performed recursively until no more adjustment is needed.
- the exact same procedure is applied to the discretized tangential level set.

2.3 Reducing the problem

Before we begin the treatment of an element that needs to be cut, we perform a prior treatment in order to reduce the problem. In the reference configuration of the parent element, we systematically split the non-simplex elements into a set of simplex cells: tetrahedral elements for 3D models and triangles for 2D models. In this way, we only have one type of cell to consider for the cutting procedure. It considerably reduces the number of cutting configurations we may come across. In Figure 8 and 9, we depict the partitioning of the non-simplex elements into simplex cells.

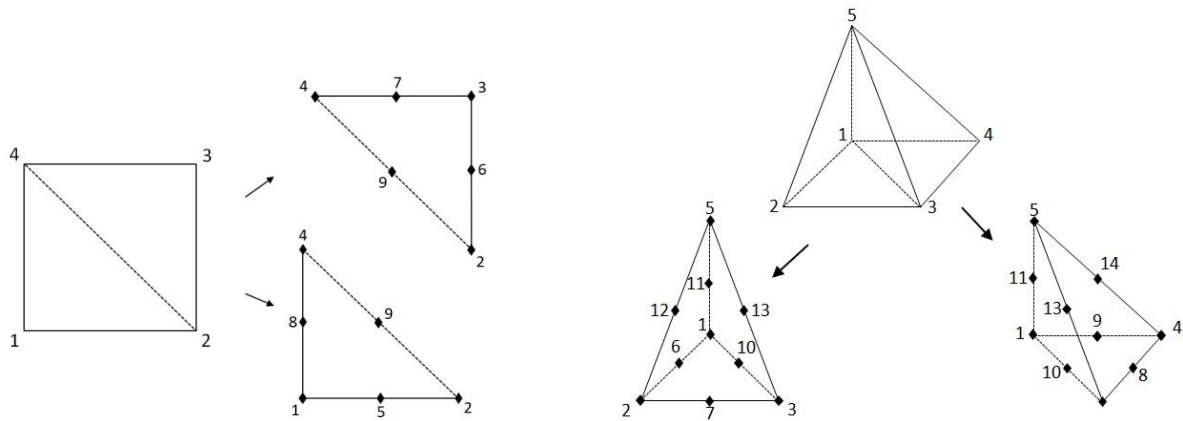


Figure 8: Partition of a quadrangle into two triangles (left) and partition of a pyramid into two tetrahedral elements (right).

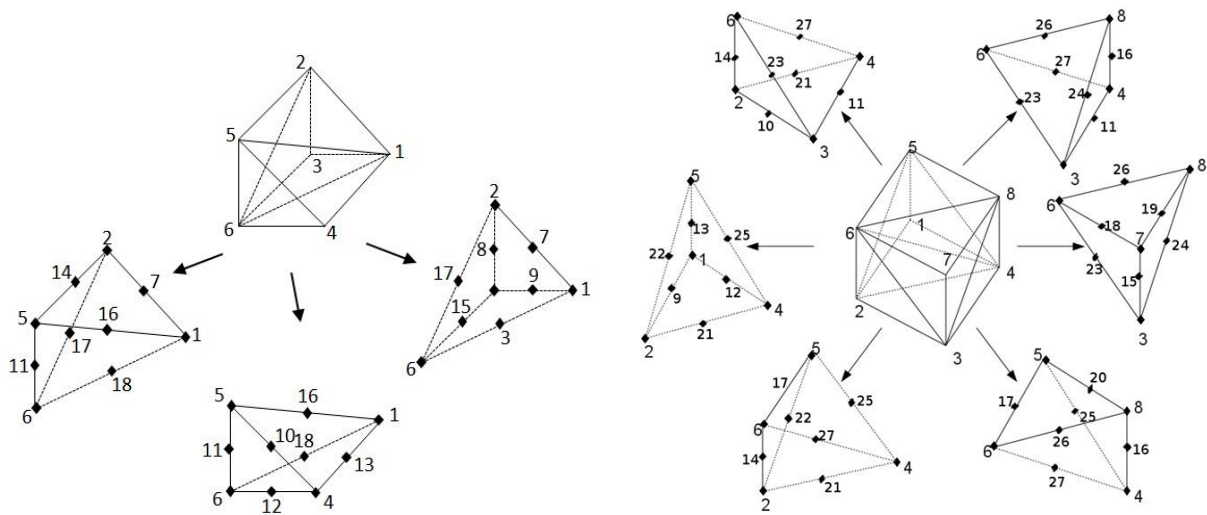


Figure 9: Partition of a pentahedron into three tetrahedral elements (left) and partition of a hexahedron into six tetrahedral elements (right).

We denote the set of simplex cells obtained as the primary simplex cells. For the simplex elements, the set of primary simplex cell is composed by the element itself.

In the quadratic case, this partitioning induces the apparition of internal edges and additional nodes. These additional nodes were not part of the initial mesh; they are fictitious nodes that do not carry degrees of freedom. As the internal edges were not part of the original mesh, they were not concerned by the level set adjustment procedure. We can thus observe double cancelations of the level sets along these internal edges. This would bring undesirable cutting configurations. A solution to bypass this problem is explained in Section 2.5. It relies on the fact that we can choose between different partitions for the non-simplex elements.

Indeed there is not a unique manner to split a quadrangular element into two triangles, there are two (see Figure 10). The same goes for the pyramidal elements. There are two eligible sets of tetrahedral elements depending on how the internal edge splits the quadrangular face. For the pentahedron, we count 6 different manners to obtain a partition of 6 tetrahedral elements (see Figure 10). Finally, there are 6 different ways to partition a hexahedron into two pentahedra (see Figure 10). So we expect a maximum of 6^3 different manners to partition a hexahedron into 6

tetrahedral elements. But amongst these 6^3 different possibilities, a lot are identical due to symmetries and we only keep those who ensure the conformity between the different tetrahedral elements. One can show we are left with 72 distinct manners to partition a hexahedron into 6 conforming tetrahedral elements. These different manners to partition the non-simplex elements into primary simplex cells constitute splitting configurations. We keep in store the possibility to select these different configurations. As we will observe in Section 2.5, the use of these different configurations is essential to prevent the apparition of undesirable double cancelation of the level set along the edges of the primary simplex cells.

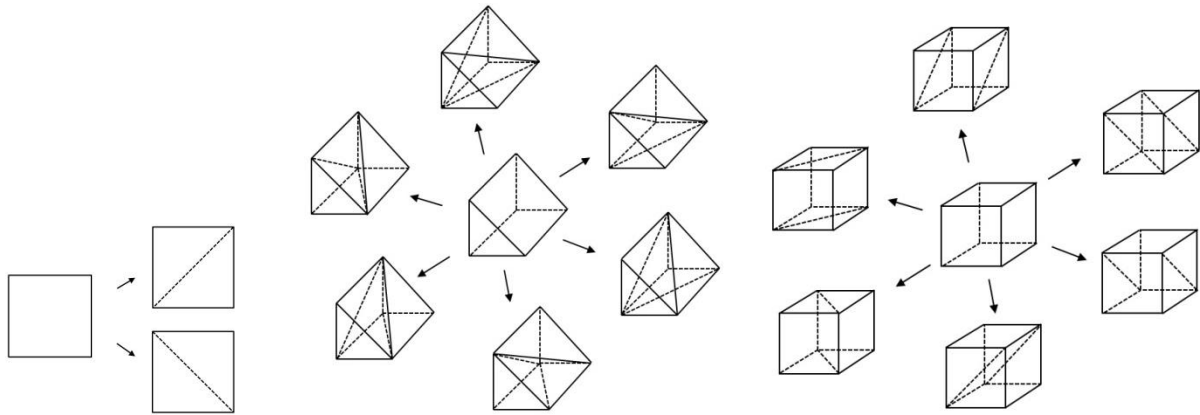


Figure 10 : the two different configurations for a quadrangular element (left), the 6 different configurations for a pentahedron (middle) an the six different ways to partition an hexahedron into two pentahedra (right).

All the elements of the mesh that need to be cut are now split into a partition of simplex cells. The problem is now reduced to the cutting of simplex cells with respect to the iso-zero of the normal level set.

2.4 Intersections between the primary simplex cells and the iso-zero of the tangential level set

For all the elements that need to be cut, we loop over the primary simplex cells and determine the intersections with the iso-zero of the discretized normal level set. For each primary simplex cell:

- we loop over the vertex nodes and identify those who coincide with the iso-zero of the normal level set ($lsn(N) = 0$). We denote as n_s the number of intersection points coinciding with vertex nodes of the primary simplex cell.
- we loop over the edges and for those verifying $lsn(N_A) * lsn(N_B) < 0$, we determine the position of the intersection point. If $lsn(N_M) = 0$, the intersection point directly coincides with the middle node N_M . In the opposite case, we use the root finding **algorithm 1** to determine the position of the intersection point. We denote by n_e the number of edges intersected by the iso-zero of the normal level set which includes the edges intersected at one vertex node and at the middle node (see Figure 6)

Remark: the unwanted cases of double cancelation of the iso-zero of the normal level set along an edge are ignored at this stage. They will be detected and cured during the next step (Section 2.5)

In order to determine the position of an intersection point on the edge of a primary simplex cell, we move to the reference configuration of the parent element and use a Newton-Raphson algorithm. The relocation in the reference configuration of the parent element presents two major advantages:

- the edges of the primary simplex cells are necessarily straight in the reference configuration of the parent element (see Figure 11).

- the convergence criterion for the Newton-Raphson algorithm is the same for all edges.

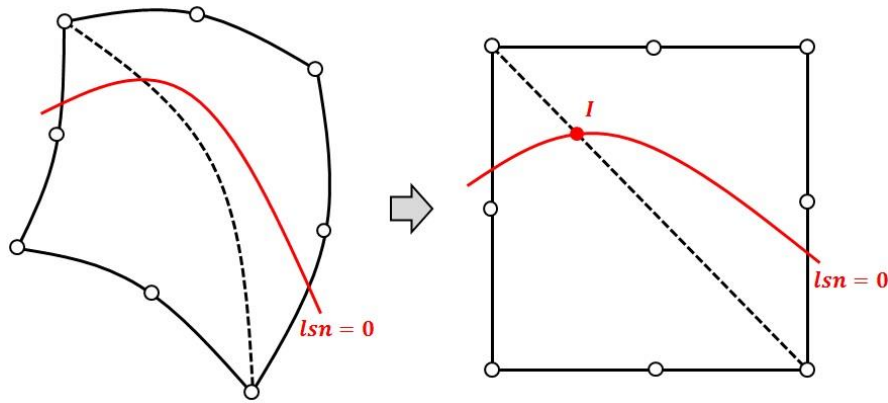


Figure 11: a quadrangular element crossed by the iso-zero of the normal level set in the real space (left) and in the reference configuration (right).

A parameterization of a straight edge $N_A N_B$ in the reference space is: $= N_A + t * \overrightarrow{N_A N_B}$, $t \in [0, 1]$

Algorithm 1 determines the intersection point between the iso-zero of a discretized level set and a straight line within an element. It only requires an initial guess of the position of the zero-level set and a unit vector carrying the search direction.

$\vec{u} = \frac{\overrightarrow{N_A N_B}}{\|\overrightarrow{N_A N_B}\|}$ provides the unit vector and a relevant initial guess is the linear approximation:
 $I_0 = N_A + \frac{lsn(N_A)}{lsn(N_A) - lsn(N_B)} \overrightarrow{N_A N_B}$

Algorithm 1: Research of an intersection point between the iso-zero of a level set and a straight line within an element

- We set n to 0, α_0 to 0 and $\Delta\alpha_0$ to 2ε
- While $|\Delta\alpha_n| > \varepsilon$
 - $n = n + 1$
 - $I_n = I_{n-1} + \Delta\alpha_{n-1} \vec{u}$
 - Compute the shape functions φ_i associated to the nodes of the parent element for I_n
 - Interpolate the value of the level set $ls(I_n) = \sum_{nodes} \varphi_i ls_i$
 - Compute the derivative of the level set field along the unit vector $\vec{u} : \vec{\nabla}ls(I_n) \cdot \vec{u}$
 - $\Delta\alpha_n = -\frac{ls(I_n)}{\vec{\nabla}ls(I_n) \cdot \vec{u}}$
- Return I_n

Tolerance ε is about a displacement increment in the reference configuration of the parent element. It is common to all elements and edges. Since the level set field are polynomial within an element, we expect a quadratic convergence for the Newton-Raphson algorithm.

Remark: when we look for an intersection point on an edge that coincide with an edge of the parent element, the discretized level set field only depends on the values of the level set at the three nodes of the edge. We could directly get the position of the intersection point from the resolution of a second order polynomial equation.

2.5 Shaping the integration subcells

Once we have determined the intersections between the primary simplex cells and the iso-zero of the normal level set, we associate to each primary simplex cell a cutting configuration. There is a total of 3 cutting configurations in the 2D case and 9 cutting configurations in the 3D case. The different cutting configurations are distinguished with the number n_s of intersection points coinciding with vertices nodes and the number n_e of edges intersected by the iso-zero of the normal level set.

In the following, we detail the different cutting configurations encountered in 2D and 3D. The table hereunder gives the number of subcells generated by the primary simplex cells for each cutting configuration. On Figure 12 to 23, we depict the different cutting configurations, the iso-zero of the tangential level set appearing in red. The nodes and the edges of the primary simplex cells that coincide with this iso-zero also appear in red.

	n_e	n_s	Number of subcells generated	Figure
2D	1	1	2	12
	2	0	3	13
	2	1	3	14
3D	1	2	2	15
	2	1	3	16
	2	2	3	17
	3	0	4	18
	3	1	4	19
	3	2	4	20
	4	0	6	21
	4	1	6	22
	3	1	5	23

Table 1: cutting configurations for the primary simplex cells.

Remark: there are two configurations labeled $\begin{cases} n_e = 3 \\ n_s = 1 \end{cases}$. The first is distinguishable from the second one because it has a common vertex for all three intersected edges.

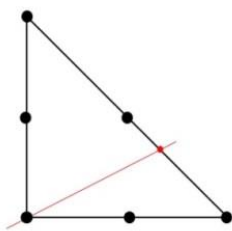


Figure 12

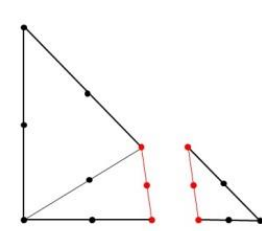
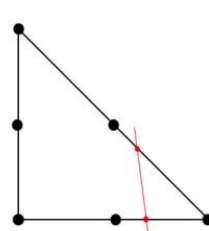
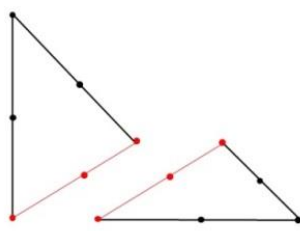


Figure 13

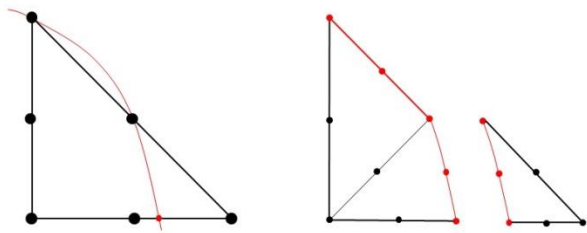


Figure 14

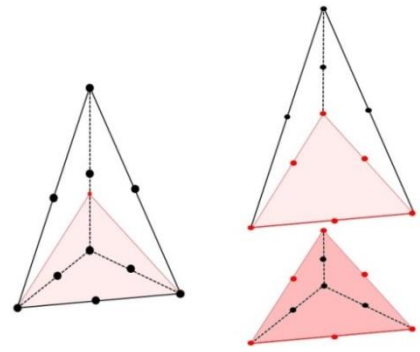


Figure 15

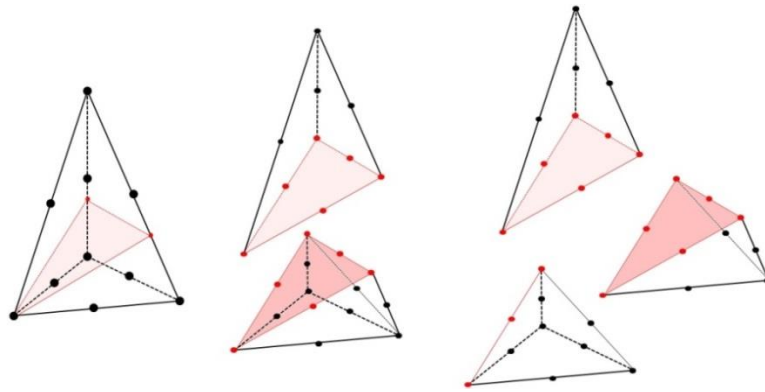


Figure 16

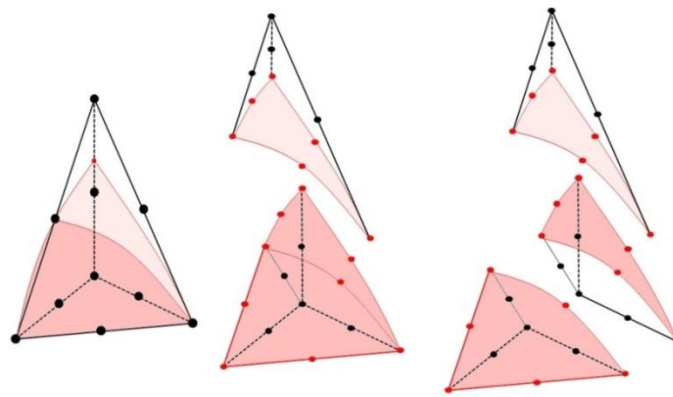


Figure 17

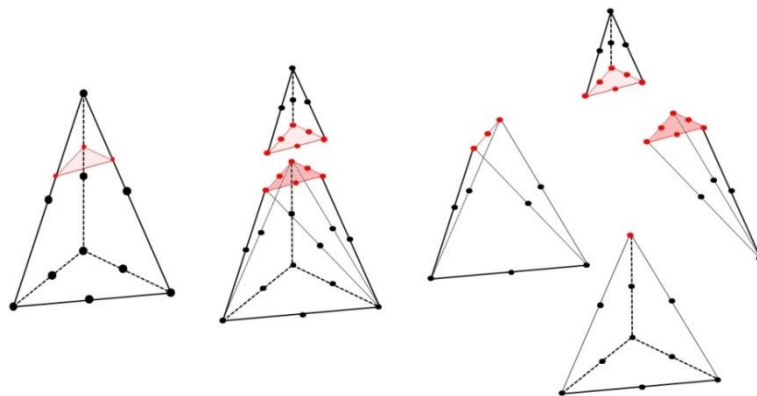


Figure 18

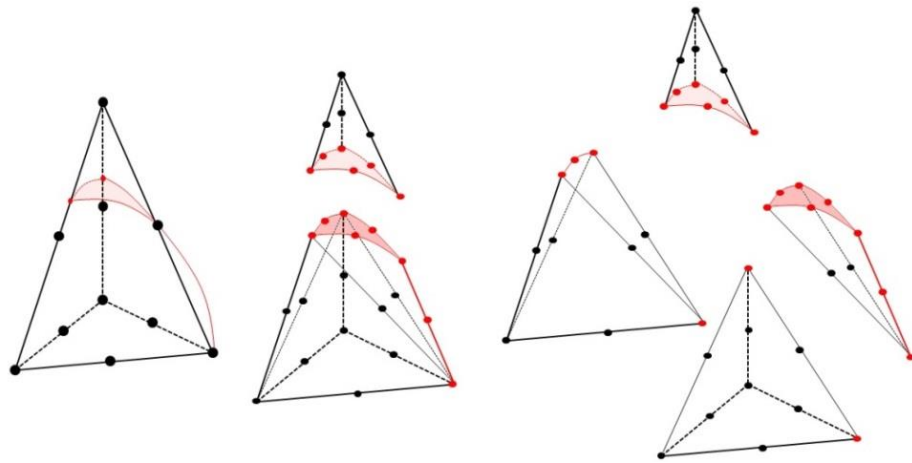


Figure 19

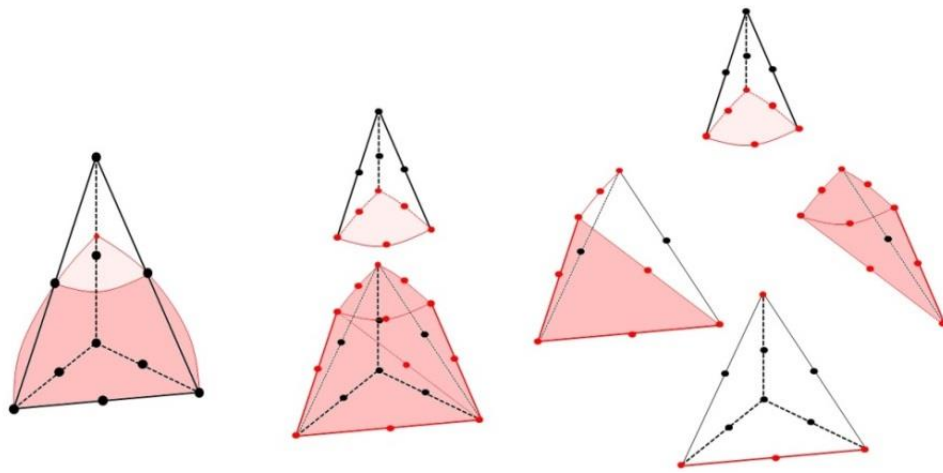


Figure 20

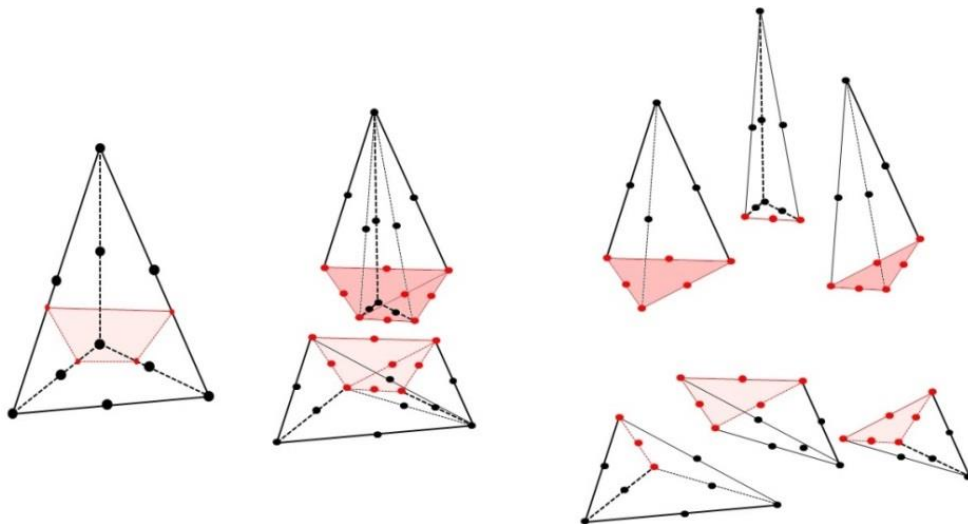


Figure 21

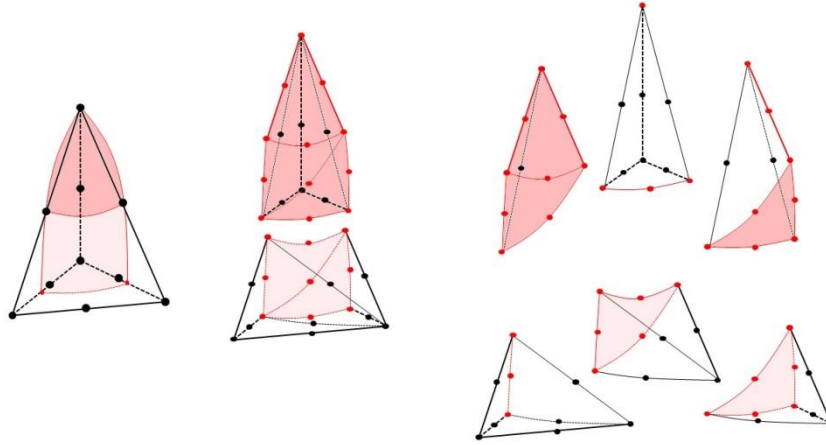


Figure 22

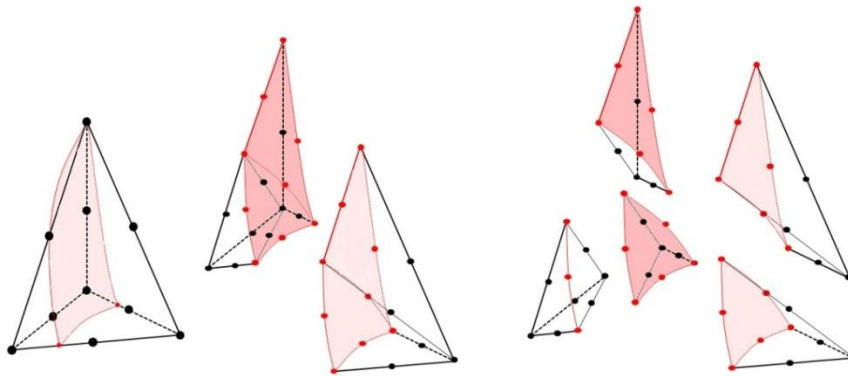


Figure 23

In order to maintain a quadratic accuracy in the approximation of the sub-domains on both sides of the discontinuity, the middle nodes of the integration subcells must be thoroughly determined.

At this stage, we have only resolved the intersections between the primary simplex cell and the iso-zero of the normal level set so that we have at our disposal the entire set of vertex nodes for the integration subcells. In the following, we detail the determination of the middle nodes for the 3D configuration $\begin{cases} n_e = 4 \\ n_s = 0 \end{cases}$. The procedure is similar for the other configurations. We distinguish 4 types of middle nodes. Here again, the determination of these nodes is performed in the reference configuration of the parent element.

1st type of middle nodes

The edges of the primary simplex cell that are intersected by the iso-zero of the normal level set are split into two edges, one on either side of the discontinuity. We place the middle nodes on each of these edges. For an edge $N_A N_B$ of a primary simplex cells intersected at point I , the positions of the middle nodes are obviously $N_A + \frac{1}{2} * \overrightarrow{N_A I}$ and $N_B + \frac{1}{2} * \overrightarrow{N_B I}$. On Figure 24, we observe the first type middle nodes placed on the intersected edges of the primary tetrahedron.

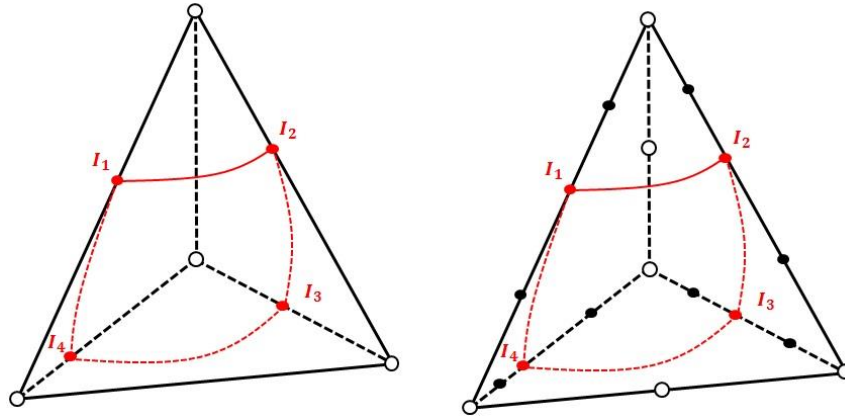


Figure 24: the primary tetrahedron with the 4 intersection points $I_{1,2,3,4}$ (left) and the first type middle nodes (black circles) on its intersected edges (right).

2nd type of middle nodes

The second type middle nodes are located on the faces of the primary tetrahedron, between the intersection points, on the approximated discontinuity. For each intersected face, we search this middle node on the perpendicular bisector to the segment formed by the two intersection points. For example in Figure 26 (left), we search the middle node between I_3 and I_4 on the perpendicular bisector to the segment $[I_3I_4]$ on the bottom face of the primary tetrahedron. We still use **algorithm 1** to locate these middle nodes. But we must ensure that the middle node stays confined in the face of the current primary simplex cell. Indeed, for non-simplex elements, the middle node can be found in the adjacent primary simplex cell, giving away a situation of double cancelation of the tangential level set along an edge (see Figure 25). In order to detect this situation, we compute first the limits t_{inf} and t_{sup} of the authorized interval along the perpendicular bisector. The middle of the segment $[I_3I_4]$ is then chosen as an initial guess and the vector \vec{u} is chosen as the unit vector on the perpendicular bisector oriented along the gradient of the normal level set. Once the algorithm has converged, if the middle node is found out of the interval $[t_{inf}, t_{sup}]$, we go back to the splitting into primary simplex cells (Section 2.3) for the parent element and select another configuration. For instance in Figure 25, we depict a 2D example of double cancelation of the normal level set along the internal edge of a quadrangle. The other splitting configuration succeeds in bypassing the problem. The 3D cases are similar. There might be some extreme cases for which no conformation succeeds in bypassing the problem. In that case, we perform a local linear approximation of the discretized normal level set. But these insolvable cases would definitely present very twisted level sets and a mesh refinement would surely solve the problem.

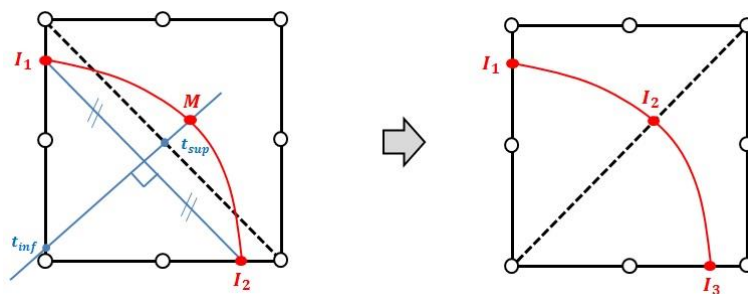


Figure 25: a case of double cancelation of the normal level set along the internal edge of a quadrangle. The middle node M between I_1 and I_2 is found out of bounds (left). The other configuration is selected to split the quadrangle into two primary triangles, bypassing the problem (right)

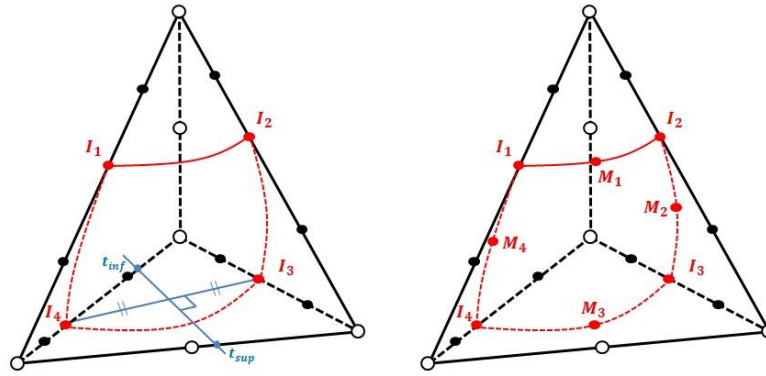


Figure 26: determination of the middle node between I_3 and I_4 (left) and the primary tetrahedron with the second type middle nodes $M_{1,2,3,4}$ (right).

3rd type of middle nodes

The third type middle nodes are located on the triangular faces of the primary simplex cells intersected by the discontinuity. These faces are split into a triangle and a quadrangle (see Figure 27 left). The quadrangle $N_1N_2I_2I_1$ is supposed to be split into two triangles. For this aim, we determine the middle node between I_2 and the opposite vertex node N_1 . Whenever possible, we choose the middle of the segment $[I_2N_1]$ as the middle node C which generates one twisted sub-triangles instead of two (see Figure 27). But this choice may not be convenient when the discontinuity is highly curved. On Figure 27, the case depicted at the bottom left generates distorted sub-triangles if the middle node C is chosen as the middle of the segment $[I_2N_1]$. In that case, we choose the middle of the quadratic quadrangle for C . In order to detect these situations, we compute the tangent vector \vec{j} to the three node segment I_2MI_2 at the point I_2 (see Figure 27). Depending on its position compared to the tangent vector \vec{k} to the segment I_2N_1 , we choose a different type of construction for the localization of the middle node C .

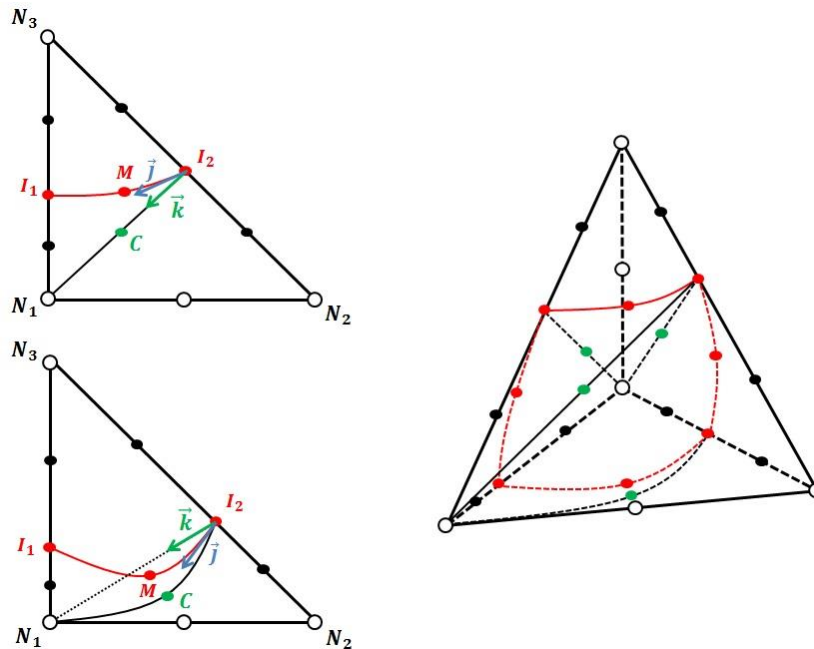


Figure 27: determination of the third type middle node: C is chosen as the middle of the segment $[I_2N_1]$ (top left) and C is chosen as the center of quadrangle $N_1N_2I_2I_1$ (bottom left). The primary tetrahedron with the third type middle nodes in green (right).

4th type of middle nodes

The last type of middle nodes are the ones located in the middle of the quadrangular faces approximating the iso-zero of the tangential level set. On Figure 28, we look for the middle of quadrangle $I_1I_2I_3I_4$. The middle of segment $[I_2I_4]$ is chosen as an initial guess and for the unit vector \vec{u} carrying the straight search path, we choose the normalized gradient of the normal level set. Indeed, the gradient of the tangential level set gives the normal direction to the interface, it forms the best search direction. Once again, first we determine the authorized interval $[t_{inf}, t_{sup}]$, corresponding to the intersection between the search direction and the primary tetrahedron. If the middle node is found out of bounds, we go back to the splitting into primary simplex cells and select another configuration.

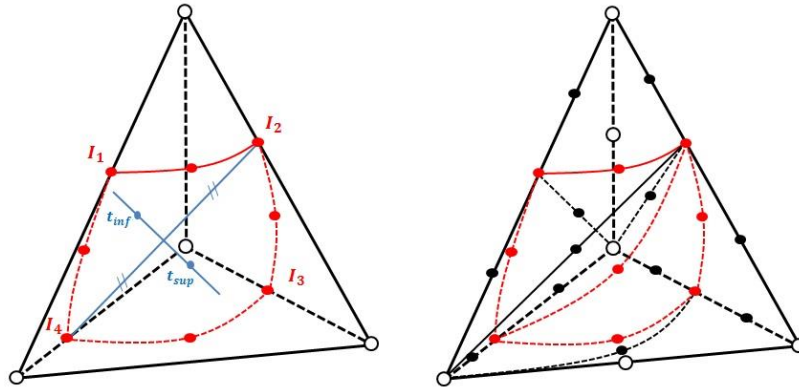


Figure 28: determination of the fourth type middle node in the interval $[t_{inf}, t_{sup}]$ (left) and the primary tetrahedron with the four types of middle nodes (right).

Now that all the nodes of the integration subcells have been determined, the primary simplex cells are split into integration subcells according to the cutting configurations depicted in Section 2.5. The integration subcells are labeled with the sign of the normal level set, depending on which side of the discontinuity they belong to. The nodes of the integration subcells coinciding with the iso-zero of the normal level set are also specifically labeled.

2.6 An extension to multi-cracked models

The splitting into integration subcells can be extended to multi-cracked models, in particular to branching discontinuities. For the elements crossed by several discontinuities, the procedure depicted for one interface is performed iteratively.

To each arbitrary discontinuity is associated a normal level set field. For branching discontinuities, the branched discontinuities are defined only on one side of a main discontinuity. On Figure 29, the second discontinuity (in green) is branched on the first discontinuity (in red) and defined only in the domain $\{l s n_1 > 0\}$. The third one (in blue) is branched on the second one, which was branched on the first one. Thus it is defined only in the domain $\{l s n_1 > 0\} \cap \{l s n_2 > 0\}$.

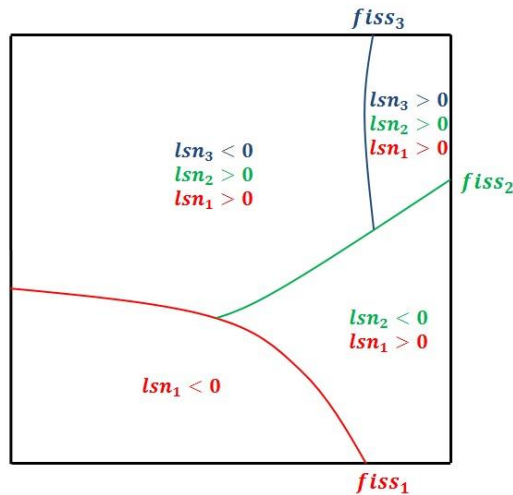


Figure 29: definition of branched arbitrary discontinuities by means of normal level set.

For an element crossed by several discontinuities, we proceed one discontinuity after another. For the first discontinuity, the procedure depicted above is normally performed. We end up with a set of integration subcells fitting the first discontinuity. For the cutting with respect to the second discontinuity, we proceed in the same way as for the first discontinuity except that the set of primary simplex cells is replaced by the set of integration subcells we obtained during the first cutting procedure. The difference lies in the fact that the edges of the integration subcells are not necessarily straight in the reference configuration of the parent element, contrarily to the edges of the primary simplex cells. In order to bypass this problem and apply the exact same procedure as for a single discontinuity, we work on the reference configuration of the integration subcells to perform the cutting procedure with respect to the second discontinuity. In this way, we always end up cutting simplex cells with straight edges. The procedure is summarized in Figure 30. The quadrangle is crossed by a main discontinuity (in red) and a branched discontinuity (in green). We work in the reference configuration of this quadrangle to perform the cutting with respect to the first discontinuity. Then we loop over the resulting integration sub-triangles. For each of them, we work in the reference configuration of the sub-triangle and apply the procedure depicted in Sections 2.3 to 2.5.

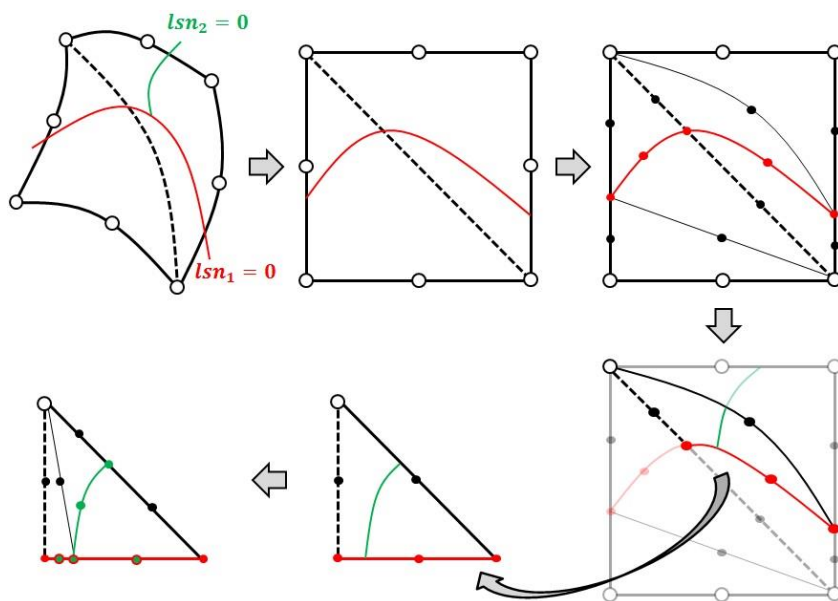


Figure 30: cutting procedure for a quadrangle including a branching discontinuity.

On Figure 31, we observe the final integration sub-triangles obtained for the quadrangle, fitting both discontinuities. The nodes coinciding with the first discontinuity appear in red and the nodes coinciding with the second discontinuity appear in green. The junction point is green and circled in red.

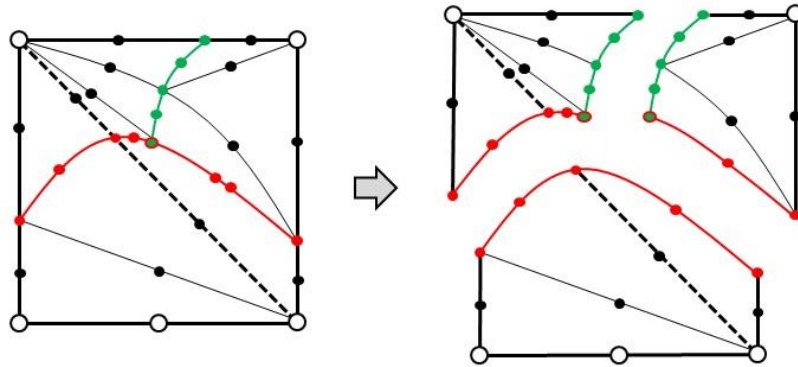


Figure 31: final integration sub-triangles for the quadrangle.

Similarly to the single discontinuity case, the integration subcells are labeled with the signs of the normal level sets, depending on which side of the discontinuities they belong to. When the level set is not defined (in the case of branching discontinuities), the default sign is 0. The nodes of the integration subcells coinciding with the iso-zero of the tangential level sets are also specifically labeled. In particular, the junction points are labeled for both discontinuities.

2.7 Integration over the sub-domains

Now that we split the elements crossed by arbitrary discontinuities into sets of integration sub-cells, making up an accurate quadratic approximation of the sub-domains, we are in position to realize a domain integration. Indeed, the integration subcells were labeled depending on which sub-domain they belong to. We can thus recover exclusively the set of elements and subcells approximating any sub-domain. For the volume integration over the integration subcells, we use the Standard Gauss integration techniques. The Gauss integration schemes we use are summarized hereunder. According to [Dathe**], the use of order 3 Gauss integration schemes in the linear case and order 5 Gauss integration schemes in the quadratic case offers satisfactory accuracy for the integration of the left side terms of equation (1) over tetrahedral elements.

		Subcell	Number of Gauss integration points per subcell	order
2D	Linear case	3 node triangle	3	3
	Quadratic case	6 node triangle	6	5
3D	Linear case	4 node tetrahedron	5	3
	Quadratic case	10 node tetrahedron	16	5

Table 2: Gauss integration schemes for the integration subcells.

Part 3: integration on the crack surface

Now that we have designed the volumetric integration subcells in the elements crossed by arbitrary discontinuities, we have to obtain a reconstructed iso-zero of the normal level set to achieve integration on the crack. Indeed, the integration on the crack surface is useful for various applications of the extended finite element method. For instance, the contact efforts preventing the interpenetration between adjacent subdomains are usually integrated on the crack surface [Géniaut**][Pierrès**] as well as the cohesive efforts when the propagation of the crack is governed by a cohesive zone model [Ferté2**]. The consideration of a fluid pressure on the fracture walls in the case hydraulic fractures [Faivre**] also requires a material approximation of the zero-level set . The volumetric subcells were built in order to offer a quadratic approximation of the domains separated by the arbitrary discontinuities. Thus the approximation of the different domains includes an approximation of the discontinuities. We use the faces of the subcells that coincide with the discontinuities to build the reconstructed implicit interfaces. When the element includes a piece of a crack front, a final cutting procedure with respect to the tangential level set is necessary to obtain a reconstructed approximation of the crack surface and front.

3.1 Overview of the recovery of the contact faces

The set of faces approximating an implicit interface is designed hereafter the contact faces as one of their main use is the integration of the contact equations between adjacent sub-domains. We impose an absolute fit between the integration subcells and the contact faces. The fit is clear for the contact faces that are directly recovered from the integration subcells. But when an element includes a piece of the crack front, the contact faces recovered from the integration subcells must be cut with respect to the normal level set associated to the crack front. In order to maintain the fit with the integration subcells and the quadratic accuracy throughout this final cutting procedure, we use the tools depicted in Part 1. In the end, the cutting procedure for the contact faces relies on the cutting procedure performed for the integration subcells.

3.2 Contact faces for an element entirely cut by an interface

First of all, we focus on the elements entirely crossed by a single arbitrary discontinuity. These elements have already been split up into tetrahedral integration subcells (triangular integration subcells in the 2D case). Some faces of these subcells correspond to the quadratic approximation of the arbitrary discontinuity. The vertex nodes of these faces were specially labeled when we shaped the integration subcells because they correspond to the intersections between the discontinuity and the edges of the primary simplex cells. As a consequence, we only need to loop over the faces of the integration subcells and select those whose 3 vertex nodes are labelled as intersection nodes (see Figures 32 and 33). In order to get each contact face exactly once, the choice is made to extract them only from the subcells labeled with a negative signed distance function.

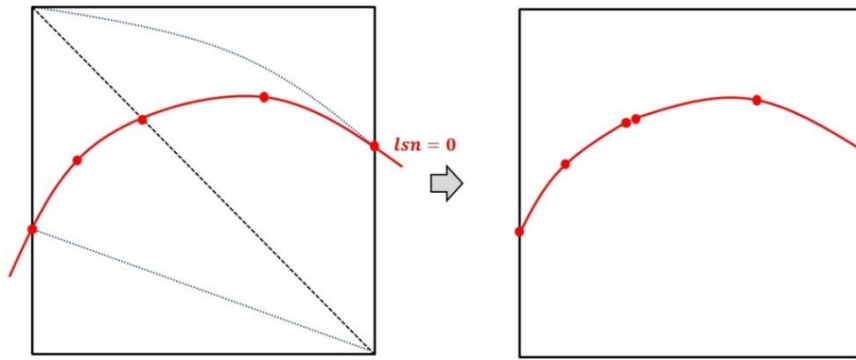


Figure 32: extracting the contact faces from a quadrangular element: triangular integration subcells (left) and resulting contact faces (right).

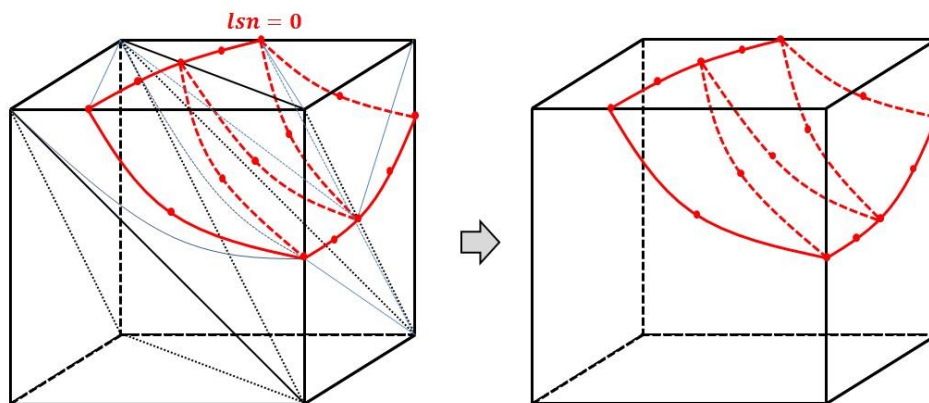


Figure 33: extracting the contact faces from a hexahedral element: tetrahedral integration subcells (left) and resulting triangular contact faces (right).

3.3 Contact faces for an element which includes a piece of the crack front

For the elements that include a piece of the crack front, another complementary cutting procedure is necessary. Indeed, the design of the integration subcells detailed in part 1 was realized regardless of the crack front. All the elements that intersect the discontinuity were split with respect to the normal level set. But in order to perform an integration over the surface of discontinuity, it is necessary to have contact faces that match the crack front depicted by the iso-zero of the tangential level set.

The first step consists in extracting the preliminary contact faces in the same manner as in the previous section. Then, we compute the value of the tangential level set at the vertex nodes of these preliminary contact faces and classify them into 3 groups:

- the contact faces whose 3 vertex nodes satisfy $lst \geq 0$ form the group 1. This group is out of bounds, its elements will be eliminated.
- the contact faces whose 3 vertex nodes satisfy $lst \leq 0$ form the group 2. We keep this entire group for the final contact faces.
- the remaining contact faces that are necessarily intersected by the iso-zero of the tangential level set form the group 3. These contact faces need a further cutting out.

In the 2D case, the intersection between the iso-zero of the tangential level set and a preliminary contact face of group 3 is determined with algorithm 1 in the reference configuration of the contact face (so that the contact face is a straight segment). Then, the new middle node of the contact face

(type 1 middle node) is mapped from the reference configuration of the segment to the contact face in the parent element. The procedure is summed up in Figure 34.

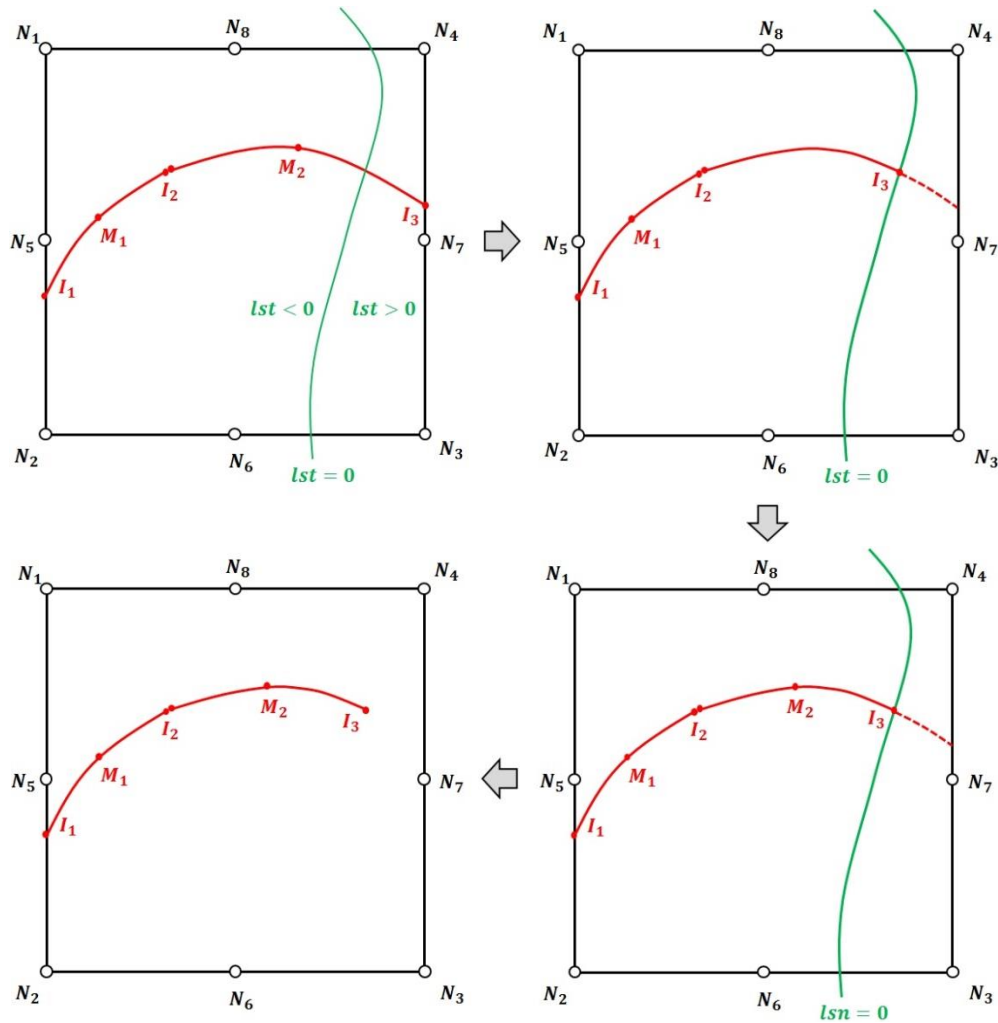


Figure 34: extracting the contact faces from a quadrangle that includes the crack front.
 1 - amongst the preliminary contact faces, we keep those whose vertex nodes satisfy $lst \leq 0$ and select the ones intersected by the iso-zero of the tangential level set for the cutting procedure (top left),
 2 - we determine the intersection between the contact face and the iso-zero of the tangential level set (top right),
 3 - we determine the new position of the middle node of the intersected contact face (bottom right),
 4 - we get the final contact faces (bottom left).

The 3D case requires more attention, amongst the triangular contact faces of group 3, we distinguish 3 cutting configurations:

- the contact faces that have one vertex node satisfying $lst = 0$ (configuration 1)
- the contact faces that have two vertex nodes satisfying $lst < 0$ (configuration 2)
- the contact faces that have one vertex node satisfying $lst < 0$ (configuration 3)

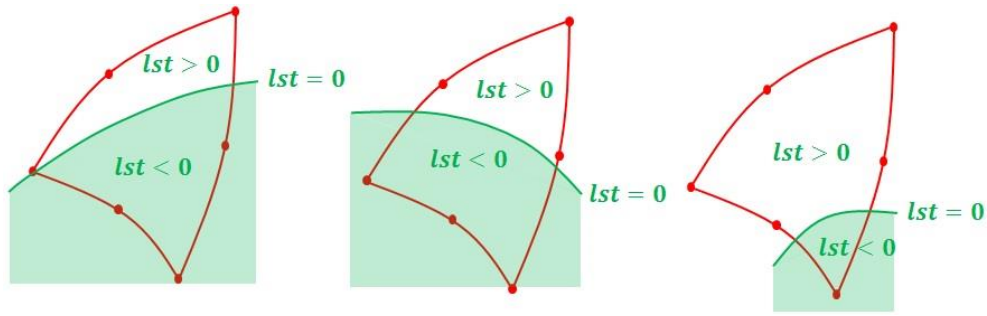


Figure 35: the 3 cutting configurations for the contact faces.

The criterion we use to determine whether a contact face is entirely in the domain $\{\mathbf{x} | lst(\mathbf{x}) \leq 0\}$ or entirely in the domain $\{\mathbf{x} | lst(\mathbf{x}) \geq 0\}$ or intersected by the iso-zero of the tangential level set only lies on the vertex nodes of the contact faces. We might then face situations for which the values of the tangential level set at the middle nodes of the contact faces contradict this classification, giving away a double cancelation of the tangential level set along the edge of the contact face. At first, we assume these situations do not occur. The last part of this section is dedicated to the treatment of these annoying cases.

In the following, we explain the cutting procedure for the second cutting configuration. The task consists in cutting a triangle with respect to the tangential level set. In the reference space of the parent element, this triangle is not necessarily plane as it approximates the iso-zero of the normal level set. In order to be reduced to the cutting procedure depicted in Sections 2.3 to 2.5, we use the same ingredient as in the case of multi-cracked elements. We map the preliminary contact face with its associated reference triangle whose edges are straight. Then, we apply the classic procedure for the cutting of 2D triangular element (Figure 36).

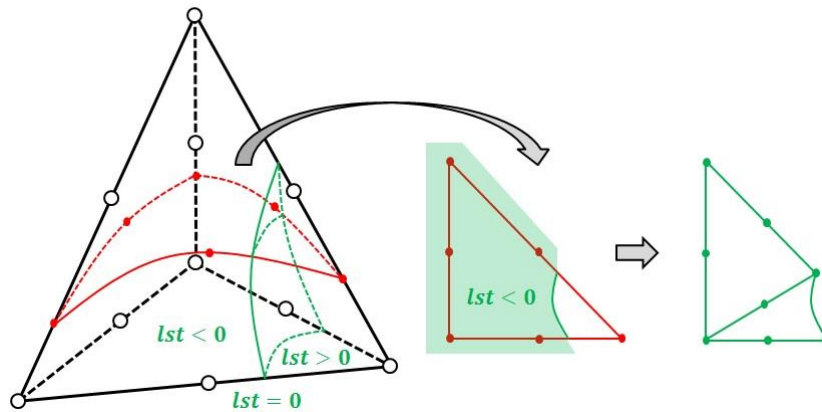


Figure 36: cutting a triangular primary contact face into two triangular final contact faces within a tetrahedral element.

On Figure 37, we observe the two final triangular contact faces in the primary tetrahedral element. They fit the iso-zero of the tangential level set.

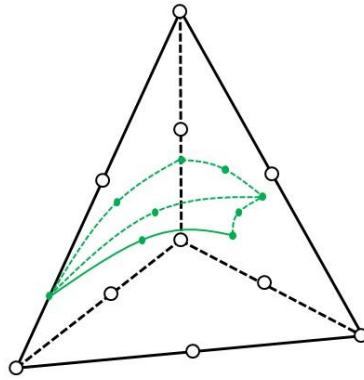


Figure 37: final contact faces in the primary tetrahedral element.

The cutting procedure is similar for the two other configurations. As depicted in Figure 38, they both give one final triangular contact face.

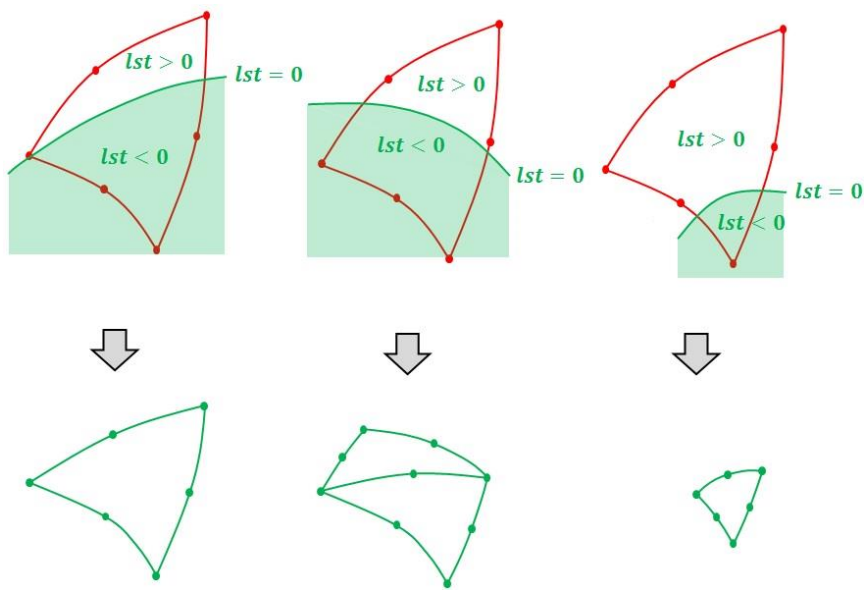


Figure 38: the final contact faces for the 3 cutting configurations.

Finally, we look at the situations of double cancelation of the tangential level set along an edge of the preliminary contact faces. On Figure 39, we observe the preliminary contact faces generated by a planar crack in a hexahedron.

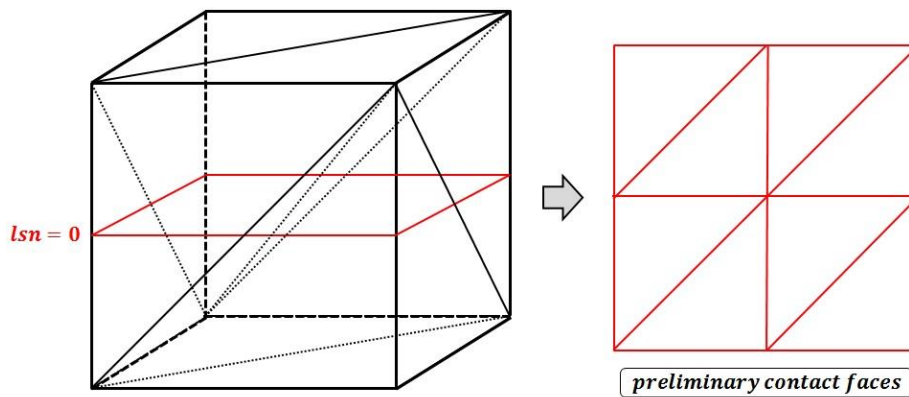


Figure 39: preliminary contact faces generated by a planar crack in a hexahedron.

The tangential level set cannot cancel twice on the edges of the preliminary contact faces that coincide with a face of the parent element because we previously performed the level set adjustment (Section 2.2). But on the edges of the preliminary contact faces that are internal to non-simplex parent element, we cannot prevent a potential double cancellation of the tangential level set. Two distinct situations may occur. The first situation is depicted in Figure 40. It corresponds to a double cancellation of the tangential level set along the edge of a preliminary contact face due to a local high convexity of the crack front. This situation is detected when we look for the middle node M between the two intersection points I_1 and I_2 . The middle node M is found out of the bounds since it exceeds the upper limit t_{sup} . Contrarily to what was done in Part 1 for the integration subcells, we authorize the middle node M to go over the upper limit t_{sup} . Indeed, the nearby preliminary contact face affected by the intrusion stands in group 1 and will be eliminated. In this way, we obtain an accurate quadratic approximation of the crack surface.

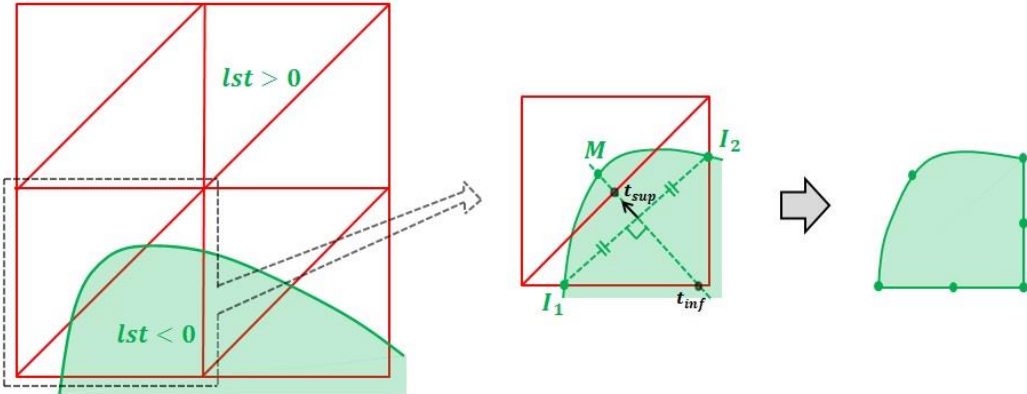


Figure 40: situation 1.

The second situation is depicted in Figure 41. It corresponds to a double cancellation of the normal level set along the edge of a preliminary contact face due to a local high concavity of the crack front. This situation is detected when we look for the middle node M between the two intersection points I_1 and I_2 . The middle node M is found out of the bounds since it exceeds the lower limit t_{inf} . In this case, we cannot allow the intrusion of the middle node M in the nearby preliminary contact face because it would bring distorted contact faces. One solution consists in making a local linear approximation of the tangential level set. The position of the middle node M is chosen as the initial guess of algorithm 1. In the reference space of the parent element, it is then the middle of the segment $[I_1I_2]$.

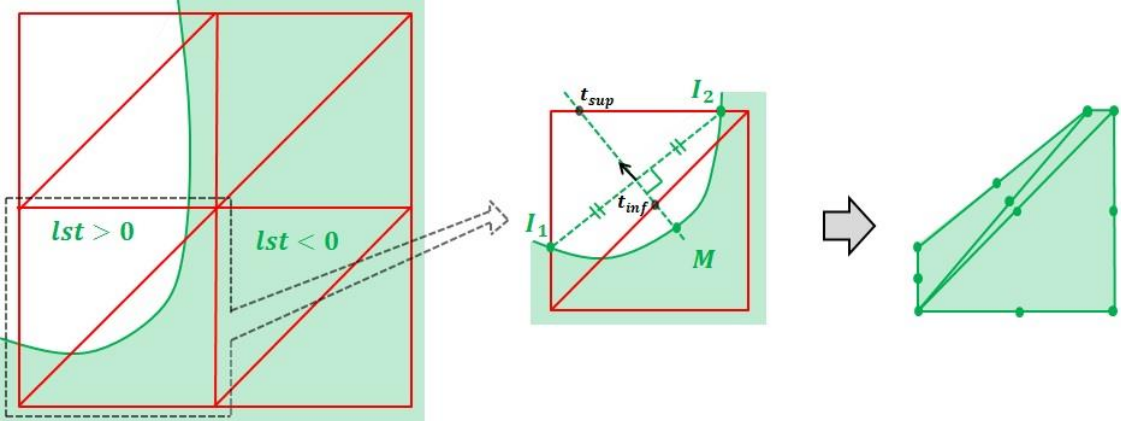


Figure 41: situation 2.

This local linear approximation of the tangential level set seriously degrades the accuracy of our integration procedure. By all means, we would like to avoid it. The solution consists in going back to the very beginning of the cutting procedure for the parent element. We try the other eligible configurations for the partitioning of the parent element into primary simplex cells (Section 2.3) until no double cancelation of the level set is recorded. As depicted in Figure 42, these other configurations generate different primary simplex cells and different patterns for the preliminary contact faces, likely to bypass the problem depicted in Figure 41.

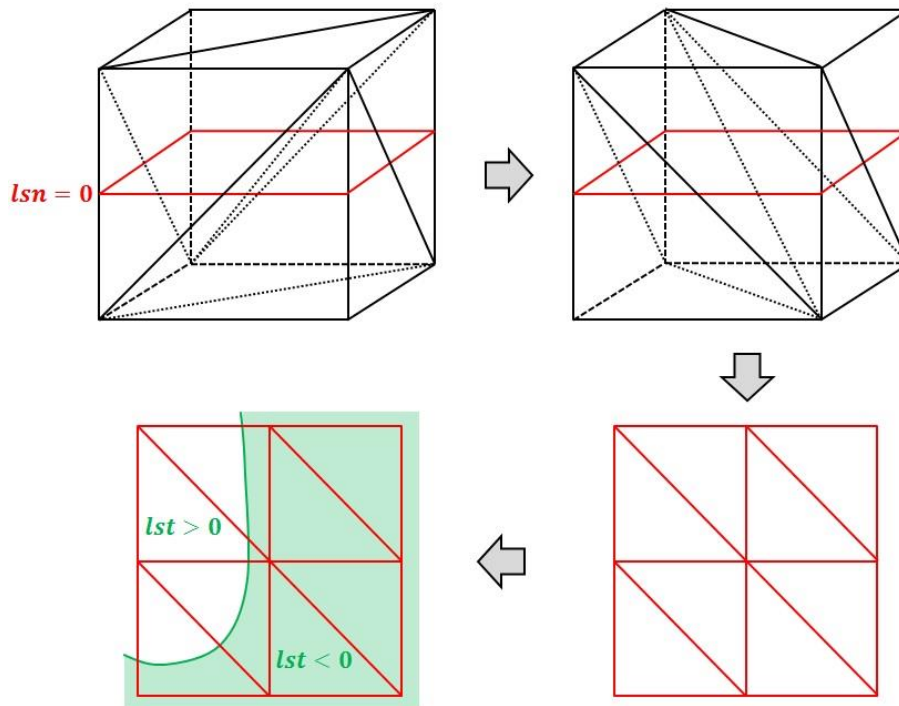


Figure 42: treatment of situation 2.

- 1 – first and previous configuration for the primary simplex cells (top left),
- 2 - another configuration of the primary simplex cells (top right),
- 3 - the new configuration of the primary simplex cells generates a new pattern of preliminary contact faces (bottom right),
- 4 - the new pattern of preliminary contact faces allows us to accurately approximate the crack surface in the vicinity of the crack front (bottom left).

As for the design of the integration subcells, we can imagine there might be some extreme cases for which no configuration succeeds in bypassing the problem of double cancelation of the level sets along an edge of a primary integration subcell and along an edge of a preliminary contact face. In that case, we perform local linear approximations of the discretized level set. But these insolvable cases would definitely present very twisted level sets. A mesh refinement would surely solve the problem.

Please note that these contact faces also allow us to obtain an accurate quadratic reconstruction of the crack front as a chain of 3 node segments for 3D models. It may be useful for fracture mechanics post-processing.

3.4 Contact faces for multi-cracked elements

In this section, we detail the recovery of the contact faces for multi-cracked elements. We only consider multi-cracked elements that do not include crack fronts. In order to recover the contact

faces of multi-cracked elements that include crack fronts, we would have to combine the procedure depicted in this section with the procedure depicted in the previous section.

The multi-cracked elements have already been split up into tetrahedral integration subcells (triangular integration subcells in the 2D case). As for the single-cracked elements, some faces of these subcells correspond to the quadratic approximation of arbitrary discontinuities. The vertex nodes of these faces were specifically labeled distinctly for each discontinuity when we shaped the integration subcells. In particular, the junction points were labelled for the two discontinuities forming the junction. So for each discontinuity, we proceed exactly as for a single discontinuity. We loop over the faces of the integration subcells and select those whose 3 vertex nodes are labelled as intersection nodes for the current discontinuity. In the case of a single discontinuity, the choice was made to loop only over the integration subcells labeled with a negative signed distance function in order to recover the contact faces exactly once. For multi-cracked elements, it is necessary to modify this rule for the main discontinuities. The main discontinuities are defined as the discontinuities on which another discontinuity is branched. For these discontinuities, we decide to recover the contact faces from the integration subcells labeled whether with a negative or a positive sign distance function depending on the position of the branched interface. We chose the sign corresponding to the side where the interface is branched. Then we loop over the integration subcells labeled with the corresponding sign to recover the contact faces for the main discontinuity. On Figure 43, we observe a quadrangular element including a branching discontinuity and the resulting triangular integration subcells fitting both discontinuities. The main discontinuity and the branched discontinuity delimit three distinct domains Ω_1 , Ω_2 and Ω_3 over the quadrangular element. On Figure 44, we observe the resulting contact faces when we extract the contact faces for the main discontinuity from the integration subcells labeled with a sign that does not correspond to the branching (left) and that does correspond to the branching discontinuity (right). In the first case, we end up with two contact faces for the main interface as if the branching was not existing. Then the contact face whose end nodes are I_2 and I_3 does not fit the junction as it was extracted from an integration subcell which is not on the side of the branching discontinuity. In the second case, we end up with 3 contact faces for the main interface. They fit the discontinuity junction because they were extracted from integration subcells located on the side of the branched discontinuity. It is essential to obtain contact faces that fit both discontinuities. For instance, we may want to perform a surface integration exclusively on the boundary separating Ω_1 from Ω_3 . This is feasible provided that the contact faces for the main discontinuity fit the branched discontinuity.

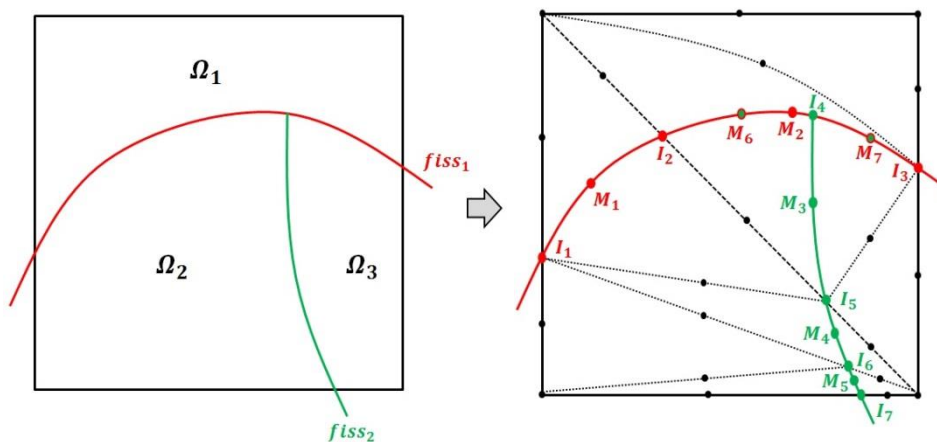


Figure 43: a quadrangular element including a branching discontinuity (left) and the resulting integration subcells (right).

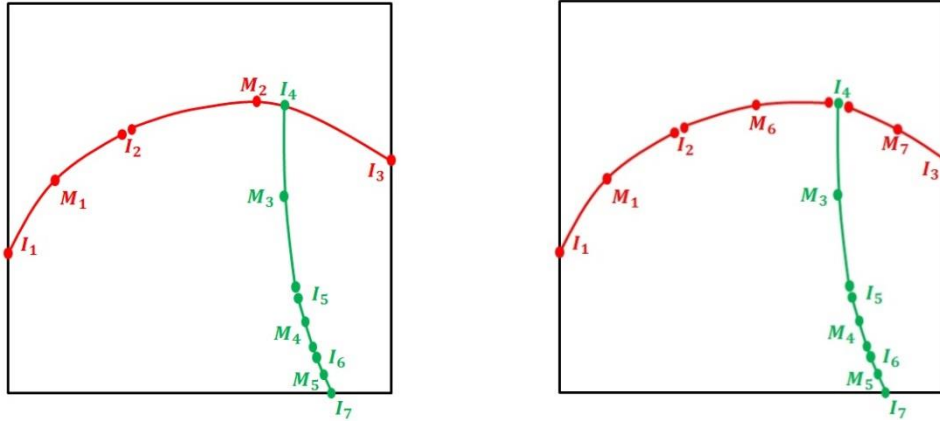


Figure 44 : the resulting contact faces when we extract the contact faces of the main interface from the integration subcells located above (left) and below (right) the main interface.

When several discontinuities are branched on the same main discontinuity within an element, we may not be able to properly recover contact faces fitting all the junctions. On Figure 45 (left), we observe a quadrangular element crossed by a main interface (red) with two branched discontinuities. The discontinuities are branched on both sides of the main interface. We also observe the resulting integration subcells. Neither the integration subcells located below the main interface neither the ones located above generate contact faces fitting both junctions for the main interface. This configuration can not be solved so that we forbid the presence of two distinct fracture junctions within the same element. This can easily be avoided by refining the mesh. However, we can still have branched fractures within the same element if the different junctions coincide (Figure 45 right).

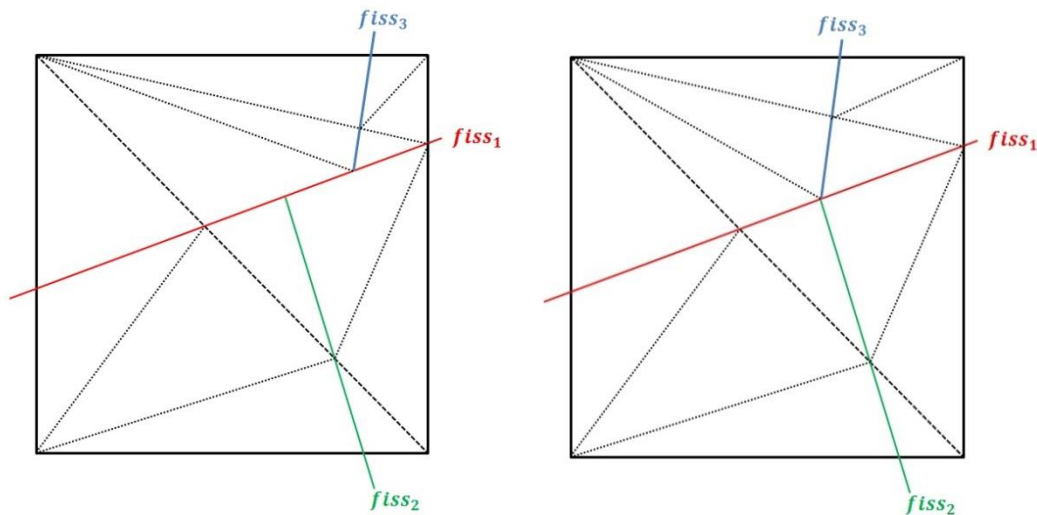


Figure 45: a quadrangular element including two discontinuity junctions and the resulting integration subcells in the case of not coinciding junctions (left) and coinciding junctions (right).

3.5 Integration over the contact faces

Now that we have built contact faces that accurately approximate the immersed boundaries, we are in position to perform surface integrations over the sub-domain boundaries. For this aim, we use classical Gauss integration techniques. For each contact face, the positions of the Gauss integration points are interpolated from the position of the nodes of the contact face. Furthermore, the surface integration often requires the normal direction to the contact face (for instance to take into account a fluid pressure in the fracture). So for each Gauss point we build a unit normal vector to the contact

face (oriented along the gradient of the normal level set) from the position of the nodes of the contact face (see Figure 46).

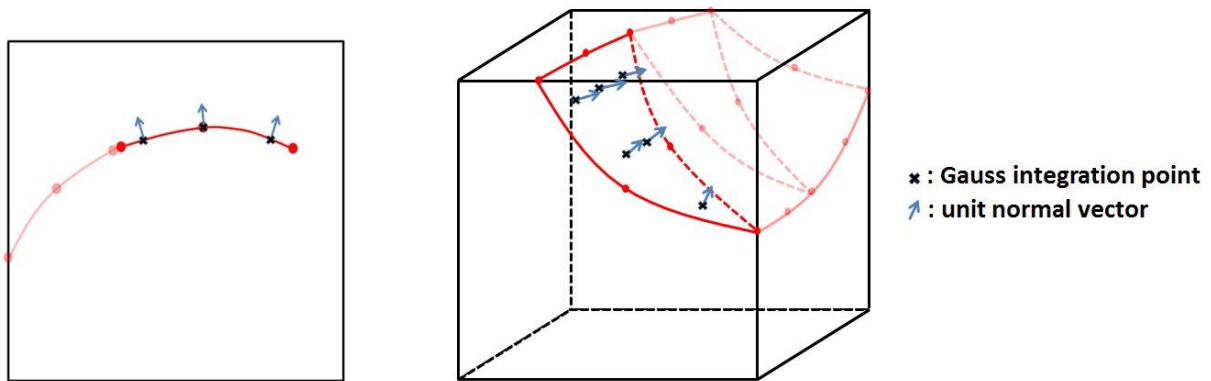


Figure 46: integration over the curved contact faces of a quadrangle (left) and a hexahedron (right).

The Gauss integration schemes used for the surface integration are summarized hereunder. According to [Dathe**], the use of order 3 Gauss integration schemes in the linear case and order 5 Gauss integration schemes in the quadratic case offers satisfactory accuracy for the integration of the right side terms of equation (1) over triangular elements.

		Contact face type	Number of Gauss integration points per contact face	order
2D	Linear case	2 node segment	2	3
	Quadratic case	3 node segment	3	5
3D	Linear case	3 node triangle	3	3
	Quadratic case	6 node triangle	6	5

Table 3: Gauss integration schemes for the contact faces.

Part 4: validation of the integration method

In the following discussion, we present some numerical results in order to illustrate and validate the accuracy and robustness of the integration method detailed in the first two sections. In particular, we perform several convergence analyses. The convergence rates we get are in accordance with the theory. First of all, we present the XFEM formulation we combined to our integration procedure to perform the numerical tests.

4.1 Description of the XFEM formulation used

In the literature there are many formulations to model strong discontinuities in continuous media. In the paper of Ndeffo et al. [Ndeffo**], a thorough analysis has been made and numerical issues have been investigated, at least concerning quadratic elements. It has been established that partition

of unity based formulations don't behave well in asymptotic configurations (when the discontinuity gets close to the nodes of the approximation mesh). Therefore, conditioning and accuracy issues need a special care when modeling higher order strong discontinuities. Hence, Ndeffo et al. suggests a convenient formulation to deal with the condition number swift increase. In this section, we extend the suggested formulation to the case of branching discontinuities. Before describing more precisely the aforementioned formulation, let's focus on the definition of branching discontinuities. In the literature, there are two methods to define branched cracks:

- The use of sign fields called likewise "junction" functions: there are encountered in the framework of X-FEM and level-sets [Daux**].
- The use of non-overlapping domains in the "neighborhood" of the branched discontinuity: there are encountered in the framework of GFEM [Reno**].

The X-FEM defines iteratively the "junction" functions based on level-sets, to represent the branched crack kinematics. GFEM fits for the description of branching discontinuity when the information about domains is available; typically, in the case of polycrystals modeling, where partitions of the whole domain are well labeled [Poly**]. As we use level-sets to model discontinuity in this paper, an X-FEM description is more convenient.

However, the X-FEM enrichment functions perform poorly in terms of conditioning and accuracy for higher order elements, as shown in [Ndeffo**]. Formulations based on the partition in domains [Hansbo**],[Reno**], have a better numerical behavior. Thus our enrichment strategy should combine both features: the X-FEM its convenience and the formulations based on domains partitioning to deal with conditioning issues. As explained in [Ndeffo**], these two approaches are intermingled. Hence, we can switch from the description of branched discontinuities using level sets to the description using domain partitioning, more fitted for higher order elements. Before solving afore-mentioned conditioning issues, let's stress again on the definition and assembling of X-FEM junction d.o.f.

Definition X-FEM/GFEM approximation spaces

In the case of multiple cracks, following the notations of equation (2), the GFEM/X-FEM approximation of the displacement field could be summarized as follows:

$$u_h(x) = \sum_{i \in N/N_g} \varphi_i(x) u_i + \sum_{j \in N_g} \sum_{k=1, \text{card}_j} \varphi_j(x) \tilde{H}_{j,k}(x) a_{j,k}$$

where N_g represents the set of enriched nodes; N/N_g the set of non enriched nodes; card_j the node-wise number of d.o.f. for an enriched node, which depends of the number cracks intersecting the support of the node, and $a_{j,k}$ the related d.o.f. $\tilde{H}_{j,k}$ is a generic notation for whether X-FEM or GFEM node-wise enrichment functions defined in the table below:

	Single crack	Single junction	Double junction
X-FEM	$\{\varphi, H_1\varphi\}$	$\{\varphi, H_1\varphi, H_2\varphi\}$	$\{\varphi, H_1\varphi, H_2\varphi, H_3\varphi\}$
GFEM	$\{\chi_{\Omega_1}\varphi, \chi_{\Omega_2}\varphi\}$	$\{\chi_{\Omega_1}\varphi, \chi_{\Omega_2}\varphi, \chi_{\Omega_3}\varphi\}$	$\{\chi_{\Omega_1}\varphi, \chi_{\Omega_2}\varphi, \chi_{\Omega_3}\varphi, \chi_{\Omega_4}\varphi\}$
NEW	$\{\varphi, \chi_{\Omega_2}\varphi\}$	$\{\varphi, \chi_{\Omega_2}\varphi, \chi_{\Omega_3}\varphi\}$	$\{\varphi, \chi_{\Omega_2}\varphi, \chi_{\Omega_3}\varphi, \chi_{\Omega_4}\varphi\}$

Table 4 : comparison between node-wise enrichment functions in the case of X-FEM, GFEM and the NEW enrichment strategy proposed; where H_i represents a sign-function and χ_{Ω_i} the characteristic function of domain Ω_i .

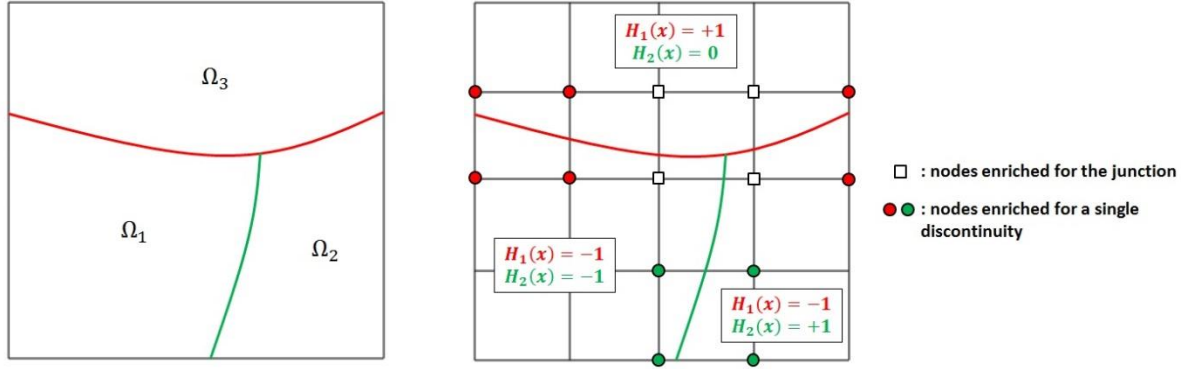


Figure 47: Heaviside functions and domains definitions. In the case of junctions, only the nodes whose support is intersected by multiple cracks are enriched with junction enrichment functions.

The X-FEM and GFEM formulations describe exactly the same approximation space and so does the new enrichment strategy proposed (see table 4), which is an extension to junctions of the formulation proposed in [Ndeffo**]. The motivation for this new enrichment strategy is the following. When the crack surface gets close to the nodes of the mesh, condition number soars with quadratic elements. Thus, a special treatment is needed to improve the numerical behavior of the approximation space. Ndeffo et al suggested that X-FEM signs functions performed rather poorly and may lead to bad results. Therefore, X-FEM couldn't be used directly even if the X-FEM junction functions are more convenient when level-sets information is used. So, we considered a reshape of X-FEM approximation to solve those conditioning issues. The construction of the new enrichment suggested above follows the principle of complementary element or complementary nodes that is encountered also in the literature [AFEM**]. For the same arguments underlined in the paper of Ndeffo et al, this new formulation combines features of both X-FEM and GFEM:

- the convenience of the X-FEM for implementation: the X-FEM exploits fully the formalism of level-sets to describe branched discontinuities.
- the good numerical properties of Hansbo's formulation: standard shape functions and enrichment are almost orthogonal when the interface gets close to the nodes.

Dealing with X-FEM conditioning issue

Nevertheless, those enrichment functions need additional treatment given their asymptotic behavior, particularly because their domain of definition are inclined to cancel out when the crack surface gets close to the nodes. Therefore, a preconditioner is needed to scale-up the related degrees of freedom. Then, we use the simple pre-conditioner suggested in [Ndeffo**]. The matrix notation of the finite element problem $Ku = f$ is replaced by:

$$K'u' = f' \quad \text{with} \quad \begin{cases} K' = D_c K D_c \\ u = D_c u' \\ f' = D_c f \end{cases}$$

where D_c is defined as:

$$[D_c]_{i,i} = d / \sqrt{K_{i,i}} \quad \text{with} \quad d = \sqrt{\frac{\max(K_{i,i}) + \max(K_{i,i})}{2}}$$

Finally, given the coupling between quadratic enrichment functions, some redundant degrees of freedom have to be removed to prevent the condition number soaring. Thus, some middle nodes

shape functions are removed within the partition of unity set of functions. Our elimination strategy relies on the use of the “stiffness criterion” suggested by [Siavelis**].

The degree of freedom $a_{j,k}$ is eliminated if:

$$\frac{\int_{Supp(j) \cap \Omega_k} \|\varphi_{j,X}\|^2 d\Omega}{\int_{Supp(j)} \|\varphi_{j,X}\|^2 d\Omega} \leq 10^{-9}$$

Where $Supp(j)$ is the support of node j , $\varphi_{j,X}$ is the derivative of the classical shape function associated to node j in the global coordinate basis X and $\|\varphi_{j,X}\|$ its usual Euclidian norm L_2 .

Remark: the elimination criterion above has a negligible impact on the accuracy of the elements, as shown in section 4.6. Therefore, a sensitivity analysis around the threshold isn't relevant here.

From level-sets to local domains

The "complementary" domain enrichment detailed here requires partitioning into domains [Reno**], which cannot be extracted straightforwardly from level-sets information within elements. At the elementary level, one needs to label partitions of global domains intersecting the support of each node. In contrast, the X-FEM's branching relies on the local behavior of Heaviside-function [Daux**] rather than on the global domains. Then, for the new enrichment formulation we adopt, the local sign field information must be transposed into domain mapping. This is feasible with several loops over the elements to compare partitions of domain between different elements.

The use of this formulation ensures good performances in terms of conditioning for the numerical examples presented below. The following numerical tests are performed with *Code_Aster* (<http://web-code-aster.org>). First of all, we focus on the approximated geometry of the sub-domains. We perform some tests with very highly curved and non-convex zero-level set in order to appreciate the ability of our cutting procedure to handle exotic implicit geometries. Then we introduce error measurements for the approximated sub-domains. Convergence analyses on the resolution of the approximated interfaces are performed for 2D and 3D models including branching discontinuities. A convergence analysis is also performed on the resolution of a curved approximated crack front. Finally we proceed to nonlinear mechanical test including curved interfaces. Convergence analyses are performed for the displacement field and the energy error.

4.2 Preliminary examples

On Figure 49, we observe 3 examples which demonstrate the ability of the cutting procedure described in the first two parts to handle highly curved and non-convex interfaces. The three examples are performed with regular meshes formed of quadratic hexahedral elements. The equations associated to the analytical normal level set used are specified for each example. In particular, we observe the faces of the integration subcells that accurately represent the arbitrary boundaries of the different domains.

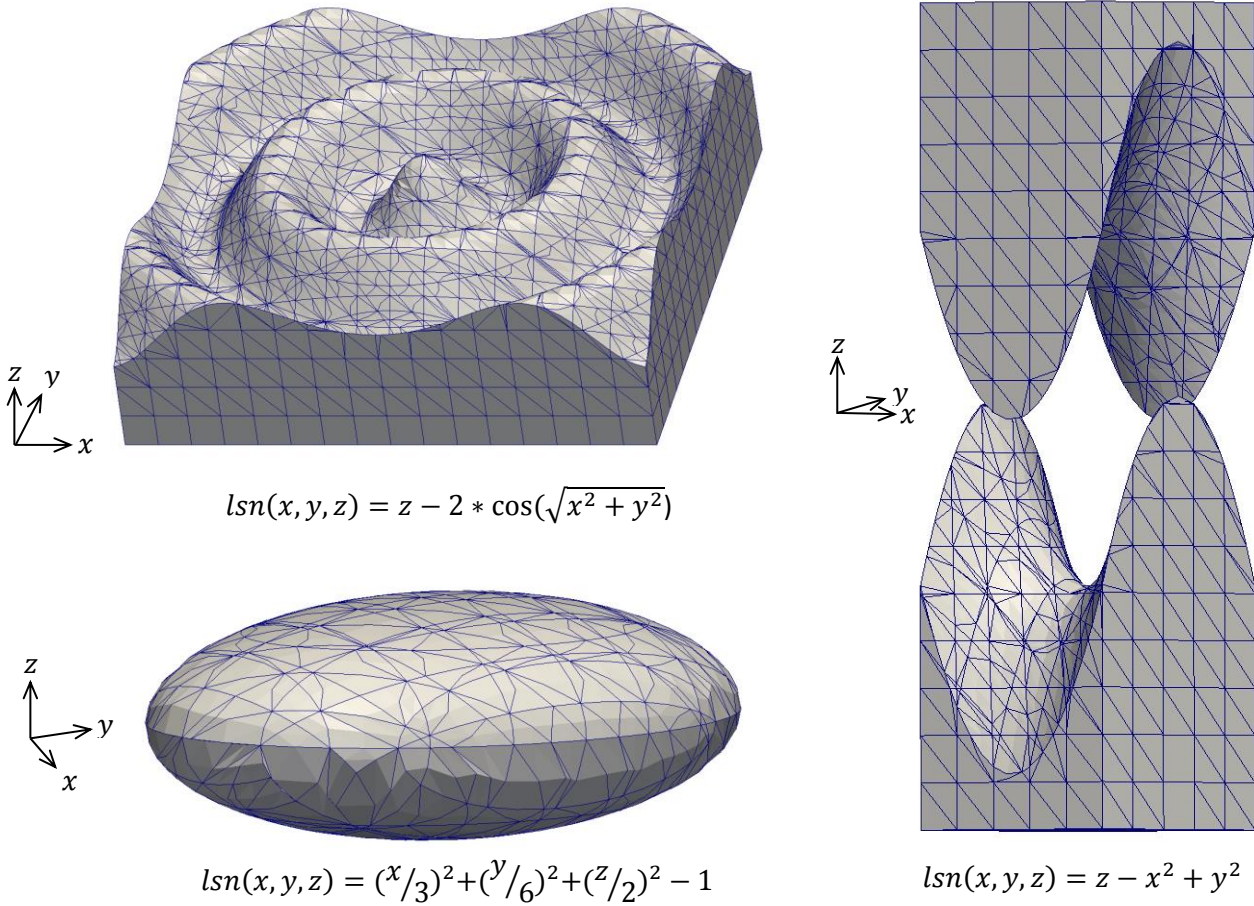


Figure 49: the concentric wavelets (top left), the ellipsoid (bottom left) and the saddle (right)

The quality of the approximation of the sub-domains seems satisfactory, given the fact that the meshes we use are relatively coarse compared to the curvature of the analytical normal level set we have chosen. In order to measure the accuracy of the approximated sub-domains, we compute the resolution ε defined as follow:

$$\varepsilon = \max_{x \in \Gamma} (\min_{\tilde{x} \in \tilde{\Gamma}} |x - \tilde{x}|)$$

where Γ is the analytical interface and $\tilde{\Gamma}$ is the approximated interface, or the set of contact faces approximating the interface. For each point x of the theoretical interface Γ , we compute the distance to the approximated interface $\min_{\tilde{x} \in \tilde{\Gamma}} |x - \tilde{x}|$. Finally, we take the maximum over $x \in \Gamma$. The resolution ε is then the maximal distance between the analytical interface and the approximated interface. The interface is thus said to be ε -resolved.

We may also use the relative volume error defined as follow:

$$volume\ error = \left| \frac{\Omega - \Omega_h}{\Omega} \right|$$

where Ω is the analytical volume of the sub-domain and Ω_h is the volume of the approximated sub-domain. It is clear from their definition that both error measures should have the same order of convergence. But as explained in [Ferté1**], higher rates of convergence are often observed for the

area error because of compensation phenomena. Therefore we favour the measurement of the resolution for the upcoming convergence analyses.

On the table below, we give the error measurements for the 3 examples depicted in Figure 48:

	Volume error (%)	Resolution
Concentric wavelets	0,32	0,0065
Ellipsoid	0,27	0,018
Saddle	0,076	0,0028

Table 5: Volume error and resolution for the examples of Figure 49.

In the following paragraph, we perform convergence analyses on the resolution for 2D and 3D models including branching discontinuities.

4.3 Convergence analyses for the approximated sub-domains

It is a 2D model of a square domain crossed by an horizontal interface and two vertical interfaces branched on the first one. The geometry and the boundary conditions are depicted on Figure 50.

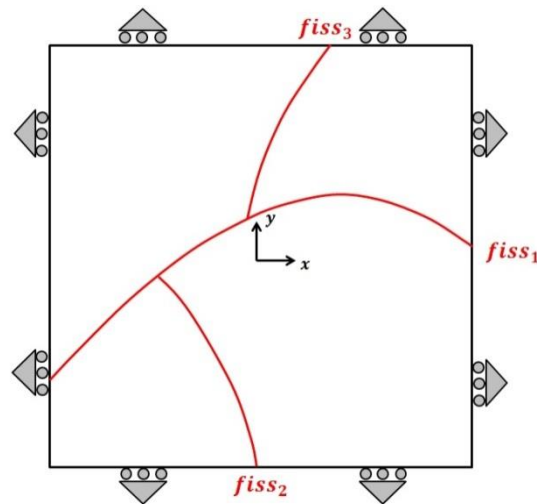


Figure 50: geometry and boundary conditions.

The dimension of the square is $10m \times 10m$. The origin of the cartesian coordinates system is located at the center of the square. The equations of the normal level sets associated to the 3 fractures are:

$$\left\{ \begin{array}{l} lsn_1(x, y) = y + 2,69 - 4,3 * \sin\left(\frac{x + 5}{4,2}\right) \\ lsn_2(x, y) = x + 5 * \exp\left(\frac{y - 2}{3}\right) \\ lsn_3(x, y) = \sqrt{(x - 10)^2 + (y + 2)^2} - 10,68 \end{array} \right.$$

The second fracture is defined in the domain $\{lsn_1(x, y) < 0\}$ and the third fracture is defined in the domain $\{lsn_1(x, y) > 0\}$. The loading consists in a uniform mechanical pressure p applied on both walls of the three fractures. The material we use is elastic and we expect an isotropic compression in each block. To perform the test, we use regular meshes formed of quadrangular elements, the

parameter h denoting the number of elements in each direction. We obtain the exact displacements for this problem as the solution we expect is linear in each block. This demonstrates the ability of our model to handle the volume and surface integrations in presence of fracture junctions.

According to [Ferté1**], the order of convergence we should obtain for the resolution ε are respectively 2 in the linear case and 3 in the quadratic case. Indeed we are able to appreciate the resolution error ε as the theoretical level set we have chosen are a sine curve, an exponential curve and a circular arc. It is not perfectly solved by the linear nor the quadratic cutting procedure we have used so far. On figure 51, we observe the zoomed deformed shape and the integration meshes in the quadratic case for $h = 4$ and $h = 16$.

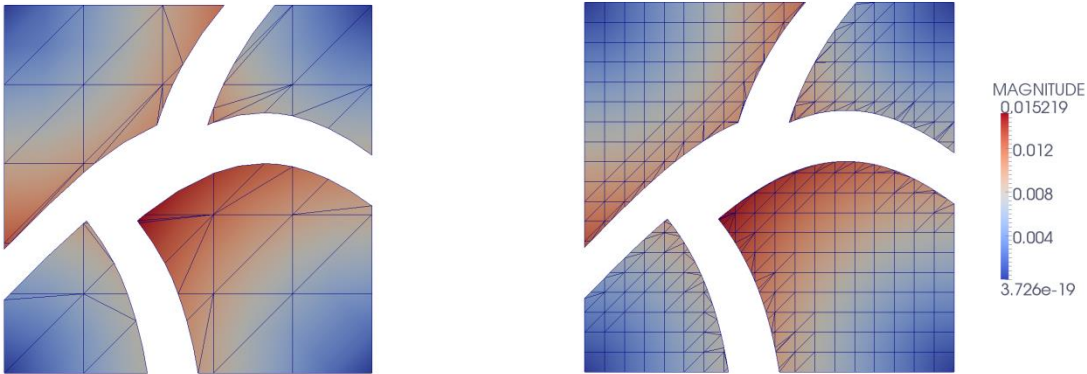
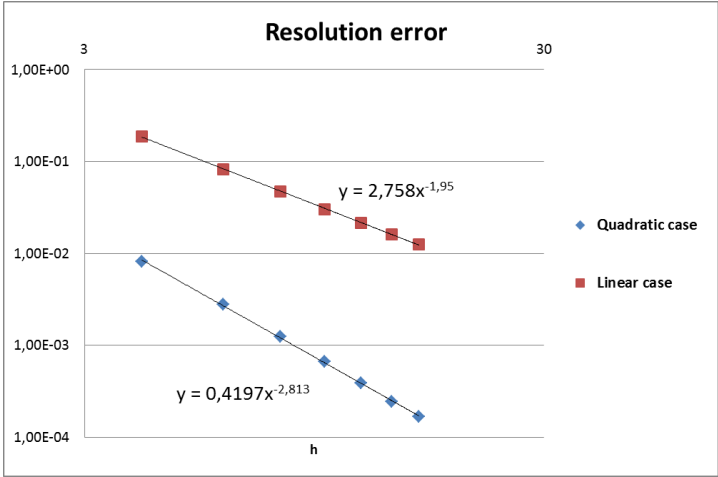


Figure 51: deformed mesh (x80) and displacement norm for $h = 4$ (left) and $h = 16$ (right) on the integration mesh for the quadratic case.

We plot the resolution for different values of h . The convergence rates we get for the linear and the quadratic model are close to the expected analytical rates. For the quadratic case, the error level is tremendously lower compared to the linear case. The quadratic cutting procedure we use proves to be highly efficient and accurate at approximating curved domain boundaries including junctions.



We perform a similar analysis in the extruded 3D case whose geometry is depicted in Figure 52. The thickness of the square is 2 m.

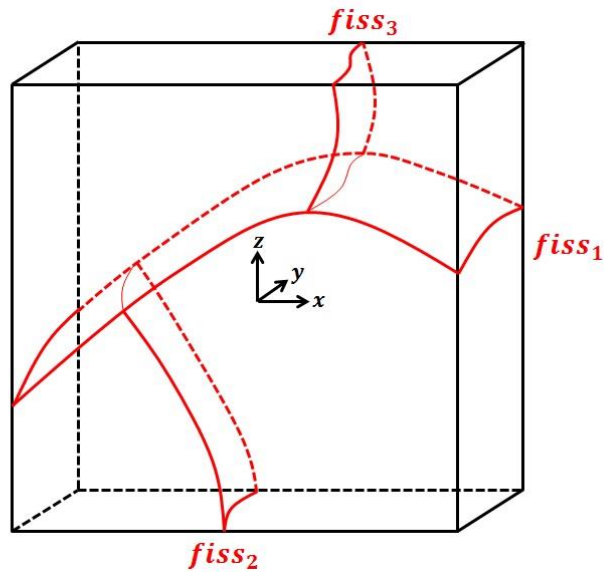


Figure 52: geometry of the 3D case.

The equations of the normal level sets associated to the 3 fractures are:

$$\left\{ \begin{array}{l} lsn_1(x, y, z) = z + 2,69 - 4,3 * \sin\left(\frac{x+5}{4,2}\right) + 0,12 * (y-1)^2 \\ lsn_2(x, y, z) = x + 5 * \exp\left(\frac{y-2}{3}\right) - 0,3 * y^2 \\ lsn_3(x, y, z) = \sqrt{(z-3)^2 + (x+5)^2} - 7,1 + 0,45 * \cos(y-1) \end{array} \right.$$

The loading and the boundary conditions are identical to the 2D case. We use regular meshes formed of hexahedral elements, the parameter h denoting the number of elements in directions (Ox) and (Oz) . The number of elements in the thickness of the square is $h/5$. On figure 53, we observe the zoomed deformed shape and the integration meshes in the quadratic case for $h = 5$ and $h = 10$. We can especially appreciate the design of the quadratic integration subcells whose faces coincide with the different interfaces.

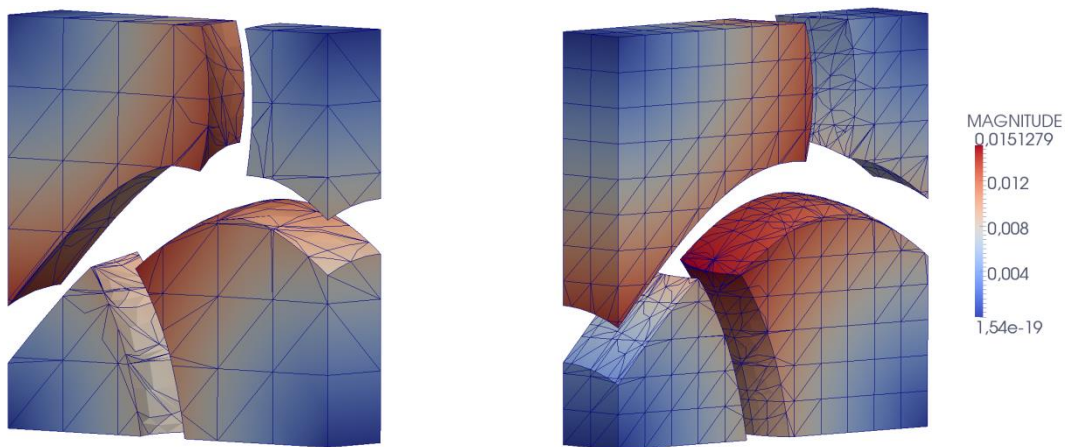
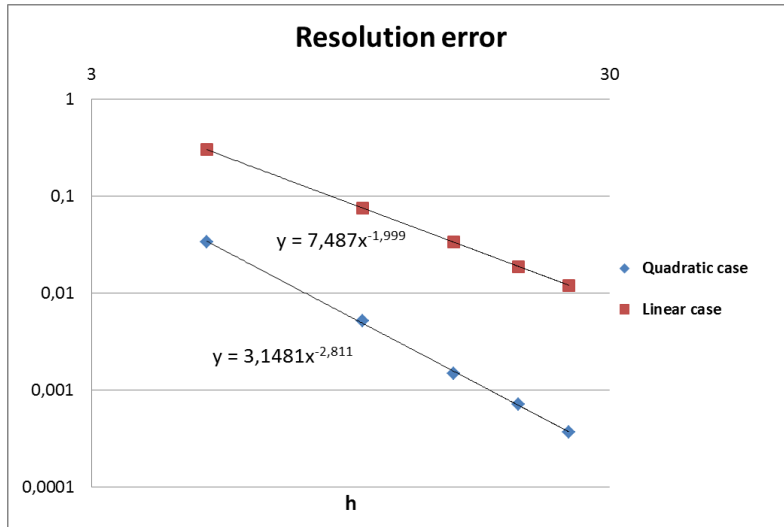


Figure 53: deformed mesh (x80) and displacement norm for $h=5$ and $h=10$.

We plot the resolution for different values of h . The convergence rates we get for the linear and the quadratic model are once again close to the expected analytical rates.



4.4 Convergence analysis for a curved approximated crack front

The next convergence analysis we perform concerns the fracture front, modeled by means of a *normal level set* and a *tangential level set*. The geometry is depicted on Figure 54. A cube of size 1 m is cut at mid height by a plane crack whose front is sinusoidal. We do not perform any mechanical simulations; we only focus on the computed surface of the crack in order to validate our integration technique for crack fronts. The crack is defined by two level set functions, a normal level set and a tangential level set:

$$\begin{cases} lsn(x, y, z) = z - 1/2 \\ lst(x, y, z) = 1 - y - 0,75 * \sin(x - 0,4) \end{cases}$$

The crack corresponds to the surface $\{l_{sn} = 0\} \cap \{l_{st} < 0\}$ and the crack front to the curve $\{l_{sn} = 0\} \cap \{l_{st} = 0\}$

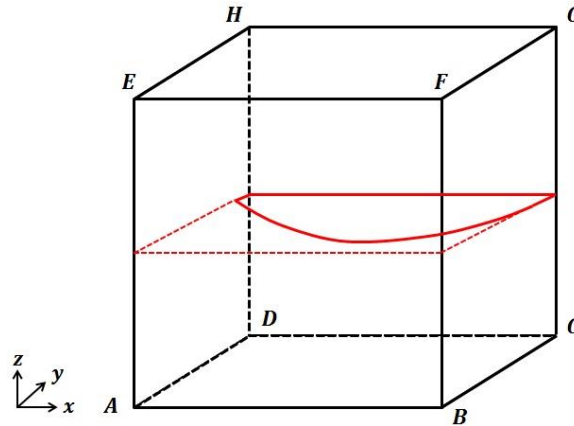
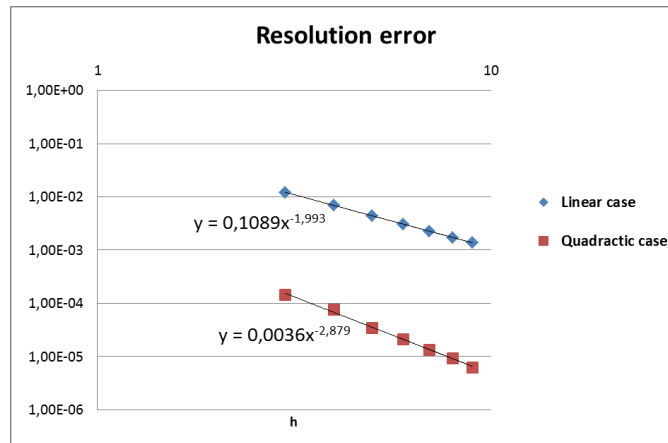


Figure 54: geometry of the problem.

We use regular meshes formed of hexaedra, the parameter h denoting the number of elements in each direction. The theoretical tangential level set we choose is a sine curve which is not perfectly solved by the linear nor the quadratic cutting procedure we use. We perform a convergence analysis on the crack front resolution ε defined previously.

The resolution for the crack front area is plotted with respect to h :



The convergence rates we get for the linear and the quadratic models are entirely satisfactory. Once again, for the quadratic case, the error level is lower than the one of the linear case. For the quadratic case, the relative error we get for $h = 3$ is close to 10^{-4} , which seems very low. This is due to an artificial refinement of the approximated crack front. The design of the contact faces in an hexahedral element that contains the crack front is summarized in Figure 55:

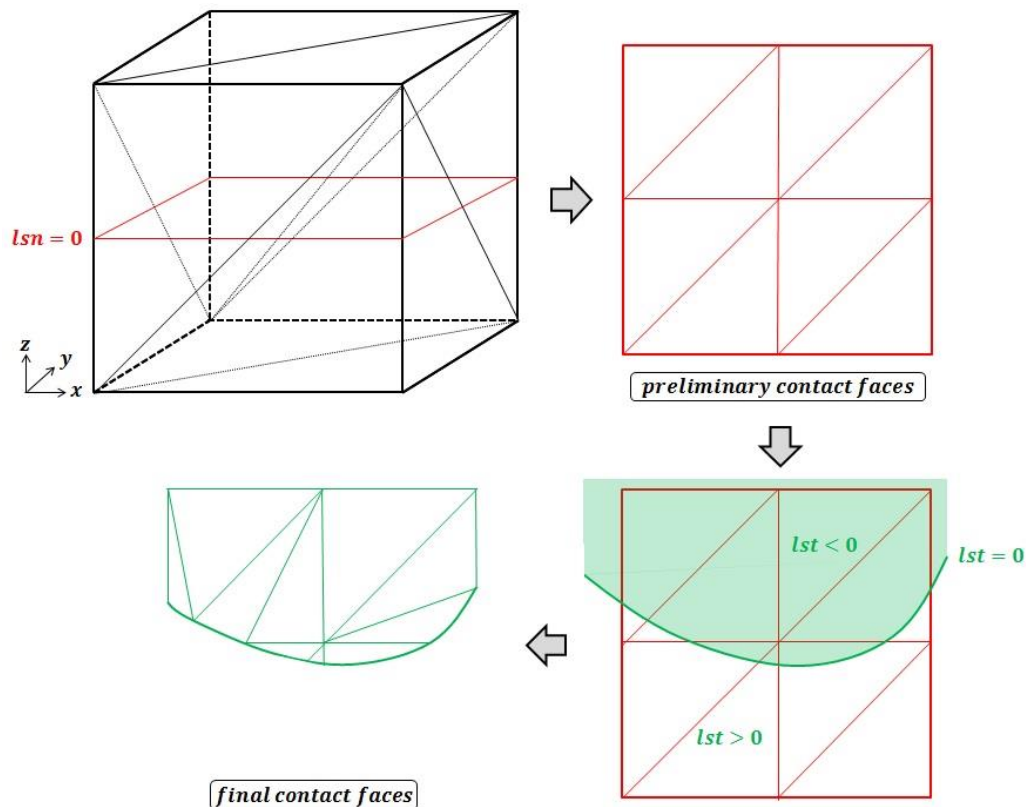


Figure 55: building of the contact faces in a hexahedral element that contains the crack front:

- 1 - the hexahedron is cut with respect to the normal level set
- 2 - we extract 8 triangular preliminary contact faces from the hexahedron
- 3 - the preliminary contact faces are cut with respect to the normal level set
- 4 - the 10 final triangular contact faces of the hexahedron

So in the case depicted in Figure 55, considering an horizontal normal level set, for one hexahedral element we get 8 preliminary contact faces, 10 final contact faces and the crack front is

approximated by 6 three nodes segments. This artificial refinement of the approximated crack front explain the very good accuracy we obtain for the geometrical resolution of the crack front.

4.5 Spherical interface on a radial mesh

The test consists in a spherical cap divided by a concentric interface. The external part of the cap is clamped and its internal part is submitted to a constant pressure p . On the interface, the theoretical contact pressure is applied mechanically on each side. Indeed, the choice has been made to exclude proper contact formulations in the following experiments, so that we can focus on the accuracy of the integration procedure, independently from the convergence of a contact algorithm. We take advantage of the symmetry of the problem and model only a quarter of hemisphere. The orthoradial displacements are then prescribed on the lateral boundaries of the domain (Figure 56). The material is linear elastic.

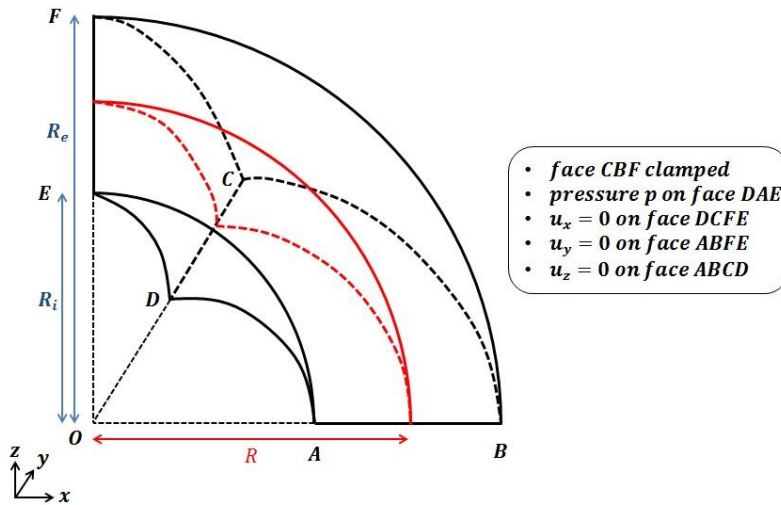


Figure 56: geometry and boundary conditions.

If the exact theoretical contact pressure is applied on the interface, the cap should behave likewise a one-piece cap. The displacements for this very classical problem only depend on the distance r to the center and are then given by:

$$\begin{cases} u_r(\mathbf{r}) = C_1 r + \frac{C_2}{r^2} \\ u_\theta(\mathbf{r}) = 0 \\ u_\varphi(\mathbf{r}) = 0 \end{cases} \quad \text{with} \quad \begin{cases} C_1 = \frac{-p}{E \left(2 \frac{R_e^3}{R_i^3} + 1 \right)} \\ C_2 = -C_1 R_e^3 \end{cases}$$

Considering the spherical coordinates system (r, θ, φ) , the stress tensor is given by:

$$\begin{cases} \sigma_{rr}(r) = E \frac{du_r(r)}{dr} \\ \sigma_{\theta\theta}(r) = \sigma_{\varphi\varphi}(r) = E \frac{u_r(r)}{r} \\ \sigma_{r\theta} = \sigma_{r\varphi} = \sigma_{\theta\varphi} = 0 \end{cases}$$

The contact pressure we must enforce on the interface is then:

$$-\mathbf{e}_r \cdot \boldsymbol{\sigma}(R) \cdot \mathbf{e}_r = -\sigma_{rr}(R) = -E \left(C_1 - 2 * \frac{C_2}{R^3} \right) = -E C_1 \left(1 + 2 * \frac{R_e^3}{R^3} \right) = p * \frac{1 + 2 * \frac{R_e^3}{R^3}}{1 + \frac{R_e^3}{R_i^3}}$$

This pressure is applied on each side of the interface.

We choose the following set of parameters:

$$R_i = 1, R_e = 2, R = 1.5, E = 5,8.10^9, \nu = 0, p = 10^6$$

The meshes we have chosen are radial (see Figure 57 left). The parameter h denotes the number of volumetric elements in the thickness of the cap. It also corresponds to the number of volumetric elements in the angular directions.

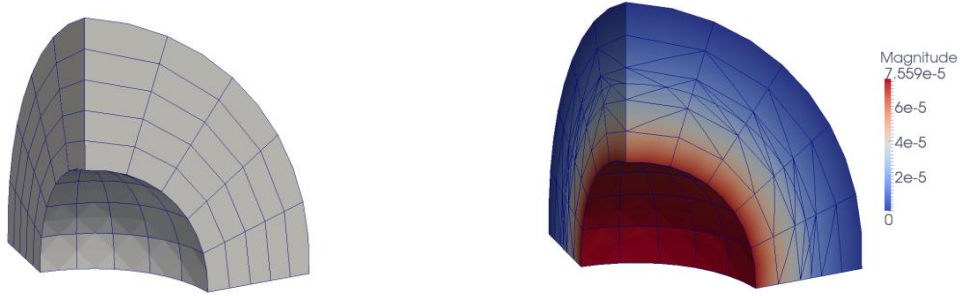


Figure 57: the quadratic mesh used for $h=5$, formed of 100 hexahedra and 25 pentahedra (left) and the obtained displacements norm (right).

On Figure 57 (right), we observe the displacements magnitude on the post-processing mesh for the quadratic case. In particular, we can see clearly the edges of the integration subcells generated around the interface.

We perform a convergence analysis for both the linear and the quadratic case on the displacement error rate and the energy error rate. The displacement relative error we compute is:

$$\text{displacement error} = \frac{\sqrt{\int_{\Omega_h} (\mathbf{u}_{h,1} - \mathbf{u}_h)^2}}{\sqrt{\int_{\Omega_h} \mathbf{u}_{h,1}^2}}$$

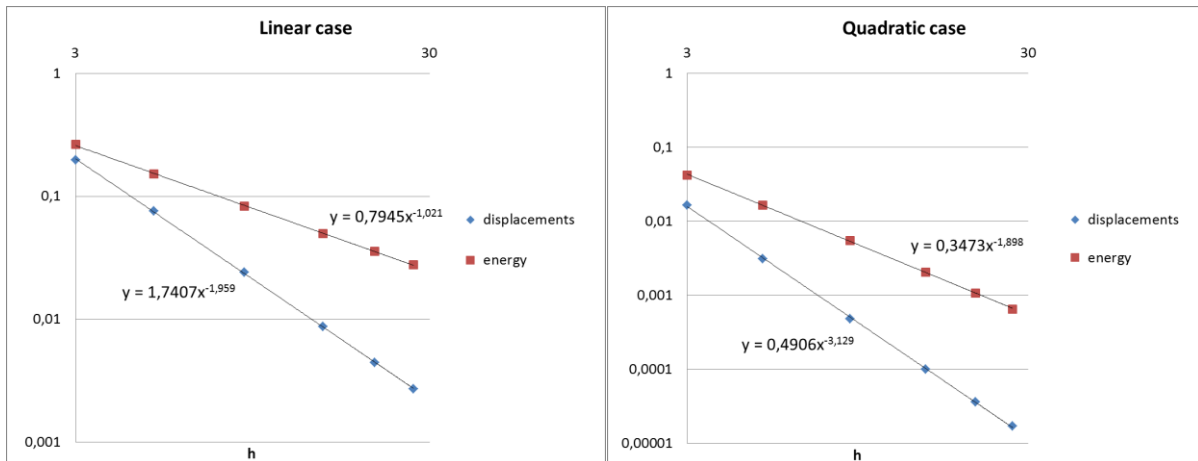
where Ω_h is the approximated domain (the domain covered by the integration mesh), $\mathbf{u}_{h,1}$ is the analytical solution projected on the displacements approximation space and \mathbf{u}_h is the displacement field we obtained. . The energy relative error we compute is:

$$\text{energy error} = \frac{\sqrt{\int_{\Omega_h} \boldsymbol{\varepsilon}(\mathbf{u}_{h,1} - \mathbf{u}_h) : \mathbf{A} : \boldsymbol{\varepsilon}(\mathbf{u}_{h,1} - \mathbf{u}_h)}}{\sqrt{\int_{\Omega_h} \boldsymbol{\varepsilon}(\mathbf{u}_{h,1}) : \mathbf{A} : \boldsymbol{\varepsilon}(\mathbf{u}_{h,1})}}$$

where $\boldsymbol{\varepsilon}$ is the Lagrangian strain tensor and \mathbf{A} is linear elastic Hooke tensor.

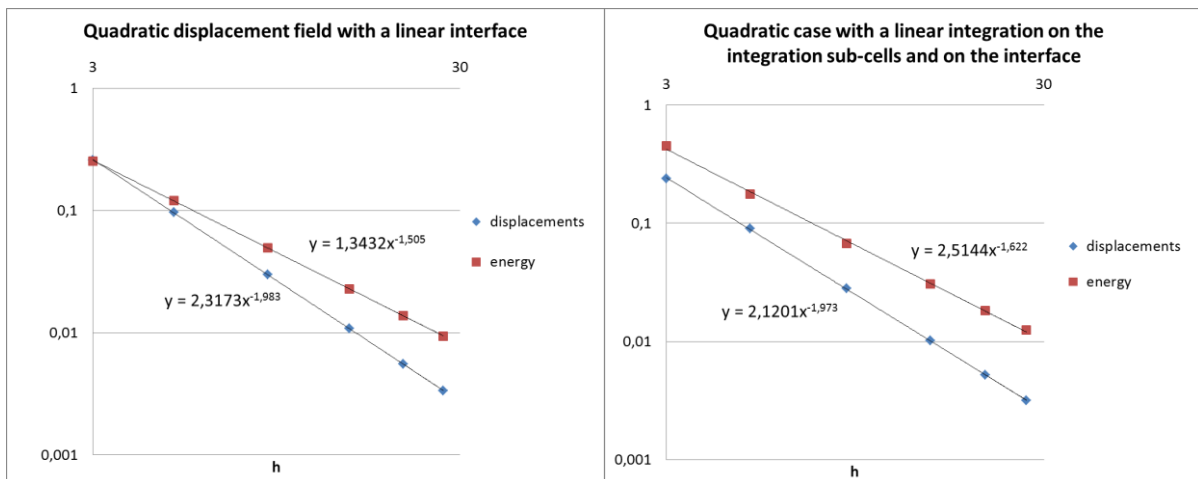
According to [Ferté1**], the convergence rates we expect for the displacement error and the energy error are respectively 2 and 3 in the quadratic case and 1 and 2 in the linear case.

Hereunder we display the energy error and the displacement error for different values of h .



The convergence rates we obtain are close to the analytical values.

In order to emphasize the importance of the quadratic approximation of the level set, we perform the same convergence analysis with a quadratic displacement field and a linear approximation of the interface. The convergence rates we get are also in agreement with the theory [Ferté1**]. The rate jumps from 3 to 2 for the relative error of the displacement norm and from 2 to 1.5 for the relative error of the energy norm. Finally, in order to confirm the relevance of the Standard Gauss integration schemes we use, we perform a last convergence analysis with a quadratic approximation space for the displacement field but Standard Gauss integration schemes corresponding to the linear case for the volumetric integration subcells and triangular contact faces. The convergence rates we get are once again deteriorated. The optimal Gauss integration schemes we had chosen in the first place are then necessary to observe the optimal theoretical convergence rates.



4.6 Spherical interface on an arbitrary mesh

This last analysis focuses on the robustness of the cutting algorithms. The problem is similar to the previous case but the mesh we use is no longer radial. For this aim we model a cube extruded from the spherical cap. The geometry of the problem is depicted on Figure 58. The size of the cube is 1 m and the radius of the spherical interface centered in A is $R = 1,2\text{ m}$. The internal radius of the spherical cap is $R_i = 0,3\text{ m}$ and its external radius is $R_e = 0,3 + \sqrt{3}\text{ m}$. The displacements are still clamped on the external boundary of the spherical cap and a mechanical pressure p is imposed on

its internal boundary. The solution is then identical to the previous case. The theoretical contact pressure is still applied on both sides of the interface.

On faces EFGH, BCGF and DCGH, we impose the theoretical displacements of this problem. On faces ABCD, ABFE and ADHE, the exact Neumann boundary conditions are imposed. Finally, on three points of the block located inside the spherical interface, the theoretical displacements are imposed in order to prevent rigid block displacements.

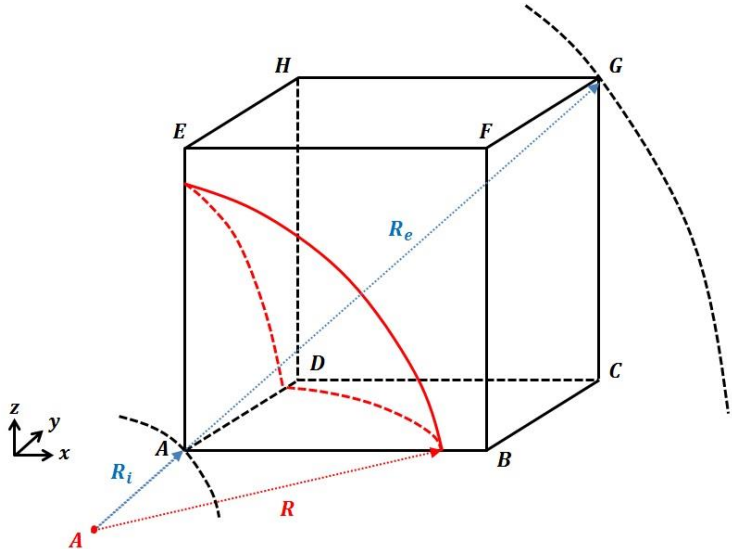


Figure 58: geometry of the problem.

We use regular meshes formed of hexaedra. Thus the elements are arbitrarily cut by the spherical interface. As a consequence, many cutting configurations are likely to happen. The test is realized for both linear and quadratic elements. h denotes here the number of elements on each edge of the cube. On Figure 59, we observe the displacements on the integration mesh for $h = 5$ in the quadratic case. In particular, we observe the edges of the volumetric integration subcells surrounding the spherical interface.

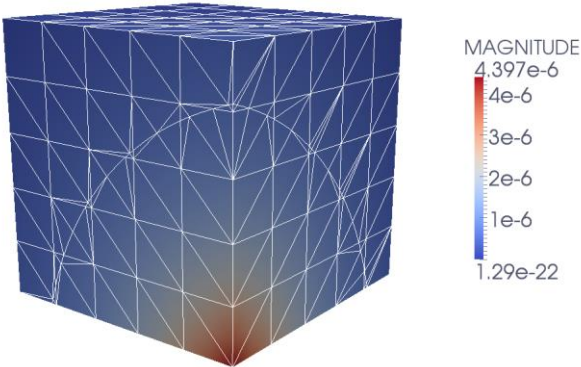
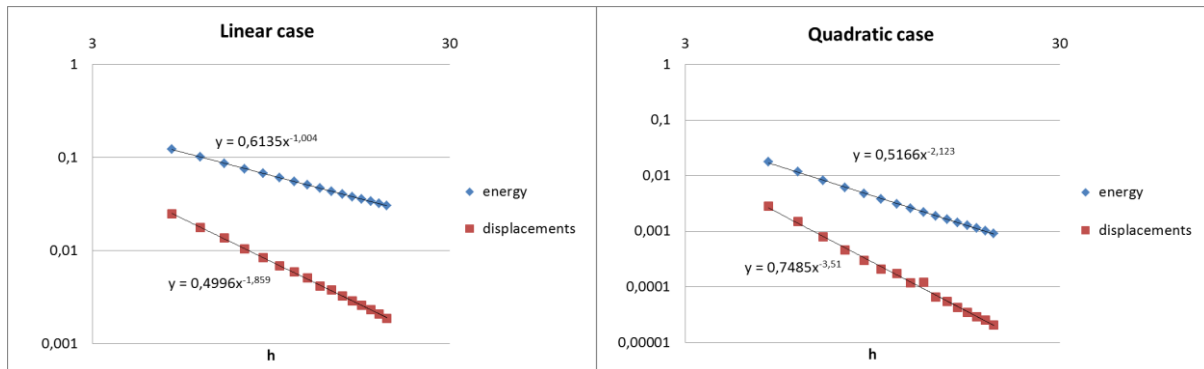


Figure 59: displacements norm on the integration mesh for $h=5$ in the quadratic case.

A convergence analysis is performed in order to prove the robustness of our integration procedure when a curved interface arbitrarily crosses a regular mesh. The relative error for the displacements

and the energy is plotted below for various values of h , in the linear case as well as in the quadratic case.



In the linear case, the order of convergence we get are in perfect agreement with the theory. In the quadratic case, we observe a superconvergence for the displacements norm compared to the theoretical order we should get. The exact same convergence rate and error level is obtained using classical quadratic finite elements on the same meshes. Once more, the convergence rates are in accordance with the theory, demonstrating the accuracy and reliability of the integration procedure. For large values of h , a great number of hexahedral elements are arbitrarily cut by a curved interface, testing the limits of our model. The conditioning is also really put to the test as the interface often passes very close to the nodes of the approximation mesh. For $h > 8$, some degrees of freedom are systematically eliminated according to the “stiffness criterion” presented in Section 4.1. These eliminations do not disrupt the convergence.

Conclusion

A robust strategy for the accurate quadratic integration of implicitly defined geometries within the framework of the eXtended Finite Element Method has been proposed. The elements of the mesh crossed by arbitrary discontinuities are split into simplex integration subcells whose faces accurately approximate the sub-domain boundaries, so that we can integrate exclusively in each sub-domain. The choice is made to offer a never failing technique: the integration strategy performs without remeshing, under the assumption that the user is aware of the relative size of the elements of the mesh compared with the pattern of the arbitrary interfaces. For this aim, we systematically reduce the problem to a few topologically distinct cutting configurations at the price of eventual shifts for the interpolated zero-level set. The design of the integration subcells is systematically performed in the associated reference element, so that overall the cutting procedure is handled by a one-dimensional root-finding algorithm. Problematic cases have to be handled with thorough consideration. For the surface integration, the contact faces are directly recovered as the faces of the integration subcells coinciding with the interpolated interfaces. Optimal convergence rates are obtained in three dimensions for both linear and quadratic models. Our integration technique is flexible with respect to the method used. It may be used in a number of methods that use implicitly defined geometries. Finally, our cutting procedure may be combined with the techniques depicted in [Mousavi**] and [Ventura**] that use a fixed sample of integration points within the elements crossed by arbitrary interfaces. The relative weights of the integration points are estimated upon the approximated sub-domains. These techniques offer excellent performances and present the advantage of reducing the number of integration points in the vicinity of the interfaces.

Annex 1: Overview of the whole cutting procedure

The overall cutting procedure, including the design of the integration subcells and the recovery of the contact faces, is summarized hereunder:

- Loop over the elements of the mesh

Step 1: Identify the elements that need to be cut

- Loop over the arbitrary discontinuities defined by means of level set functions
 - ✓ The elements that are entirely crossed by the discontinuity are labeled as “interface elements” for the current discontinuity.
 - ✓ The elements that are cut by the discontinuity and include a piece of the crack front are labeled as “crack-tip elements” for the current discontinuity.

Step 2: Cut the elements into volumetric subcells which fit the arbitrary discontinuities

The variable j is introduced in order to mark the configuration chosen to split the element into primary simplex cells. j is initialized to 1 and cannot go past j_{max} which corresponds to the number of eligible configurations for the current type of mesh.

- Loop over the discontinuities for which the current element was labeled (at this stage, we do not distinguish the “crack-tip elements” from the “interface elements”)
 - If the element is cut for the first time (first discontinuity for this element)
 - ✓ The element is split up into primary simplex cells according to configuration j
 - ✓ For each primary simplex cell, we determine the intersections with the zero-level set in the reference space of the parent element using algorithm 1 and shape the integration subcells generated by each primary simplex cell. If a double cancelation of the normal level set along the edge of a primary simplex subcell is detected and $j < j_{max}$ → **go back to the previous step with $j = j + 1$.**
 - ✓ We obtain a set of integration subcells fitting the current discontinuity.
 - ✓ The nodes of the integration subcells that coincide with the discontinuity are specifically labeled.
 - ✓ Each integration subcell is labeled with the local sign of the normal level set.
 - Else (the element has already been split into integration subcells)
 - ✓ We map each integration subcell with its associated reference element and determine the intersections with the zero-level set using algorithm 1. Then we shape the integration subcells generated by each integration subcell.
 - ✓ We obtain a set of integration subcells fitting not only the current discontinuity but also the previous ones.
 - ✓ The nodes of the integration subcells that coincide with the current discontinuity are specifically labeled.
 - ✓ Each integration subcell is labeled with the local sign of the current normal level set .

Step 3: Recover the contact faces

- Loop over the discontinuities for which the current element was labeled
 - If the element is an “interface element” for the current discontinuity
 - ✓ Determine the sign of the normal level set associated to the side of the interface chosen for the recovery of the preliminary contact faces.
 - ✓ Loop over the integration subcells labeled with the corresponding sign and determine the faces that coincide with the iso-zero of the normal level set.
 - ✓ We obtain the set of contact faces for the current discontinuity.

- Else, the element is a “crack-tip” element for the current discontinuity
 - ✓ Determine the sign of the normal level set associated to the side of the interface chosen for the recovery of the preliminary contact faces.
 - ✓ Loop over the integration subcells labeled with the corresponding sign and determine the faces that coincide with the iso-zero of the normal level set.
 - ✓ We obtain a set of preliminary contact faces for the current discontinuity.
 - ✓ Classify the preliminary contact faces into 3 groups :
 - The group of preliminary contact faces satisfying $lst > 0$. The elements of this group are out of the bounds.
 - The group of preliminary contact faces satisfying $lst < 0$. We select this entire group for the final contact faces.
 - The group of preliminary contact faces intersected by the iso-zero of the tangential level set.
 - ✓ The elements of this last group need a further cutting. We map the preliminary contact faces with their associated reference element and use algorithm 1 to cut these preliminary contact faces with respect to the iso-zero of the tangential level set. If a double cancelation of the tangential level set along the edge of a preliminary contact face is detected and $j < j_{max}$ → **go back to the beginning of step 2 and $j = j + 1$.**
 - ✓ We obtain the set of final contact faces for the current discontinuity.

References

[Ndeffo]

M. Ndeffo, P. Massin, N. Moës, A. Martin, S. Gopalakrishnan, *On the construction of approximation space to model discontinuities and cracks with linear and quadratic elements*, IJNME, submitted

[Daux]

J. Dolbow N. Sukumar C. Daux, N. Moës and T. Belytschko. *Arbitrary branched and intersecting cracks with the extended finite element method*. Int. J. Numer. Meth. Engng., Vol 48, p 1741-1760, (2000).

[Duarte]

Duarte, C. A., L. G. Reno, et A. Simone. *A high-order generalized fem for through-the-thickness branched cracks*. Int. J. Numer. Meth. Engng, Vol 72, p 325–351, (2007).

[Simone]

A. Simone, C.A. Duarte, E. Van der Giessen. *A generalized Finite Element Method for polycrystals with discontinuous gran boundaries*, Int. J. Numer. Meth. Engng, Vol 67, p1122-1145, (2006)

[Siavelis]

P. Massin M. Siavelis, M.L.E. Guiton and N. Moës. *Large sliding contact along branched discontinuities with X-FEM*. *International Journal for Numerical Methods in Engineering*, Vol 52, p 201-219, (2013).

[Ferté1]

Guilhem Ferté, Patrick Massin and Nicolas Moës. *Convergence analysis of linear or quadratic X-FEM for curved free boundaries*. *Computer Methods in Applied Mechanics and Engineering*, Vol 278, p 794-827, (2014)

[Ventura]

Ventura G, Gracie R, Belytschko T. Fast integration and weight function blending in the extended finite element method. *International Journal for Numerical Methods in Engineering* 2009; **77**:1–29.

[Mousavi]

Mousavi S.E., Sukumar N. Numerical integration of polynomials and discontinuous functions on irregular convex polygons and polyhedrons. *Computational Mechanics* 2011; **47**:535–554.

[Faivre]

M. Faivre, B. Paul, F. Golfier, R. Giot, P. Massin, D. Colombo, 2D coupled HM-XFEM modeling with cohesive zone model and applications to fluid driven fracture network, *Engineering Fracture Mechanics*, accepted (2016)

[Géniaut]

S. Géniaut, P. Massin, N. Moës, A stable 3D contact formulation using X-FEM, *European Journal of Computational Mechanics*, DOI :10.3166/remn.16.259-275 (2012)

[Pierrès]

E. Pierrès, M. C. Baietto, A. Gravouil, A two-scale extended finite element method for modeling 3D crack growth with interfacial contact, *Comput. Methods Appl. Mech. Engrg.* 199 1165–1177 (2010)

[Ferté2]

G. Ferté, P. Massin, N. Moës, 3D crack propagation with cohesive elements in the extended finite element method, *Comput. Methods Appl. Mech. Engrg.* <http://dx.doi.org/10.1016/j.cma.2015.11.018> (2015)

[Hansbo]

A. Hansbo, P. Hansbo. A finite element method for the simulation of strong and weak discontinuities in solid mechanics, *Comput. Methods Appl. Mech. Engrg.*, volume 193, 3523–3540, (2004)

[Dathe]

G. Dhatt, G. Touzot, E. Lefrançois, Une présentation de la méthode des éléments finis, Editions Lavoisier (1981)

[Béchet]

E. Béchet, H. Minnebo, N. Moës, B. Burgardt Improved implementation and robustness study of the X-FEM for stress analysis around cracks. *Int. J. Numer. Meth. Engng.* 64(8), 1033-1056 (2009)

[Fries]

Fries T.P., Omerovic S. Higher-order accurate integration of implicit geometries. *Int. J. for Numer. Meth. Engng* 2015, 10.1002/nme.5121

[Moës]

N. Moës, A. Gravouil, T. Belytschko, Non-planar 3D crack growth by the extended finite element and level sets-Part 1: Mechanical Model, *Int. J. Numer. Meth. Engng.* 53:2549-2568 (DOI: 10.1002/nme.429) (2002)

[Minnebo]

H. Minnebo, Three-dimensional integration strategies of singular functions introduced by the XFEM in the LEFM, *Int. J. Numer. Meth. Engng.* 92:1117-1138 (DOI: 10.1002/nme.4378) (2012)

[Legrain1]

G. Legrain, R. Allais, P. Cartraud, On the use of the extended finite element method with quadtree/octree meshes, *Int. J. Numer. Meth. Engng.* 86:717-743 (DOI: 10.1002/nme.3070) (2010)

[Dolbow]

J. Dolbow N. Moës and T. Belytschko. *A finite element method for crack growth without remeshing. International Journal for Numerical Methods in Engineering*, Vol 46, p 131-150, (1999).

[Melenk]

J.M. Melenk, I. Babuska, The partition of unity finite element method: basic theory and applications, *Comput. Methods Appl. Mech. Engrg.* 39 (1996) 289-314.

[Stazi]

Stazi FL, Budyn E, Chessa J, Belytschko T. An extended finite element method with higher-order elements for curved cracks. *Computational Mechanics* 2003; **31**:38–48.

[Chessa]

J. Chessa, H. Wang, T. Belytschko, On the construction of blending elements for local partition of unity enriched finite elements, *Internat. J. Numer. Methods Engrg.* 57 (2003) 1015–1038.

[Legay]

Legay A, Wang HW, Belytschko T. Strong and weak arbitrary discontinuities in spectral finite elements. *International Journal for Numerical Methods in Engineering* 2005; **64**:991–1008.

[Cheng]

Cheng KW, Fries TP. Higher-order XFEM for curved strong and weak discontinuities. *International Journal for Numerical Methods in Engineering* 2010; **82**:564–590.

[Dréau]

Dréau K, Chevaugéon N, Moës N. Studied X-FEM enrichment to handle material interfaces with higher order finite element. *Computer Methods in Applied Mechanics and Engineering* 2010; **199**:1922–1936.

[Legrain2]

Legrain G, Chevaugéon N, Dréau K. High order X-FEM and level sets for complex microstructures: uncoupling geometry and approximation. *Computer Methods in Applied Mechanics and Engineering* 2012; **241-244**:172–189.

[Moumnassi]

Moumnassi M, Belouettar S, Béchet É, Bordas SPA, Quoirin D, Potier-Ferry M. Finite element analysis on implicitly defined domains: an accurate representation based on arbitrary parametric surfaces. *Computer Methods in Applied Mechanics and Engineering* 2011; **200**:774–796.

[Osher]

S. Osher, J.A. Sethian, Fronts propagation with curvature dependent speed: algorithms based on Hamilton-Jacobi formulations, *J. Comput. Phys.* 79 (1998) 12-49

[Huerta]

Sala-Lardies E, Fernández-Méndez S, Huerta A. Optimally convergent high-order X-FEM for problems with voids and inclusion. *Proceedings of the ECCOMAS 2012*, Vienna, Austria, 2012; 1–14.

[Sethian]

JA. Sethian Level Set Methods and Fast Marching Methods (2nd edn). Cambridge University Press: Cambridge, 1999.

[Laborde]

Laborde P, Pommier J, Renard Y, Salaün M. High-order extended finite element method for cracked domains. *International Journal for Numerical Methods in Engineering* 2005; **64**:354–381.

[AFEM]

D. Ling, Q. Yang, B. Cox An augmented finite element method for modelling arbitrary discontinuities in composite materials, *Int. J. Fract.* 156 53-73 (2009)

[Chahine]

E. Chahine, P. Laborde, Y. Renard crack-tip enrichment in the XFEM method using a cutt-off function, *Internat. J. Numer. Methods Engrg.* 75 629-646 (2008)

[Chevaugéon]

N. Chevaugéon, N. Moës, H. Minnebo, Improved crack-tip enrichment functions and integration for crack modeling using the extended finite element method, *J. Multiscale Comput. Eng.* 11 597-631 (2013)

[Nicaise]

S. Nicaise, Y. Renard, E. Chahine, Optimal convergence analysis for the extended finite element method, *Internat. J. Numer. Methods. Engrg.* 84 1115-1138 (2010)

[Béchet**]

E. Béchet, H. Minnebo, N. Moës, B. Burgardt, Improved implementation and robustness study of the X-FEM for stress analysis around cracks, *Internat. J. Numer. Methods Engrg.* 64 1033-1056 (2005)