

A Generalized Finite Element Method for the Simulation of Three-Dimensional Dynamic Crack Propagation

C. A. Duarte*, O. N. Hamzeh, T. J. Liszka and W. W. Tworzydło

COMCO, Inc., 7800 Shoal Creek Blvd. Suite 290E Austin, Texas, 78757, USA

* Corresponding author, e-mail: armando@altair.com

To appear in “Computer Methods in Applied Mechanics and Engineering”.

Abstract

This paper is aimed at presenting a partition of unity method for the simulation of three-dimensional dynamic crack propagation. The method is a variation of the partition of unity finite element method and *hp*-cloud method. In the context of crack simulation, this method allows for modeling of arbitrary dynamic crack propagation without any remeshing of the domain.

In the proposed method, the approximation spaces are constructed using a Partition of Unity (PU) and local enrichment functions. The PU is provided by a combination of Shepard and finite element partitions of unity. This combination of PUs allows the inclusion of arbitrary crack geometry in a model without any modification of the initial discretization. It also avoids the problems associated with the integration of moving least squares or conventional Shepard partitions of unity used in several meshless methods. The local enrichment functions can be polynomials or customized functions. These functions can efficiently approximate the singular fields around crack fronts. The crack propagation is modeled by modifying the partition of unity along the crack surface and does not require continuous remeshings or mappings of solutions between consecutive meshes as the crack propagates.

In contrast with the boundary element method, the proposed method can be applied to any class of problems solvable by the classical finite element method. In addition, the proposed method can be implemented into most finite element data bases. Several numerical examples demonstrating the main features and computational efficiency of the proposed method for dynamic crack propagation are presented.

1 Introduction

This paper is aimed at presenting a partition of unity method tailored for three-dimensional crack simulations. The importance and difficulty of such simulations is reflected by the num-

ber of approaches that have been proposed over the past decades. Most of the techniques proposed so far are restricted to stationary cracks or to cracks propagating in two-dimensional manifolds. A survey of methods available can be found in [23, 34]. In addition, many of the techniques aimed at modeling three-dimensional crack propagation are restricted to planar crack configurations [9] or require considerable intervention of the analyst during the simulation process. Among the most versatile and promising techniques for simulation of arbitrary crack propagation in three-dimensions are: (i) The boundary element method (BEM) is a very appealing approach to solve this class of problems because it leads to a reduced dimensionality. Examples of BEMs for three-dimensional crack propagation can be found in [20, 21, 26, 47]. The main drawbacks of this approach are those inherent to the BEM. Namely, they are difficult to be extended to non-linear problems and can be quite computationally expensive; (ii) Finite element methods with local remeshing around the crack front [8, 30, 43], while versatile, are quite complex and can not be implemented in most existing finite element data structures. The continuous remeshing and projections between successive meshes are also a drawback of this approach; (iii) The element free Galerkin method [4–6, 22, 23, 39], the *hp* cloud method [15–17, 35, 37], the reproducing kernel particle method [27] are examples of the so-called meshless methods. Krysl and Belytschko [22, 23] have recently shown that the high flexibility of these methods can be exploited to model arbitrary crack propagation in three-dimensional spaces. Nonetheless the high flexibility of these methods comes at a substantial computational cost. Moreover, they can not be implemented into existing finite element data structures.

This paper presents a partition of unity (PU) method aimed at modeling crack propagation in a three-dimensional space. This method uses the same partition of unity framework used in *hp*-cloud [15, 16], partition of unity finite element (PUFEM) [2, 31] and generalized finite element method (GFEM) [12, 41]. The key difference between these methods and the method presented here is in the choice of the partition of unity. Here, the partition of unity is provided by a combination of Shepard [24, 40] and finite element partitions of unity. This partition of unity allows the inclusion of arbitrary crack geometry in a model without any modification of the initial discretization. We call this partition of unity a *finite element-Shepard partition of unity*. This choice of partition of unity also avoids the problem of integration associated with the use of moving least squares or conventional Shepard partitions of unity which are used in several meshless methods [6, 28, 33]. Although the partition of unity used in the method proposed in this paper differs from that used in the GFEM presented in [12, 41], we believe that it is appropriate to refer to method developed here as a generalized finite element method (GFEM). This is justified by the fundamental similarities of the two methods and because the method presented here can also be interpreted as a variation or generalization of the classical finite element method. Therefore we refer to the partition of unity method proposed here as the generalized finite element method (GFEM).

This paper is organized as follows. In Section 2, the formulation of generalized finite element approximations is presented. This includes the definition of the FE-Shepard partition of unity used over cracked elements, the definition of generalized finite element (GFE) shape

functions and modeling of the crack front using customized functions. Sections 3 and 4 describe the crack mechanics and physics used in the study. In Section 5, the computational engine used to represent the crack surface and the boundary of the domain and their interaction is briefly described. Several numerical examples are presented in Section 6. Finally, in Section 7, major conclusions of this study are given.

2 Formulation of Generalized Finite Element Approximations for 3D Crack Modeling

We begin this section by reviewing the concept of a partition of unity (PU).

Let Ω be an open domain in \mathbb{R}^n , $n = 1, 2, 3$ and \mathcal{T}_N an open covering of Ω consisting of N supports ω_α (often called *clouds*) with centers at \mathbf{x}_α , $\alpha = 1, \dots, N$, i.e.,

$$\mathcal{T}_N = \{\omega_\alpha\}_{\alpha=1}^N \quad \bar{\Omega} \subset \bigcup_{\alpha=1}^N \omega_\alpha,$$

where the over bar indicates closure of a set.

The basic building blocks of any partition of unity approximation are a set of functions φ_α defined on the supports ω_α , $\alpha = 1, \dots, N$, and having the following property

$$\begin{aligned} \varphi_\alpha &\in C_0^s(\omega_\alpha), \quad s \geq 0, \quad 1 \leq \alpha \leq N \\ \sum_{\alpha} \varphi_\alpha(\mathbf{x}) &= 1 \quad \forall \mathbf{x} \in \Omega \end{aligned}$$

The first property implies that the functions φ_α , $\alpha = 1, \dots, N$, are non-zero only over the supports ω_α , $\alpha = 1, \dots, N$. The functions φ_α are called a *partition of unity subordinate to the open covering* \mathcal{T}_N . Examples of partitions of unity are Lagrangian finite elements, moving least squares and Shepard functions [16, 24]. In the current work, two types of PU's are utilized: Finite element PU and a version modified for cracked elements.

2.1 Finite Element Partition of Unity

The case of a finite element partition of unity (FE PU) over non-cracked elements is briefly discussed in this section. The case of elements intersecting the crack surface is discussed in Sections 2.3 and 2.7.

In the case of a FE PU, the support (cloud) ω_γ is simply the union of the finite elements sharing a vertex node \mathbf{x}_γ (see, for example, [31, 36] and Fig. 1). The partition of unity function

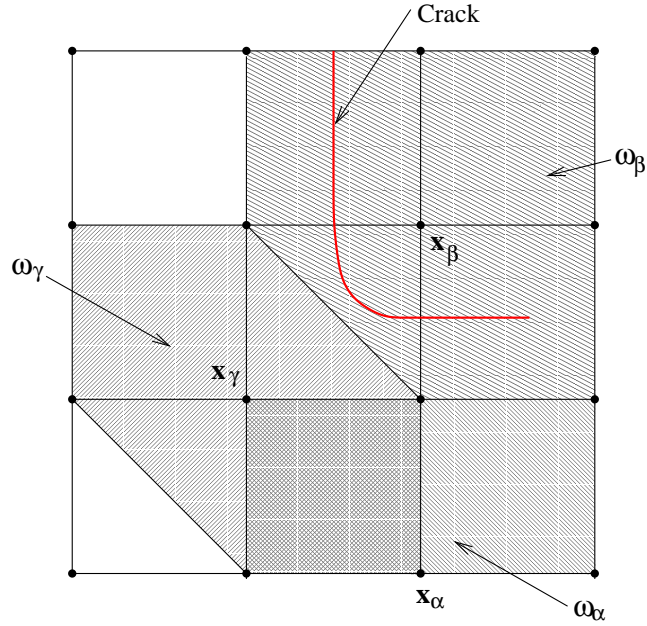


Figure 1: Clouds ω_α , ω_β and ω_γ for a finite element mesh with a crack. Polynomials of differing degree p_α , p_β and p_γ can be associated with nodes at \mathbf{x}_α , \mathbf{x}_β and \mathbf{x}_γ so as to produce non-uniform p approximations.

φ_γ is equal to the usual global finite element shape function N_γ associated with a vertex node \mathbf{x}_γ .

Let τ be a finite element with nodes \mathbf{x}_β , $\beta \in \mathcal{I}_\tau$ where \mathcal{I}_τ is an index set and let $\mathbf{x} = (x, y, z) \in \bar{\tau}$. In this work, we restrict ourselves to the case of linear Lagrangian FE PU and perform p -enrichments using the technique presented in Section 2.4. Let N_β be linear shape functions associated with nodes \mathbf{x}_β , $\beta \in \mathcal{I}_\tau$. Then, from the definition of N_β , there exist constants $a_\beta^x, a_\beta^y, a_\beta^z$, $\beta \in \mathcal{I}_\tau$, such that $\forall \mathbf{x} \in \bar{\tau}$,

$$\sum_{\beta \in \mathcal{I}_\tau} N_\beta(\mathbf{x}) = 1 \quad (1)$$

$$\sum_{\beta \in \mathcal{I}_\tau} a_\beta^x N_\beta(\mathbf{x}) = x \quad (2)$$

$$\sum_{\beta \in \mathcal{I}_\tau} a_\beta^y N_\beta(\mathbf{x}) = y \quad (3)$$

$$\sum_{\beta \in \mathcal{I}_\tau} a_\beta^z N_\beta(\mathbf{x}) = z \quad (4)$$

These basic properties of the finite element partition of unity are used in subsequent sections.

2.2 Shepard Partition of unity

The construction of a partition of unity using the so-called Shepard formula [25,40] is reviewed in this section.

Let $\mathcal{W}_\alpha : \mathbb{R}^n \rightarrow \mathbb{R}$ denote a *weighting function* with compact support ω_α that belongs to the space $C_0^s(\omega_\alpha)$, $s \geq 0$. Suppose that such weighting function is built at every cloud ω_α , $\alpha = 1, \dots, N$. Then the partition of unity functions φ_α associated with the clouds ω_α , $\alpha = 1, \dots, N$, is defined by

$$\varphi_\alpha(\mathbf{x}) = \frac{\mathcal{W}_\alpha(\mathbf{x})}{\sum_\beta \mathcal{W}_\beta(\mathbf{x})} \quad \beta \in \{\gamma \mid \mathcal{W}_\gamma(\mathbf{x}) \neq 0\} \quad (5)$$

which are known as Shepard functions [25,40].

Consider now the case in which the weighting functions \mathcal{W}_α are taken as the global linear finite element shape functions N_α associated with node \mathbf{x}_α , $\alpha = 1, \dots, N$. Let τ be a finite element with nodes \mathbf{x}_β , $\beta \in \mathcal{I}_\tau$ where \mathcal{I}_τ is an index set. The only non-zero partition of unity functions at $\mathbf{x} \in \bar{\tau}$ are given by

$$\varphi_\beta(\mathbf{x}) = \frac{N_\beta(\mathbf{x})}{\sum_{\gamma \in \mathcal{I}_\tau} N_\gamma(\mathbf{x})} = \frac{N_\beta(\mathbf{x})}{1} = N_\beta(\mathbf{x}) \quad \beta \in \mathcal{I}_\tau$$

since the finite element shape functions form a partition of unity. Therefore, it is not necessary to use the Shepard formula to build the partition of unity when the weighting functions are taken as global finite element shape functions. However, as demonstrated next, the above formula is the key to build a partition of unity when the finite element τ is severed by a crack.

2.3 Construction of a Discontinuous Partition of Unity

In this section, a technique to modify a finite element partition of unity over elements cut by a crack surface is described. The PU is modified such that a discontinuity in the displacement field across the crack surface is created. The PU is modified only for elements cut by the crack. Elsewhere, a finite element partition of unity, as described in Section 2.1, is used. The technique allows for elements to be arbitrarily cut by the crack surface without any mesh modification. Over cracked elements, the PU is built using the Shepard formula (5) and finite element shape functions as weighting functions in combination with the visibility criteria [5,6]. This partition of unity is denoted by FE-Shepard PU. The technique is first presented in a general setting followed by several illustrative examples in a two-dimensional manifold.

Let τ be a finite element with nodes \mathbf{x}_β , $\beta \in \mathcal{I}_\tau$, where \mathcal{I}_τ is an index set. Let $N_\beta, \beta \in \mathcal{I}_\tau$, denote a linear finite element shape functions for element τ . In the visibility criteria, the crack surface is considered opaque. At a given point $\mathbf{x} \in \tau$, a weighting function

N_α used in (5) is taken as non-zero if and only if the segment $[\mathbf{x} - \mathbf{x}_\alpha]$ connecting \mathbf{x} and \mathbf{x}_α does not intersect the crack surface. This criteria was originally introduced by Belytschko et al. [5,6] to model cracks in the element free Galerkin method and has since been used in several other meshless methods.

Let $\mathcal{I}_\tau^{vis}(\mathbf{x}) \subset \mathcal{I}_\tau$ denote the index set for all weighting functions that are non-zero at point $\mathbf{x} \in \tau$ according to the visibility criteria, that is

$$\mathcal{I}_\tau^{vis}(\mathbf{x}) = \{\gamma \in \mathcal{I}_\tau \mid [\mathbf{x} - \mathbf{x}_\gamma] \cap \text{crack surface} = \emptyset\} \quad (6)$$

Note that this set may be different for each point inside an element.

The FE-Shepard partition of unity for an element τ with nodes \mathbf{x}_β , $\beta \in \mathcal{I}_\tau$ is defined by

$$\varphi_\beta(\mathbf{x}) = \begin{cases} \frac{N_\beta(\mathbf{x})}{\sum_{\gamma \in \mathcal{I}_\tau^{vis}(\mathbf{x})} N_\gamma(\mathbf{x})} & \text{if } \beta \in \mathcal{I}_\tau^{vis}(\mathbf{x}) \\ 0 & \text{if } \beta \notin \mathcal{I}_\tau^{vis}(\mathbf{x}) \end{cases} \quad (7)$$

The definition above is valid for any type of finite elements and in any dimension. The discontinuity in the FE-Shepard PU is created by the fact that two points located at opposite sides of the crack surface use different sets of weighting functions to build the PU. The index set \mathcal{I}_τ^{vis} is distinct at these points although they may be geometrically very close to each other. Several illustrative examples are given below using the discretization depicted in Fig. 2.

For elements that do not intersect the crack surface, the FE-Shepard PU is provided by linear finite element shape functions. For example, at any point $\mathbf{x} \in \tau_1$ shown in Fig. 2 we have

$$\mathcal{I}_{\tau_1}^{vis}(\mathbf{x}) = \mathcal{I}_{\tau_1} = \{1, 2, 5, 6\}$$

Therefore, the PU is given by

$$\varphi_1(\mathbf{x}) = N_1, \quad \varphi_2(\mathbf{x}) = N_2, \quad \varphi_5(\mathbf{x}) = N_5, \quad \varphi_6(\mathbf{x}) = N_6$$

The FE-Shepard formula (7) can also be used to build the PU but, in this case, the results are trivial.

$$\varphi_\alpha(\mathbf{x}) = \frac{N_\alpha(\mathbf{x})}{N_1(\mathbf{x}) + N_2(\mathbf{x}) + N_5(\mathbf{x}) + N_6(\mathbf{x})} = N_\alpha(\mathbf{x}) \quad \alpha = 1, 2, 5, 6$$

Consider now the case of the element τ_2 shown in Fig. 2. At point $\mathbf{y} \in \tau_2$, according to the visibility criteria, the only non-zero weighting functions are N_6 and N_7 , since the segments $[\mathbf{y} - \mathbf{x}_{10}]$ and $[\mathbf{y} - \mathbf{x}_{11}]$ intersect the crack surface. That is

$$\mathcal{I}_{\tau_2}^{vis}(\mathbf{y}) = \{6, 7\}$$

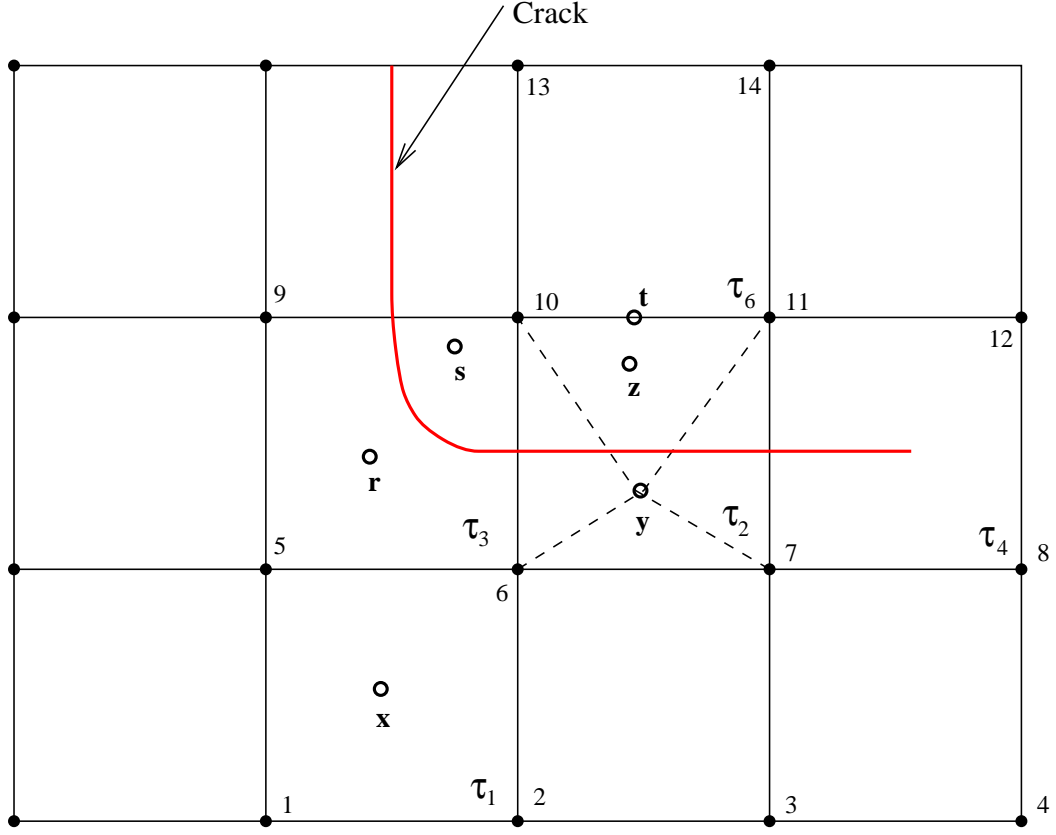


Figure 2: Example of element partition discretization cut by a crack

The partition of unity at $\mathbf{y} \in \tau_2$ is then given by

$$\varphi_6(\mathbf{y}) = \frac{N_6(\mathbf{y})}{N_6(\mathbf{y}) + N_7(\mathbf{y})} \quad \varphi_7(\mathbf{y}) = \frac{N_7(\mathbf{y})}{N_6(\mathbf{y}) + N_7(\mathbf{y})} \quad \varphi_{10}(\mathbf{y}) = 0 \quad \varphi_{11}(\mathbf{y}) = 0$$

Note that

$$\varphi_6(\mathbf{y}) + \varphi_7(\mathbf{y}) = \frac{N_6(\mathbf{y}) + N_7(\mathbf{y})}{N_6(\mathbf{y}) + N_7(\mathbf{y})} = 1$$

Therefore, the functions φ_6 and φ_7 , as defined above, constitute a partition of unity.

At point $\mathbf{z} \in \tau_2$

$$\mathcal{I}_{\tau_2}^{vis}(\mathbf{z}) = \{10, 11\}$$

and the partition of unity is given by

$$\varphi_{10}(\mathbf{z}) = \frac{N_{10}(\mathbf{z})}{N_{10}(\mathbf{z}) + N_{11}(\mathbf{z})} \quad \varphi_{11}(\mathbf{z}) = \frac{N_{11}(\mathbf{z})}{N_{10}(\mathbf{z}) + N_{11}(\mathbf{z})} \quad \varphi_6(\mathbf{z}) = 0 \quad \varphi_7(\mathbf{z}) = 0$$

Therefore, the FE-Shepard partition of unity as defined above is discontinuous across the crack surface.

As another example, consider the case of the element τ_3 with nodes $\mathbf{x}_5, \mathbf{x}_6, \mathbf{x}_9$ and \mathbf{x}_{10} as depicted in Fig. 2. At point $\mathbf{r} \in \tau_3$

$$\mathcal{I}_{\tau_3}^{vis}(\mathbf{r}) = \{5, 6, 9\}$$

and the partition of unity is given by

$$\begin{aligned} \varphi_5(\mathbf{r}) &= \frac{N_5(\mathbf{r})}{N_5(\mathbf{r}) + N_6(\mathbf{r}) + N_9(\mathbf{r})} & \varphi_6(\mathbf{r}) &= \frac{N_6(\mathbf{r})}{N_5(\mathbf{r}) + N_6(\mathbf{r}) + N_9(\mathbf{r})} \\ \varphi_9(\mathbf{r}) &= \frac{N_9(\mathbf{r})}{N_5(\mathbf{r}) + N_6(\mathbf{r}) + N_9(\mathbf{r})} & \varphi_{10}(\mathbf{r}) &= 0 \end{aligned}$$

At point $\mathbf{s} \in \tau_3$

$$\mathcal{I}_{\tau_3}^{vis}(\mathbf{s}) = \{10\}$$

and the partition of unity is given by

$$\varphi_{10}(\mathbf{s}) = \frac{N_{10}(\mathbf{s})}{N_{10}(\mathbf{s})} = 1 \quad \varphi_5(\mathbf{s}) = \varphi_6(\mathbf{s}) = \varphi_9(\mathbf{s}) = 0$$

Let us now show that the FE-Shepard PU is continuous at the boundary between cracked and non-cracked elements.

Let τ_1 and τ_2 be a cracked and a non-cracked element, respectively. Let $\mathbf{t} \in \bar{\tau}_1 \cap \bar{\tau}_2$. Suppose that the elements share a face in three-dimensions or an edge in two-dimensions. Let $\mathcal{I}_{\tau_1 \cap \tau_2}$ denote the index set of the nodes along this common face/edge. Note that

$$\mathcal{I}_{\tau_1}^{vis}(\mathbf{t}) \supset \mathcal{I}_{\tau_1 \cap \tau_2}$$

since τ_2 is not cracked (which implies that the crack does not intersect the face/edge $\bar{\tau}_1 \cap \bar{\tau}_2$).

In addition,

$$N_\alpha(\mathbf{t}) = 0 \quad \text{if } \alpha \in (\mathcal{I}_{\tau_2} - \mathcal{I}_{\tau_1 \cap \tau_2}) \quad \text{or} \quad \alpha \in (\mathcal{I}_{\tau_1}^{vis}(\mathbf{t}) - \mathcal{I}_{\tau_1 \cap \tau_2}) \quad \forall \mathbf{t} \in \bar{\tau}_1 \cap \bar{\tau}_2$$

since the only non-zero FE shape functions along a face/edge are those associated with nodes on the face/edge. Therefore, the face/edge shape functions must form a PU. Then, for any $\beta \in \mathcal{I}_{\tau_1 \cap \tau_2}$,

$$\begin{aligned} \varphi_\beta|_{\tau_1}(\mathbf{t}) &= \frac{N_\beta(\mathbf{t})}{\sum_{\gamma \in \mathcal{I}_{\tau_1}^{vis}(\mathbf{t})} N_\gamma(\mathbf{t})} = \frac{N_\beta(\mathbf{t})}{\sum_{\gamma \in \mathcal{I}_{\tau_1 \cap \tau_2}} N_\gamma(\mathbf{t})} = N_\beta(\mathbf{t}) \\ \varphi_\beta|_{\tau_2}(\mathbf{t}) &= N_\beta(\mathbf{t}) \end{aligned}$$

Consider, as an example, the point \mathbf{t} located at the boundary between elements τ_2 and τ_6 , shown in Fig. 2. In this case,

$$\mathcal{I}_{\tau_2 \cap \tau_6} = \{10, 11\} = \mathcal{I}_{\tau_2}^{vis}(\mathbf{t})$$

Consider the PU function φ_{10} associated with node \mathbf{x}_{10} . If this function is computed from element τ_2 we have

$$\varphi_{10}|_{\tau_2}(\mathbf{t}) = \frac{N_{10}(\mathbf{t})}{N_{10}(\mathbf{t}) + N_{11}(\mathbf{t})} = N_{10}(\mathbf{t})$$

If the function φ_{10} is computed from element τ_6 we have

$$\varphi_{10}|_{\tau_6}(\mathbf{t}) = \frac{N_{10}(\mathbf{t})}{N_{10}(\mathbf{t}) + N_{11}(\mathbf{t}) + N_{13}(\mathbf{t}) + N_{14}(\mathbf{t})} = N_{10}(\mathbf{t})$$

Therefore $\varphi_{10}|_{\tau_2}(\mathbf{t}) = \varphi_{10}|_{\tau_6}(\mathbf{t})$ and the function φ_{10} is continuous at \mathbf{t} .

The FE-Shepard PU defined in (7) allows arbitrary cut of the finite element mesh by the crack surface. Therefore, from the view point of modeling crack propagation, this technique enjoys all the flexibility of the so-called meshless methods. The computational cost of FE-Shepard PU over cracked elements is only marginally higher than usual finite element shape functions. For non-cracked elements this PU degenerates to the usual FE PU. In contrast, the computational cost of moving least squares functions, which are used in several meshless methods, is orders of magnitude higher than usual finite element shape functions, especially in three dimensional settings (see [17] for a comparison).

The FE-Shepard PU functions (7) are, in general, rational polynomials which are non-zero only over part of a cracked element. Therefore special care must be taken to numerically integrate these functions over cracked elements. In our current implementation, higher order Simpson rule is used for cracked elements. More efficient approaches, however, can be used. One possibility is to use an integration mesh over cracked elements that follows the crack boundaries. This mesh can easily be generated since it is used only for integration/visualization purposes and does not have to conform with neighboring elements. In the case of non-cracked elements, standard Gaussian quadrature can be used.

All the computations can be carried out at the element level as in standard finite element codes. And, importantly, the numerical integration of FE and FE-Shepard PU can be done very efficiently since the intersections of these functions coincide with the integration domains. This is in clear contrast with meshless methods based on moving least squares functions where the integration of the stiffness and mass matrices is computationally expensive.

Linear combination of FE-Shepard functions can *not*, in general, reproduce linear polynomials. That is, properties (2), (3) and (4) do not hold for the partition of unity associated with elements intersecting the crack surface. In Section 2.4, we present a technique to hierarchically add shape functions to cracked elements such that the resulting GFEM approximation can reproduce linear or higher order polynomials.

2.3.1 FE-Shepard PU for Elements at the Crack Front

Let us consider more closely the case of elements that contain the crack front. The same technique described above to build the FE-Shepard PU can be used at these elements. Consider, for example, point \mathbf{y} in element τ with nodes $\mathbf{x}_5, \mathbf{x}_6, \mathbf{x}_8$ and \mathbf{x}_9 as depicted in Fig. 3. According to (7), the PU is given by

$$\varphi_5(\mathbf{y}) = \frac{N_5(\mathbf{y})}{N_5(\mathbf{y}) + N_6(\mathbf{y})} \quad \varphi_6(\mathbf{y}) = \frac{N_6(\mathbf{y})}{N_5(\mathbf{y}) + N_6(\mathbf{y})} \quad \varphi_8(\mathbf{y}) = 0 \quad \varphi_9(\mathbf{y}) = 0$$

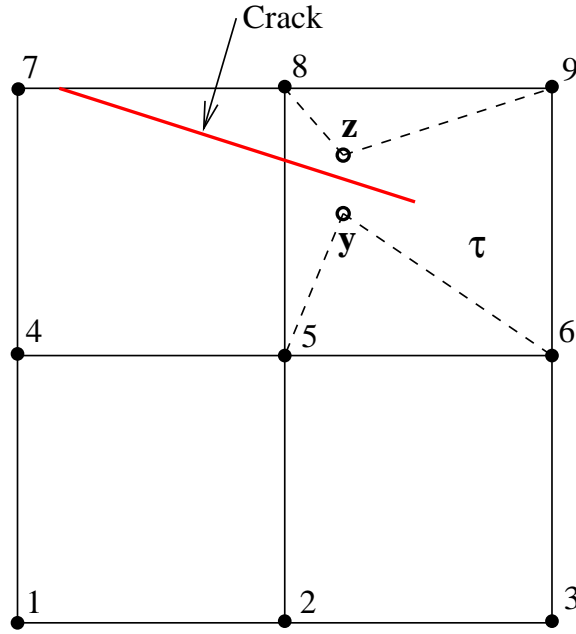


Figure 3: Construction of a FE-Shepard PU for an element at the crack front using the visibility criteria.

Consider now point $\mathbf{z} \in \tau$, at the other side of the crack surface. Here, the partition of unity is given by

$$\varphi_8(\mathbf{z}) = \frac{N_8(\mathbf{z})}{N_8(\mathbf{z}) + N_9(\mathbf{z})} \quad \varphi_9(\mathbf{z}) = \frac{N_9(\mathbf{z})}{N_8(\mathbf{z}) + N_9(\mathbf{z})} \quad \varphi_5(\mathbf{z}) = 0 \quad \varphi_6(\mathbf{z}) = 0$$

The PU is therefore discontinuous along the crack surface.

Consider now points \mathbf{r} and $\mathbf{s} \in \tau$ as depicted in Fig. 4. Equation (7) gives

$$\varphi_8(\mathbf{s}) = \frac{N_8(\mathbf{s})}{N_5(\mathbf{z}) + N_6(\mathbf{z}) + N_8(\mathbf{z}) + N_9(\mathbf{z})} = N_8(\mathbf{s}) \neq 0$$

while

$$\varphi_8(\mathbf{r}) = 0$$

since the segment $[\mathbf{r} - \mathbf{x}_8]$ intersects the crack surface. Therefore, the visibility criteria leads to spurious lines/surfaces of discontinuities inside the elements at the crack front. This problem is intrinsic to the visibility criteria and it appears in all meshless methods that use it to model crack surfaces [4, 15, 38]. In spite of this drawback, the visibility criteria is the favorite technique to model cracks in the context of meshless methods, probably because of its flexibility and relative ease of implementation in any dimension. Also, numerical experiments demonstrate that in spite of the spurious discontinuities introduced, the visibility criteria allows the computation of accurate stress intensity factors (see Section 6 and [5, 22, 23]).

Another technique to model the crack front is presented in Section 2.7. This technique is based on the wrap-around algorithm [15, 17] and the use of customized functions. In contrast to the visibility criteria, the wrap-around criteria does not introduce spurious discontinuities.

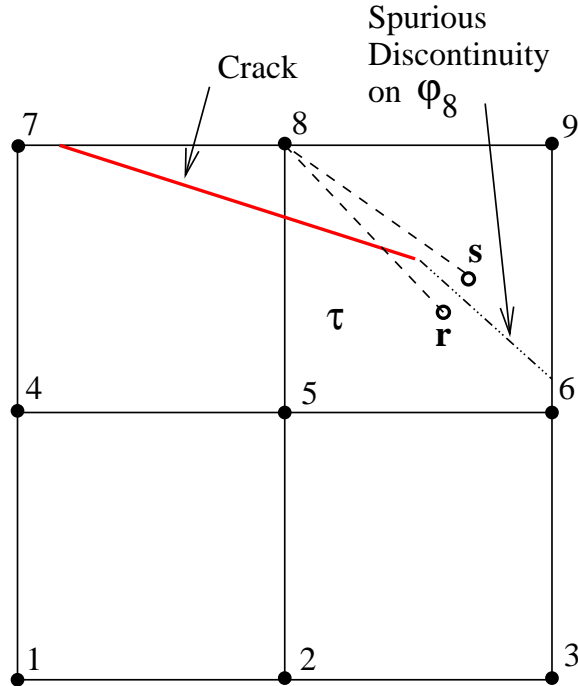


Figure 4: Spurious discontinuity on a FE-Shepard PU created by the visibility criteria.

2.4 Generalized Finite Element Shape Functions: The Family \mathcal{F}_N^p

The construction of the so-called generalized finite element or cloud shape functions is based on the following observation:

Let $\{L_i\}_{i \in \mathcal{I}}$ denote a set of functions which can approximate well, in an appropriate norm $\|\cdot\|_E$, the solution u of a boundary value problem posed on a domain Ω . Therefore, there exists $u_{hp} \in \text{span}\{L_i\}$ given by

$$u_{hp} = \sum_{i \in \mathcal{I}} u_i L_i,$$

where \mathcal{I} denotes an index set, such that

$$\|u_{hp} - u\|_E < \epsilon, \quad \epsilon \ll 1$$

Now consider the following set of so-called *cloud or generalized finite element (GFE) shape functions*, defined as

$$\phi_i^\alpha := \varphi_\alpha L_i, \quad \alpha = 1, \dots, N, \quad i \in \mathcal{I}$$

where φ_α , $\alpha = 1, \dots, N$ constitute a partition of unity (of any type) subordinate to an open covering \mathcal{T}_N of Ω . Then, it is not difficult to show that linear combinations of these shape functions can also approximate well the function u

$$\begin{aligned} \sum_{\alpha} \sum_i u_i \phi_i^\alpha &= \sum_{\alpha} \sum_i u_i \varphi_\alpha L_i = \sum_{\alpha} \varphi_\alpha \sum_i u_i L_i \\ &= \sum_{\alpha} \varphi_\alpha u_{hp} = u_{hp} \sum_{\alpha} \varphi_\alpha = u_{hp} \end{aligned} \quad (8)$$

Note that:

- (i) The shape functions ϕ_i^α , $\alpha = 1, \dots, N$, $i \in \mathcal{I}$, are non-zero only over the support of the function φ_α , i.e., the cloud ω_α .
- (ii) Linear combination of generalized finite element (GFE) shape functions can reproduce the approximation u_{hp} to the function u .
- (iii) The functions L_i , $i \in \mathcal{I}$, can be chosen with great freedom. The most straightforward choice is polynomial functions since they can approximate well smooth functions. However, for many classes of problems including the case of fracture mechanics problems, there are better choices. This case is discussed in detail in Section 2.5.

In this section, generalized finite element shape functions are defined in an n -dimensional setting using the idea outlined above.

Let the functions φ_α , $\alpha = 1, \dots, N$, denote a FE or FE-Shepard partition of unity subordinate to the open covering $\mathcal{T}_N = \{\omega_\alpha\}_{\alpha=1}^N$ of a domain $\Omega \subset \mathbb{R}^n$, $n = 1, 2, 3$. Here, N is the number of vertex nodes in the finite element mesh. The cloud ω_α is the union of the finite elements sharing the vertex node \mathbf{x}_α , regardless if the element is cracked or not (Cf. Fig. 1).

Let $\chi_\alpha(\omega_\alpha) = \text{span}\{L_{i\alpha}\}_{i \in \mathcal{I}(\alpha)}$ denote local spaces defined on ω_α , $\alpha = 1, \dots, N$, where $\mathcal{I}(\alpha), \alpha = 1, \dots, N$, are index sets and $L_{i\alpha}$ denote local approximation functions defined over the cloud ω_α . Possible choices for these functions are discussed below.

The generalized finite element (known also as cloud) shape functions of degree p are defined by

$$\mathcal{F}_N^p = \{\phi_i^\alpha = \varphi_\alpha L_{i\alpha} \mid \alpha = 1, \dots, N, i \in \mathcal{I}(\alpha)\} \quad (9)$$

Let \mathcal{I}_c denote the index set of the nodes \mathbf{x}_α that belong to cracked elements. For this set of nodes, we enforce that

$$\mathcal{P}_p(\omega_\alpha) \subset \chi_\alpha(\omega_\alpha) \quad p \geq 1 \quad \alpha \in \mathcal{I}_c$$

where \mathcal{P}_p denotes the space of polynomials of degree less than or equal to p . For all other nodes, it is only required that

$$\mathcal{P}_{p-1}(\omega_\alpha) \subset \chi_\alpha(\omega_\alpha) \quad p \geq 1 \quad \alpha \notin \mathcal{I}_c$$

The above requirements guarantee that linear combination of the GFE shape functions over any element (cracked or not) can reproduce polynomials of degree p . The proof follows the same arguments used in (8) and is presented in details in [12].

The GFE shape functions can then be used in combination with, e.g., a Galerkin method to solve any class of boundary value problem solvable by the finite element method. We call this approach the generalized finite element method (GFEM). The implementation of the method is essentially the same as in standard finite element codes, the main difference being the definition of the shape functions given in (9). The FE and FE-Shepard partitions of unity avoid the problem of integration associated with moving least squares PU built on circles or spheres. This type of PU is used in several meshless methods. Here, the integrations can be efficiently performed with the aid of the so-called master elements since the intersections of the global GFE shape functions coincide with the integration domains. Therefore, the GFEM can use existing infrastructure and algorithms for the classical finite element method. In the context of crack modeling, the GFEM allows arbitrary cut of the finite element mesh by the crack surface while being computationally efficient. The computational cost of GFE shape functions over cracked elements is only marginally higher than usual finite element shape functions. For non-cracked elements the computational cost of GFE shape functions is basically the same as finite element shape functions of the same polynomial order.

There is considerable freedom in the choice of the local spaces χ_α . The most obvious choice for a basis of χ_α is polynomial functions which can approximate well smooth functions. In this case, the GFEM over non-cracked elements is essentially identical to the classical FEM. Some important differences do exist though [11, 12]. There are many situations in which the solution of a boundary value problem is not a smooth function. In these situations, the use of polynomials to build the approximation space, as in the FEM, may be far from optimal and may lead to poor approximations of the solution u unless carefully designed meshes are used.

In the GFEM, we can use any a-priori knowledge about the solution to make better choices for the local spaces χ_α . This is the case of fracture mechanics problems. The construction of these so-called *customized* GFE shape functions over cracked elements is discussed in detail in the next section.

2.5 Customized GFE Shape Functions for a Crack in 3D

The construction of customized GFE shape functions for a crack in a three-dimensional space is summarized in this section. This is a special case of the formulation presented by Duarte et al. [12] which deals with a convex edge of arbitrary angle.

Consider a crack embedded in a three-dimensional body as depicted in Fig. 5. Both a local Cartesian coordinate system (ξ, η, ζ) and a cylindrical coordinate system (r, θ, ζ') are associated with the crack at origin (O_x, O_y, O_z) .

The displacement field $\mathbf{u}(r, \theta, \zeta')$ in the neighborhood of a straight crack front far from its ends can be written as [44, 45]

$$\mathbf{u}(r, \theta, \zeta') = \begin{Bmatrix} u_\xi(r, \theta) \\ u_\eta(r, \theta) \\ u_\zeta(r, \theta) \end{Bmatrix} = \sum_{j=1}^{\infty} \left[A_j^{(1)} \begin{Bmatrix} u_{\xi j}^{(1)} \\ u_{\eta j}^{(1)} \\ 0 \end{Bmatrix} + A_j^{(2)} \begin{Bmatrix} u_{\xi j}^{(2)} \\ u_{\eta j}^{(2)} \\ 0 \end{Bmatrix} + A_j^{(3)} \begin{Bmatrix} 0 \\ 0 \\ u_{\zeta j}^{(3)} \end{Bmatrix} \right] \quad (10)$$

where (r, θ, ζ') are the cylindrical coordinates relative to the system shown in Fig. 5, $u_\xi(r, \theta)$, $u_\eta(r, \theta)$ and $u_\zeta(r, \theta)$ are Cartesian components of \mathbf{u} in the ξ -, η - and ζ - directions, respectively.

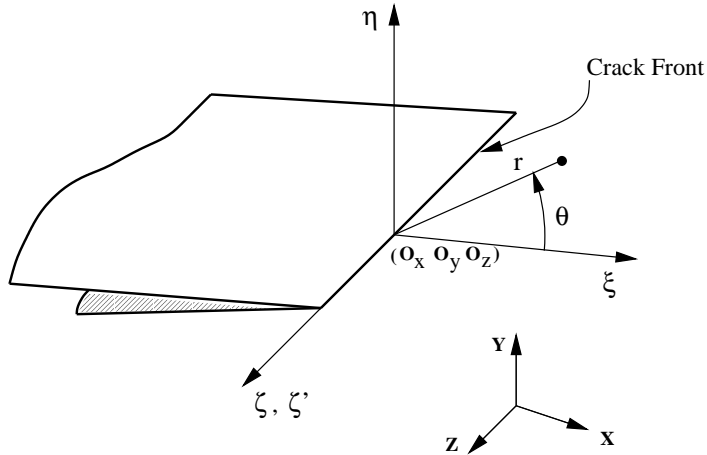


Figure 5: Coordinate systems associated with an edge in 3D space.

Assuming that the crack boundary is traction-free and neglecting body forces, the

functions $u_{\xi_j}^{(1)}$, $u_{\eta_j}^{(1)}$, $u_{\xi_j}^{(2)}$, $u_{\eta_j}^{(2)}$ are given by [44, 45]

$$\begin{aligned} u_{\xi_j}^{(1)}(r, \theta) &= \frac{r^{\lambda_j}}{2G} \{ [\kappa - Q_j^{(1)}(\lambda_j + 1)] \cos \lambda_j \theta - \lambda_j \cos(\lambda_j - 2)\theta \} \\ u_{\xi_j}^{(2)}(r, \theta) &= \frac{r^{\lambda_j}}{2G} \{ [\kappa - Q_j^{(2)}(\lambda_j + 1)] \sin \lambda_j \theta - \lambda_j \sin(\lambda_j - 2)\theta \} \\ u_{\eta_j}^{(1)}(r, \theta) &= \frac{r^{\lambda_j}}{2G} \{ [\kappa + Q_j^{(1)}(\lambda_j + 1)] \sin \lambda_j \theta + \lambda_j \sin(\lambda_j - 2)\theta \} \\ u_{\eta_j}^{(2)}(r, \theta) &= -\frac{r^{\lambda_j}}{2G} \{ [\kappa + Q_j^{(2)}(\lambda_j + 1)] \cos \lambda_j \theta + \lambda_j \cos(\lambda_j - 2)\theta \} \end{aligned}$$

where the eigenvalues λ_j are $\lambda_1 = 1/2$, $\lambda_j = (j+1)/2$, $j \geq 2$. The material constant $\kappa = 3 - 4\nu$ and $G = E/2(1 + \nu)$, where E is Young's modulus and ν is Poisson's ratio.

The parameters $Q_j^{(1)}$ and $Q_j^{(2)}$ are given by

$$Q_j^{(1)} = \begin{cases} -1 & j = 3, 5, 7, \dots \\ -\Lambda_j & j = 1, 2, 4, 6, \dots \end{cases} \quad Q_j^{(2)} = \begin{cases} -1 & j = 1, 2, 4, 6, \dots \\ -\Lambda_j & j = 3, 5, 7, \dots \end{cases}$$

where $\Lambda_j = (\lambda_j - 1)/(\lambda_j + 1)$.

Assuming that the crack boundary is traction-free, body forces are negligible, and crack front is straight, the functions $u_{\zeta_j}^{(3)}$ are given by [44]

$$u_{\zeta_j}^{(3)} = \begin{cases} \frac{r^{\lambda_j^{(3)}}}{2G} \sin \lambda_j^{(3)} \theta & j = 1, 3, 5, \dots \\ \frac{r^{\lambda_j^{(3)}}}{2G} \cos \lambda_j^{(3)} \theta & j = 2, 4, 6, \dots \end{cases}$$

where $\lambda_j^{(3)} = j/2$, $j \geq 1$.

Prior to employing the above functions to build customized generalized finite element shape functions, they first have to be transformed to the physical coordinates $\mathbf{x} = (x, y, z)$. The transformation method is described in [12].

Let

$$u_{x1}^{(1)}, u_{y1}^{(1)}, u_{z1}^{(3)}, u_{x1}^{(2)}, u_{y1}^{(2)}, u_{z2}^{(3)} \quad (11)$$

denote the result of such transformation applied to $u_{\xi_1}^{(1)}$, $u_{\eta_1}^{(1)}$, $u_{\zeta_1}^{(3)}$, $u_{\xi_1}^{(2)}$, $u_{\eta_1}^{(2)}$, $u_{\zeta_2}^{(3)}$, respectively.

The construction of customized GFE shape functions using singular functions then follows the same approach as in the case of polynomial type shape functions. Here, the singular functions from (11) take the role of the basis functions $L_{i\alpha}$ defined in Section 2.4 and are multiplied by the partition of unity functions φ_α associated with nodes near a crack front. The customized GFE shape functions used in the computations of Section 6 are built as

$$\varphi_\alpha \times \{u_{x1}^{(1)}, u_{y1}^{(1)}, u_{z1}^{(3)}, u_{x1}^{(2)}, u_{y1}^{(2)}, u_{z2}^{(3)}\} \quad (12)$$

Here, α is the index of a finite element vertex node near a crack front in 3D. Note that not necessarily the same set of singular functions is used at all enriched nodes. As discussed in Section 5, the crack front is modeled as a piecewise linear object. Therefore the orientation of the local coordinate systems used to build the customized functions (see Fig. 5) changes along the crack front.

The customized functions $u_{xj}^{(1)}$, $u_{yj}^{(1)}$, $u_{zj}^{(3)}$, $u_{xj}^{(2)}$, $u_{yj}^{(2)}$ defined above are presented here as a simple illustrative example and, as such, they are limited to the special case of piecewise linear crack fronts. It also assumes that the crack surface near the crack front is flat. More general types of customized functions, built analytically as above or perhaps numerically, can be used without any change to the definition of the customized GFE shape functions. The GFEM does not require the availability of customized functions to be able to model a crack. As shown in Section 2.3, the crack can be modeled through proper construction of the partition of unity. Nonetheless, if customized functions are available, they can considerably improve the accuracy of the method and, as shown in the next section, they can also be used to model the crack front without the spurious discontinuities created by the visibility criteria.

The enrichment of the elements near the crack front with singular functions brings up the issue of numerical integration. In this investigation, our main goal is to analyze the effectiveness of this type of enrichment and, for simplicity, we use a high order quadrature rule in the elements with singular functions. We adopt the Simpson rule with 10 points in each direction for the case of hexahedral elements. Integration of the singular functions can, of course, be implemented in a much more efficient way. Adaptive integration schemes, such as the one proposed in [42], can be used.

2.6 Examples of GFE Shape Functions

In this section, GFE shape functions for cracked and non-cracked elements are presented using the definition (9). The issue of linear dependence of these functions and how to solve the resulting system of equations is discussed in [12, 41].

Let $\tau \subset \mathbb{R}^3$ be a finite element with nodes \mathbf{x}_β , $\beta \in \mathcal{I}_\tau$ where \mathcal{I}_τ is an index set. The case of elements in one- or two-dimensional spaces is analogous.

2.6.1 Quadratic GFE shape functions for a non-cracked element

Quadratic GFE shape functions for a non-cracked element τ are given by

$$S_\tau^{p=2} := \varphi_\beta \times \left\{ 1, \frac{x - x_\beta}{h_\beta}, \frac{y - y_\beta}{h_\beta}, \frac{z - z_\beta}{h_\beta} \right\} \quad \beta \in \mathcal{I}_\tau \quad (13)$$

where φ_β is a standard linear Lagrangian FE PU, $\mathbf{x}_\beta = (x_\beta, y_\beta, z_\beta)$ are the coordinates of node β and h_β is the diameter of the largest finite element sharing the node β . Details are

described in [12]. It can be shown that the shape functions defined above are complete of degree two [12].

2.6.2 Linear GFE shape functions for a cracked element

Here, we consider the case in which an element τ_c with nodes $\mathbf{x}_\beta \in \mathcal{I}_{\tau_c}$ is fully or partially severed by the crack surface. The partition of unity over this element is given by the FE-Shepard formula (7). Linear combination of these functions can *not*, in general, reproduce linear polynomials although in many cases, depending on how the crack surface is located within the element, *some* linear monomials can still be reproduced. For simplicity, and taking into account that the number of cracked elements is much smaller than the total number of elements, we prefer to assume that the PU over cracked elements can reproduce only a constant. Therefore, the PU has to be enriched with linear polynomials in order to guarantee that the GFE shape functions over cracked elements are complete of degree one.

$$S_{\tau_c}^{p=1} := \varphi_\beta \times \left\{ 1, \frac{x - x_\beta}{h_\beta}, \frac{y - y_\beta}{h_\beta}, \frac{z - z_\beta}{h_\beta} \right\} \quad \beta \in \mathcal{I}_{\tau_c} \quad (14)$$

The only difference between (13) and the above is in the definition of the partition of unity φ_β .

2.6.3 Linear GFE shape functions for a cracked element enriched with customized shape functions

Customized GFE shape functions as those defined in (12) can be added to the shape functions of elements containing the crack front. A linear cracked element enriched with functions (12) has the following shape functions

$$S_{\tau_c}^{p=1,*} := \varphi_\beta \times \left\{ 1, \frac{x - x_\beta}{h_\beta}, \frac{y - y_\beta}{h_\beta}, \frac{z - z_\beta}{h_\beta}, u_{x1}^{(1)}, u_{y1}^{(1)}, u_{z1}^{(3)}, u_{x1}^{(2)}, u_{y1}^{(2)}, u_{z2}^{(3)} \right\} \quad \beta \in \mathcal{I}_\tau \quad (15)$$

2.7 Crack Front Modeling Using Wrap-Around Approach and Customized Functions

In this section, another technique to model the crack at elements intersecting the crack front is presented. The technique is based on the wrap-around algorithm [15, 17] and the use of customized functions like those defined in Section 2.5. In contrast with the technique based on the visibility criteria, the wrap-around criteria does not introduce spurious discontinuities.

From the definition of the GFE shape functions given in (9) it is observed that if the functions $L_{i\alpha}$ are discontinuous, the resulting GFE shape functions are also discontinuous.

The crack can therefore be modeled by simply multiplying a standard FE PU (or any other PU) by appropriate customized functions that can accurately represent the displacement field near the crack (not only at the crack front). This approach has been successfully used by Duarte and Oden [17,35,37] in two dimensions. It is also essentially the same technique used in [42] to model holes and inclusions in elastic plates. More recently, Belytschko et al. [3,10,32] have applied this approach to propagating cracks in two dimensions. They have also proposed several variations for the functions $L_{i\alpha}$ including discontinuous step functions and near crack tip asymptotic fields. Suppose now that such customized functions are available at least near the crack front. Then, customized GFE shape functions can be used for elements near the crack front and the definition of the FE-Shepard PU given in (7) can be modified such that no spurious discontinuity is created.

The FE-Shepard PU is defined as follows in the case of wrap-around (WA) algorithm. First, all nodes belonging to elements that intersect the crack front are marked as *wrap-around nodes*. Then, instead of the index set defined in (6), the following is used at a point \mathbf{x} belonging to finite element τ

$$\mathcal{I}_\tau^{wa}(\mathbf{x}) = \{\gamma \in \mathcal{I}_\tau \mid [\mathbf{x} - \mathbf{x}_\gamma] \cap \text{crack surface} = \emptyset \text{ or } \mathbf{x}_\gamma \text{ is a wrap-around node}\} \quad (16)$$

The FE-Shepard partition of unity for an element τ with nodes \mathbf{x}_β , $\beta \in \mathcal{I}_\tau$ is then defined as

$$\varphi_\beta(\mathbf{x}) = \begin{cases} \frac{N_\beta(\mathbf{x})}{\sum_{\gamma \in \mathcal{I}_\tau^{wa}(\mathbf{x})} N_\gamma(\mathbf{x})} & \text{if } \beta \in \mathcal{I}_\tau^{wa}(\mathbf{x}) \\ 0 & \text{if } \beta \notin \mathcal{I}_\tau^{wa}(\mathbf{x}) \end{cases} \quad (17)$$

For elements at the crack front it is as if the crack does not exist since all nodes of the element are marked as WA nodes then,

$$\mathcal{I}_\tau^{wa}(\mathbf{x}) = \mathcal{I}_\tau \quad \forall \mathbf{x} \in \tau$$

In this case, the crack front is modeled by customized shape functions as those defined in (12). In addition to rendering a discontinuous field at the crack front, these GFE shape functions allow accurate approximation of the solution without any mesh modifications. For elements that have a neighboring element at the crack front, the crack is modeled by a combination of the visibility algorithm and the customized shape functions. But in this case case, no spurious discontinuities are created. In the case of other cracked elements, the crack is modeled solely by the visibility algorithm. Here again, no spurious discontinuities are created.

As an example consider the element τ depicted in Fig. 6. At point $\mathbf{r} \in \tau$, we have

$$\mathcal{I}_\tau^{wa}(\mathbf{r}) = \mathcal{I}_\tau = \{5, 6, 8, 9\}$$

In the same figure, at point $\mathbf{t} \in \bar{\tau}$ we have

$$\mathcal{I}_{\bar{\tau}}^{wa}(\mathbf{t}) = \mathcal{I}_{\bar{\tau}} = \{4, 5, 7, 8\}$$

While for point $\mathbf{z} \in \bar{\tau}$,

$$\mathcal{I}_{\bar{\tau}}^{wa}(\mathbf{z}) = \{5, 8\}$$

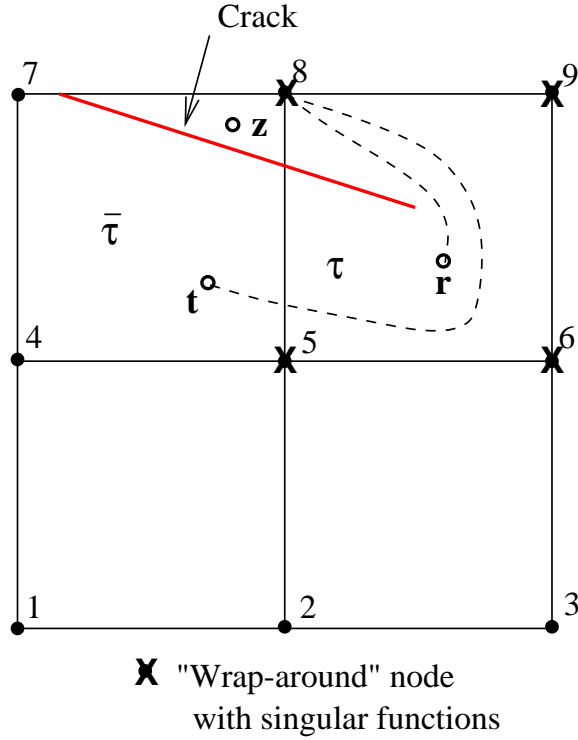


Figure 6: Modeling the crack front using the wrap-around algorithm.

3 Extraction of Stress Intensity Factors: The Least Squares Fit Method

Once the solution is obtained at some time step, the amount and direction of crack propagation over the next time increment can be predicted. The crack front is represented as a series of straight line segments connected at vertices. The stress intensity factors (SIF's) are calculated at the vertex points along the crack front. Figure 7 represents flowchart of the computer program PHLEXcrackTM used for this study with the fracture dynamics process implemented in it.

The Least Squares Fit method is used for the calculations of SIF's. In this method, the SIF's are obtained by minimizing the errors among the discretized stresses calculated from the solution and their asymptotic values. The method has produced accurate results in finite element settings. In this work, it has been extended to be used with the three-dimensional dynamic GFE model used.

Define the least-squares functional as:

$$J(K_M^l(\mathbf{y})) := (\boldsymbol{\sigma}^h - \boldsymbol{\sigma}, \boldsymbol{\sigma}^h - \boldsymbol{\sigma})_{\mathbf{y}} \quad M = I - III \quad \text{and} \quad l = 1, \dots, l_{max}$$

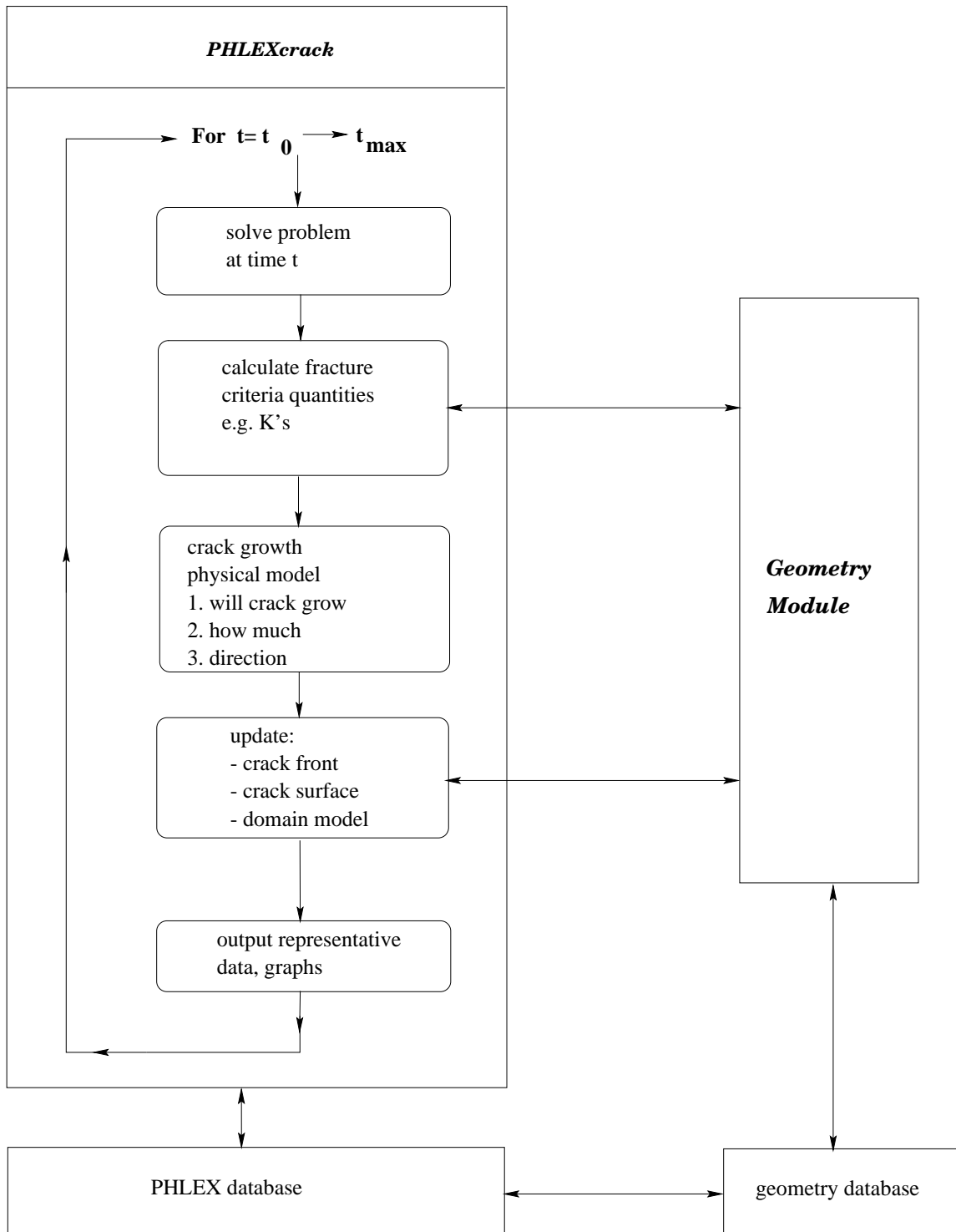


Figure 7: Flowchart of PHLEXcrackTM.

where K_M^l is the l th stress intensity factor associated with mode M at vertex \mathbf{y} . The inner product $(\cdot, \cdot)_{\mathbf{y}}$ is defined as:

$$(\mathbf{u}, \mathbf{v})_{\mathbf{y}} = \sum_{a=1}^{N_{spl}} \left(\sum_{j=1}^m \sum_{i=1}^m u_i(\mathbf{x}_a) D_{ij}^{-1} v_j(\mathbf{x}_a) \right) \mathcal{W}_a(\mathbf{y}),$$

and (see Figure 8)

$\mathbf{u} = \{u_1, u_2, \dots, u_m\}$, $\mathbf{v} = \{v_1, v_2, \dots, v_m\}$ are any two vectors in \mathbb{R}^m ,

$\boldsymbol{\sigma}^h(\mathbf{x})$ is the discretized stress vector at point \mathbf{x} ,

$\boldsymbol{\sigma}(\mathbf{x}) := \sum_{M=I}^{III} \sum_{l=1}^{l_{max}} [K_M^l \mathbf{F}_M^l(\mathbf{x})]$ is the asymptotic stress vector,

$\mathbf{F}_M^l(\mathbf{x}) = \mathbf{f}_M^l(r) \mathbf{g}_M^l(\theta)$ are the asymptotic functions,

\mathbf{y} is the position vector of a vertex on the crack front,

\mathbf{x}_a is the position vector of a sampling point,

N_{spl} is the number of sampling points in a domain centered at \mathbf{y} ,

$\mathbf{D}^{-1} \in \mathbb{R}^m \times \mathbb{R}^m$ is the inverse of an auxiliary matrix \mathbf{D} . Appropriate choices for \mathbf{D} are the material stiffness matrix or the identity matrix, and

$\mathcal{W}_a(\mathbf{y}) \in \mathbb{R}(\Omega)$ is a weighting function associated with sampling point \mathbf{x}_a and is given by

$$\mathcal{W}_a(\mathbf{y}) = \frac{1}{\|\mathbf{y} - \mathbf{x}_a\|_{\mathbb{R}^m}^p} \quad \text{where } p \text{ is typically 3-6.}$$

Stress intensity factors K_M^l are found by minimizing the least-squares functional:

$$\frac{\partial J}{\partial K_M^l} = 0 \quad M = I - III \quad l = 1, 1, \dots, l_{max}$$

This leads to the following system of equations:

$$\sum_{M=I}^{III} \sum_{q=1}^{q_{max}} K_M^q (\mathbf{F}_M^q, \mathbf{F}_{M'}^l)_{\mathbf{y}} = (\boldsymbol{\sigma}^h, \mathbf{F}_{M'}^l)_{\mathbf{y}} \quad M' = I - III \quad l = 1, \dots, l_{max}$$

If only the first terms ($l = 1$) of the three modes are used, the three corresponding K 's are found by solving:

$$\begin{bmatrix} (\mathbf{F}_I^1, \mathbf{F}_I^1)_{\mathbf{y}} & (\mathbf{F}_I^1, \mathbf{F}_{II}^1)_{\mathbf{y}} & (\mathbf{F}_I^1, \mathbf{F}_{III}^1)_{\mathbf{y}} \\ (\mathbf{F}_{II}^1, \mathbf{F}_I^1)_{\mathbf{y}} & (\mathbf{F}_{II}^1, \mathbf{F}_{II}^1)_{\mathbf{y}} & (\mathbf{F}_{II}^1, \mathbf{F}_{III}^1)_{\mathbf{y}} \\ (\mathbf{F}_{III}^1, \mathbf{F}_I^1)_{\mathbf{y}} & (\mathbf{F}_{III}^1, \mathbf{F}_{II}^1)_{\mathbf{y}} & (\mathbf{F}_{III}^1, \mathbf{F}_{III}^1)_{\mathbf{y}} \end{bmatrix} \begin{Bmatrix} K_I^1 \\ K_{II}^1 \\ K_{III}^1 \end{Bmatrix} = \begin{Bmatrix} (\boldsymbol{\sigma}^h, \mathbf{F}_I^1)_{\mathbf{y}} \\ (\boldsymbol{\sigma}^h, \mathbf{F}_{II}^1)_{\mathbf{y}} \\ (\boldsymbol{\sigma}^h, \mathbf{F}_{III}^1)_{\mathbf{y}} \end{Bmatrix}$$

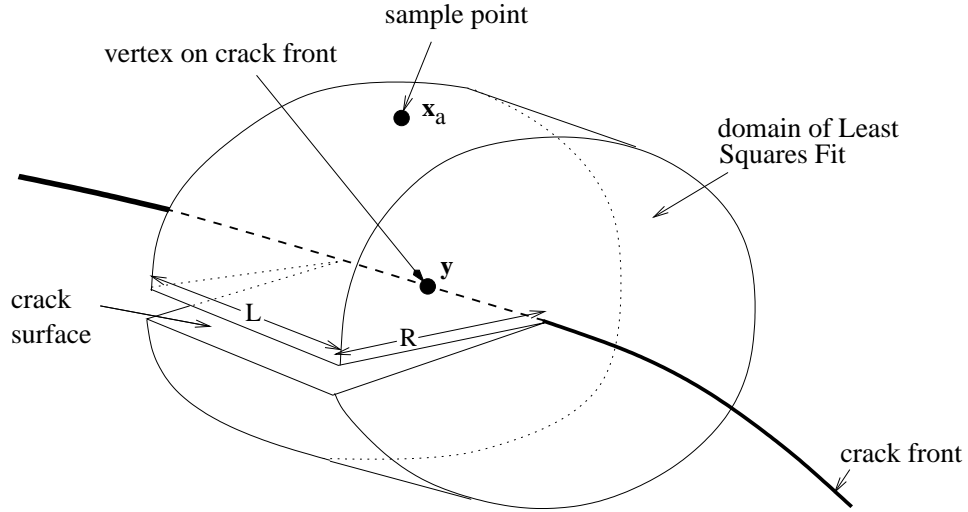


Figure 8: Domain used for the Least Squares Fit method.

The domain used for the Least Squares Fit method is a cylinder centered at the vertex with its axis along the tangent of the crack front at that vertex. The dimensions of the cylinder and the number of sampling points in the r , θ , and z directions are input data. In addition, one can choose the type of the \mathbf{D} matrix and the weight p .

4 Crack Evolution Models

The Stress Intensity Factors (SIF's) calculated above are used to determine whether the crack will advance or not, and the amount and direction of propagation, if any. The front is then advanced to its new position, the crack surface is extended, and the numerical model is updated accordingly.

Crack propagation quantities are calculated based on some physical models. Crack physics, however, are not well known, especially so for three dimensional problems. Therefore, fracture models usually make extensive use of plane strain physics models. In this work, two physical models have been used.

4.1 The Freund Model [18, 22, 39]

In this model, direction of crack growth in the plane normal to the crack front is given by:

$$\theta = 2 \tan^{-1} \left\{ \frac{1}{4} \left(\frac{K_I}{K_{II}} - \text{sign}(K_{II}) \sqrt{\left(\frac{K_I}{K_{II}} \right)^2 + 8} \right) \right\} \quad (18)$$

for $K_{II} \neq 0$, and $\theta = 0$ for $K_{II} = 0$. In the equation above, θ is measured with respect to the forward vector \mathbf{n}_1 . Vector \mathbf{n}_1 is the crack front forward normal vector; it lies along the intersection of the planes normal and tangent to the front at the vertex (see Figure 9).

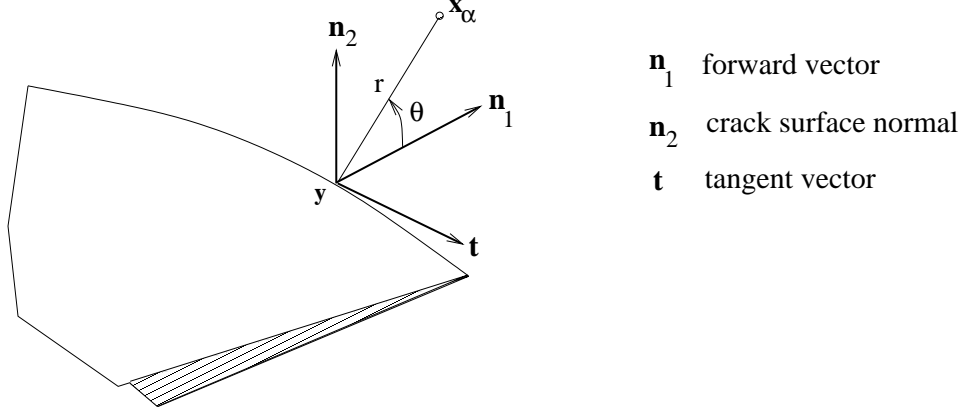


Figure 9: Local unit vectors at crack vertex.

It was assumed that mode III does not affect crack direction; it only affects crack speed. The term $sign(K_{II})$ in equation (18) above is to guarantee a positive stress intensity factor along the direction given by θ . It also corresponds to the direction normal to the maximum hoop stress.

The current energy release rate of a stationary crack is calculated at every vertex as:

$$G(0) = \frac{K_{I,equiv}^2}{E^*} + \frac{K_{III}^2}{2\mu},$$

where

$$K_{I,equiv} = K_I \cos^3(\theta/2) - \frac{3}{2}K_{II} \cos(\theta/2) \sin \theta,$$

μ is the shear modulus, and E^* is the effective Young's modulus. For plane strain, E^* is given by:

$$E^* = \frac{E}{1 - \nu^2},$$

where E is Young's modulus and ν is Poisson's ratio. Crack will propagate at a vertex if:

$$G(0) > G_{crit}, \tag{19}$$

where G_{crit} is the critical energy release rate given by:

$$G_{crit} = \frac{K_{ID}^2(\dot{a})}{E^*} \approx \frac{\hat{K}_{ID}^2}{E^*}$$

and K_{ID} is the dynamic fracture toughness in a pure mode I crack (in general a function of crack speed, \dot{a}); it is approximated by the constant value \hat{K}_{ID} which is given as a material property.

The speed at which the crack will propagate at a vertex (\dot{a}) is then calculated by solving for the roots of the quadratic equation:

$$\Psi\epsilon^2 - (\Psi + 1)\epsilon + \frac{\Gamma - 1}{\Gamma} = 0$$

where $\Gamma = G(0)/G_{crit} \geq 1$, $\Psi = c_R/c_{lim} > 1$, and $\epsilon = \dot{a}/c_R \leq 1$. c_{lim} is the limiting crack speed ($< c_R$) given as an input quantity. c_R is Rayleigh wave speed given as a root of:

$$4\beta_1\beta_2 - (1 + \beta_2^2)^2 = 0,$$

where,

$$\beta_1^2 = 1 - \left(\frac{c_R}{c_d}\right)^2 \quad (c_d \equiv \text{dilatational wave speed} = \sqrt{\frac{\kappa+1}{\kappa-1}}\sqrt{\frac{\mu}{\rho}} \text{ in plane strain}),$$

$$\beta_2^2 = 1 - \left(\frac{c_R}{c_s}\right)^2 \quad (c_s \equiv \text{shear wave speed} = \sqrt{\frac{\mu}{\rho}} \text{ in plane strain}), \text{ and}$$

$\kappa \equiv \text{Kosolov constant} = 3 - 4\nu$ (for plane strain).

In the above, μ is the shear modulus and ρ is the mass density.

4.2 Prescribed Velocity Model

In this model, the propagation angle is calculated using equation (18) above, and the propagation criteria is given by equation (19). The crack speed, however, is calculated according to a pre-defined function. This function is given as an input data. The function is usually a piecewise linear function of time that can take into account for instance the initial time needed for stress waves to reach the crack front.

5 Representation of the Crack Surface

The representation of the crack surface in the proposed method is completely independent of the mesh used. In our present implementation, the crack surface is represented as a set of flat triangles as shown in, for example, Figs. 11(a), 13(b), 27. The crack front is represented by straight line segments connecting the nodes of the triangles along the front. This model of the crack surface is the same used by [23]. The representation of the outer skin of the body is also required by the ‘‘crack geometric engine’’, i.e., the part of the code that handles the crack representation. Figures 11(a) and 27 show examples of the representation of the other skin

of a body. They are as simple as possible and can be composed of several types of geometric entities. The geometric engine handles queries about intersection of segments with the crack surface, distance of a point from the crack surface, orientation of normal and tangent vectors along the crack surface, etc. The geometric engine also updates the crack surface after each crack advancement. It automatically refines the triangles at the crack front in order to ensure a geometrically precise representation of the crack surface. This can easily be implemented since the triangulation of the crack surface does not have to constitute a valid finite element mesh. The geometric engine uses the representation of the outer skin of the body in order to handle surface breaking cracks and cracks intersecting the boundary.

6 Numerical Examples

The generalized finite element method (GFEM) presented previously is used in this section to solve several illustrative examples. In all examples, the stress intensity factors are computed using the least squares fit method presented in 3.

6.1 Single Crack with Mode I Solution Under Static Loading

The edge-cracked panel illustrated in Fig. 10(a) is analyzed in this section using the GFEM. The following parameters are assumed in the computations: $\mathbf{h} = \mathbf{b} = 1.0$, $\mathbf{a} = 0.5$, distributed tractions $\sigma = 1.0$ and uniform thickness $\mathbf{t} = 0.1$. The material is assumed to be linearly elastic with $E = 1000.0$ and $\nu = 0.3$.

The domain is discretized using the hexahedral mesh shown in Fig. 10(b). There are 961 elements in the mesh. A state of plane strain is modeled by constraining the displacement in the z -direction at $z = 0$ and $z = t$. The representation of the crack surface is shown in Fig. 11(a). It is composed of four triangles and two edge elements (used to identify the crack front). These triangles are used only for the geometric representation of the crack surface. There are no degrees of freedom associated with them. The stress intensity factors reported for this problem are computed at $\mathbf{x} = (0.5, 1.0, 0.05)$ which is a point at the crack front located at the middle plane of the body. The geometric definition of the outer skin of the domain (shown on Fig. 11(a)) is also given as input data for the crack geometric engine. The base vectors of the coordinate system associated with singular functions used at the crack front are also displayed in Fig. 11(a). The base vectors and corresponding singular functions are computed completely automatically using the geometric engine. This functionality is specially important during dynamic crack propagation simulations or when the geometric of the domain or crack surface are not so trivial as in this example.

Figure 11(b) shows a closer look at the discretization near the crack front. It can be observed that the crack surface does not respect the element boundaries—*It can arbitrarily cut*

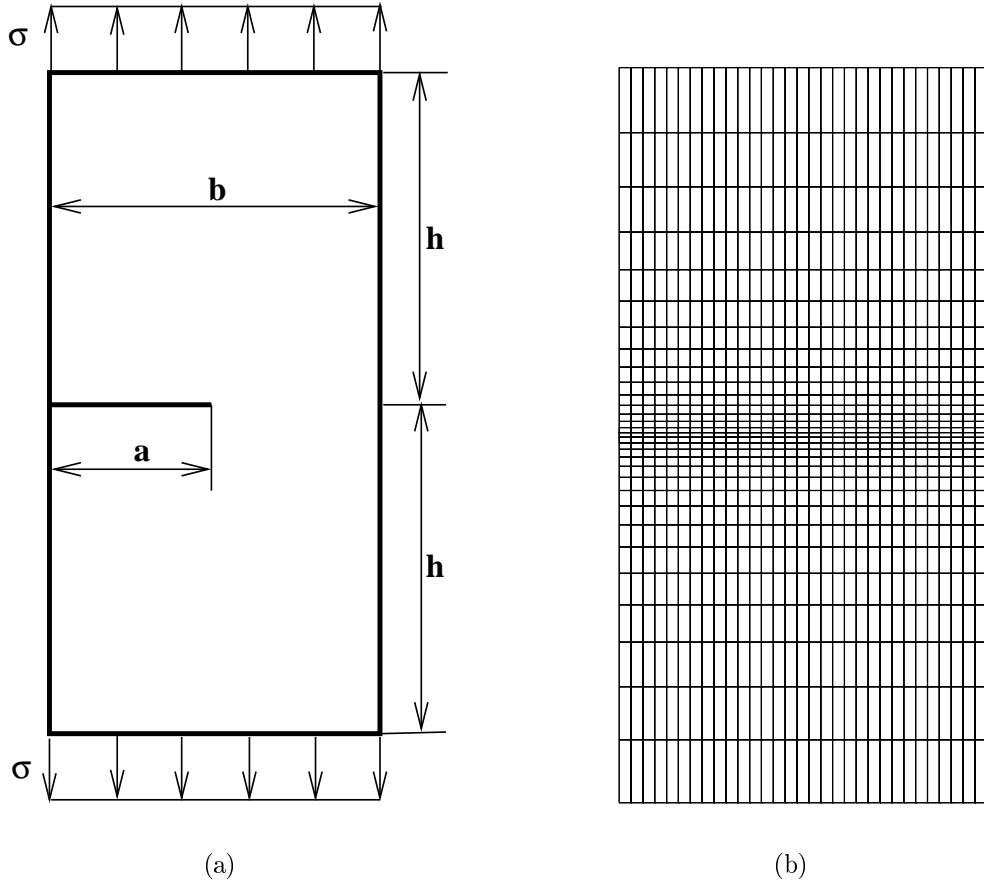
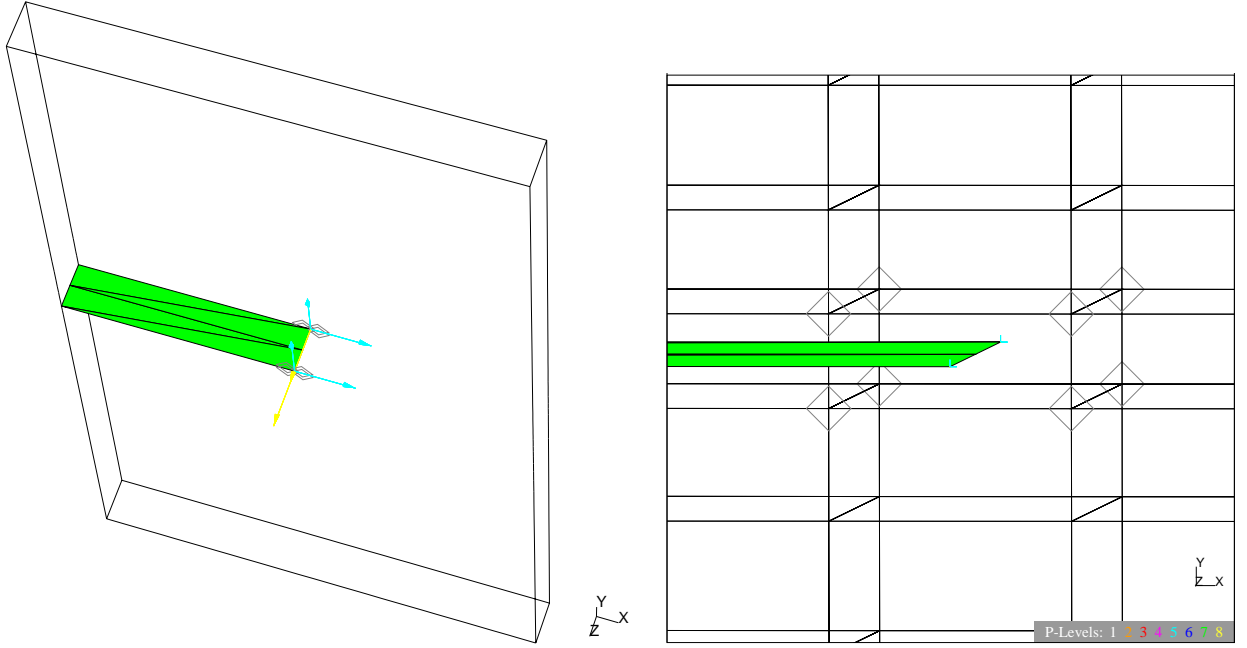


Figure 10: Single edge-notch test specimen and GFEM discretization.

the elements in the mesh. The nodes carrying singular degrees of freedom are represented by diamond-shaped dots. The singular functions used at these nodes are those presented in Section 2.5. Whenever they are used, only the nodes of the elements that contain the crack front are enriched with these singular shape functions (in this example, there is only one element at the crack front). The enrichment of the elements at the crack front with appropriate singular functions is done automatically using the geometric engine to construct appropriate coordinate systems at the crack front.

The computed values of K_I and of the strain energy, U , for several discretizations are shown on Tables 1 and 2. In the tables, Neq denotes the number of equations associated with a particular discretization. The crack is modeled using a FE-Shepard partition of unity as defined in Section 2.3 or 2.7. In the first case, the discontinuity in the displacement field is modeled using the visibility algorithm in combination or not with singular functions. The results using this approach are shown in Table 1. In the second case, the wrap-around approach in combination with singular functions at the crack front is used. The results using



(a) Triangularization of the crack surface and representation of the outer skin of the body.

(b) Zoom at the crack front.

Figure 11: Crack representation.

this approach are shown in Table 2. In both cases, p enrichment is done using the family of functions \mathcal{F}_N^p defined in Section 2.4. The notation $\mathbf{p} = 1 + p_x, 1 + p_y, 1 + p_z = \mathbf{1} + (p_x, p_y, p_z)$ is used to denote the polynomial order of the approximation over *non-cracked* elements. The “1’s” indicating the linear order of the functions defining the partition of unity (linear hexahedral finite element shape functions in this case) and p_x, p_y, p_z denote the degrees of the polynomial basis functions $L_{i\alpha}$ in the x, y, z directions, respectively. The functions $L_{i\alpha}$ are defined in Section 2.4. If only the partition of unity is used, we have $\mathbf{p} = 1 + 0, 1 + 0, 1 + 0 = \mathbf{1} + \mathbf{0}$. A quadratic approximation in the plane XY and linear in the z -direction is denoted by $\mathbf{p} = 1 + 1, 1 + 1, 1 + 0 = \mathbf{1} + (1, 1, 0)$. For cracked elements, the polynomial degree of the approximation is denoted by $\mathbf{p}_c = 0 + p_x, 0 + p_y, 0 + p_z = \mathbf{0} + (p_x, p_y, p_z)$. The “0’s” indicating that, in general, the functions defining the partition of unity over cracked elements can only represent a constant function. A quadratic approximation in the plane XY and linear in the z -direction over cracked elements (fully or partially cracked) is denoted by $\mathbf{p}_c = 0 + 2, 0 + 2, 0 + 1 = \mathbf{0} + (2, 2, 1)$.

The stress intensity factors are computed using the least squares fit method presented in Section 3. The following parameters are used in all computations presented in this section:

- Dimensions of the extraction domain: $\mathbf{d} = (0.4, 1.0, 0.05)$. Which represents a cylinder

of radius 0.4 and length 0.05.

- Number of integration points in the r, θ and z directions: $\mathbf{n} = (10, 20, 1)$, respectively.
- Type of weighting matrix \mathbf{D} : identity matrix
- Power of the weighting functions: 6.

It was observed from numerical experiments on this and other examples that the choice of the matrix \mathbf{D} (material or identity) does not have tangible effects on the calculated values of the stress intensity factors. In addition, different values for the power of the weighting function were tested. A value in the range of [3-6] was observed to be sufficient.

As a reference, the value of K_I computed by Tada et al. [46] using a boundary technique is used. The value $K_I^{Tada} = 3.54259$ reported in [46] has an error smaller than 0.5%.

The discretizations Vis-1, Vis-3 and Vis-5 do not use singular functions at the crack front while the discretizations Vis-2, Vis-4 and Vis-6 do. All 'Vis' discretizations use the visibility approach to build the PU over cracked elements as described in Section 2.3. It can be observed from Table 1 that the use of singular functions gives a noticeable improvement on the computed stress intensity factors while the increase in the number of degrees of freedom is only marginal (less than one percent in the case of the discretization Vis-5). In contrast, the p -enrichment of the cracked elements (discretizations Vis-1 and Vis-3) gives little improvement on the computed K_I . Nonetheless, the enrichment does improve the computed strain energy by about 3%. This behavior indicates that the technique used to compute the stress intensity factor is not optimal since, optimally, the computed stress intensity factors must converge at the same rate as the computed strain energy [44, 45].

The discretizations WA-1, WA-2 and WA-3 use singular functions in combination with the wrap-around technique to build the PU over cracked elements as described in Section 2.7. Comparing the results for the discretizations Vis-2 with WA-1 or Vis-4 with WA-2, it can be observed that, for the same number of degrees of freedom, the wrap-around approach gives better results for the stress intensity factors than the visibility approach. This is in spite of the fact that the discretizations using wrap-around have a smaller strain energy than the corresponding discretizations using visibility. This can be explained by the fact that the visibility approach creates spurious discontinuities in the displacement field near the crack front which results in a less stiff discretization compared with the wrap-around approach (which does not create such spurious discontinuities). It can be observed that the discretization WA-2 gives a better value for K_I than the discretization WA-3 in spite of the fact that the latter gives a larger value for the strain energy than the former. This, again, points to limitations of the technique used to compute the stress intensity factor.

Table 1: GFEM using the partition of unity defined in Section 2.3. The stress intensity factor is computed at (0.5, 1.0, 0.05). In the table, “Discr” stands for discretization, “Sing Fn” stands for singular functions at the crack front, “Neq” stands for number of equations and “U” stands for strain energy.

Discr.	$\mathbf{p} = \mathbf{1}+$	$\mathbf{p}_c = \mathbf{0}+$	Sing Fn	Neq	$U \times 10^4$	K_I	K_I/K_I^{Tada}
Vis-1	(0,0,0)	(1,1,1)	No	6,756	2.27136	3.2387	0.91422
Vis-2	(0,0,0)	(1,1,1)	Yes	6,900	2.32941	3.3741	0.95244
Vis-3	(0,0,0)	(2,2,1)	No	7,776	2.33848	3.2495	0.91727
Vis-4	(0,0,0)	(2,2,1)	Yes	7,920	2.36701	3.3938	0.95800
Vis-5	(1,1,0)	(2,2,1)	No	19,656	2.38643	3.4339	0.96932
Vis-6	(1,1,0)	(2,2,1)	Yes	19,800	2.42281	3.4584	0.97623

Table 2: GFEM using the partition of unity defined in Section 2.7. The stress intensity factor is computed at (0.5, 1.0, 0.05).

Discr.	$\mathbf{p} = \mathbf{1}+$	$\mathbf{p}_c = \mathbf{0}+$	Sing Fn	Neq	$U \times 10^4$	K_I	K_I/K_I^{Tada}
WA-1	(0,0,0)	(1,1,1)	Yes	6,900	2.22465	3.3886	0.95653
WA-2	(0,0,0)	(2,2,1)	Yes	7,920	2.25596	3.4228	0.96618
WA-3	(1,1,0)	(2,2,1)	Yes	19,800	2.29396	3.3431	0.94369

6.2 An Inclined Crack Problem

As another test problem, we consider the cracked panel shown in Fig. 12(a). This problem was analyzed by Szabó and Babuška [44] using the p -version of the finite element method and by Oden and Duarte [35] using the hp Cloud method. In both references, plane stress condition and unit thickness are used. Here, the plane stress condition is approximated by using a small thickness, $t = 0.1$, for the domain compared to the other dimensions. In addition, we assume Young’s modulus $E = 1$, Poisson’s ratio $\nu = 0.3$, distributed traction $\sigma = 1.0$ and $w = 1$ (see Fig. 12(a)). These same values are used in references [44] and [35]. We adopt as a reference, the values of K_I , K_{II} and strain energy, U , computed by Oden and Duarte [35]. They are, respectively,

$$K_I^{Ref} = 1.508284$$

$$K_{II}^{Ref} = -0.729706$$

$$U^{Ref} = 0.170402$$

These values agree very well with those computed by Szabó and Babuška [44] (less than 0.1% difference). The value of the U^{Ref} was scaled to take into account the difference in thickness used here ($t = 0.1$) and adopted by Oden and Duarte [35] ($t = 1.0$).

The discretization of the domain using 765 hexahedral elements is shown on Fig. 12(b).

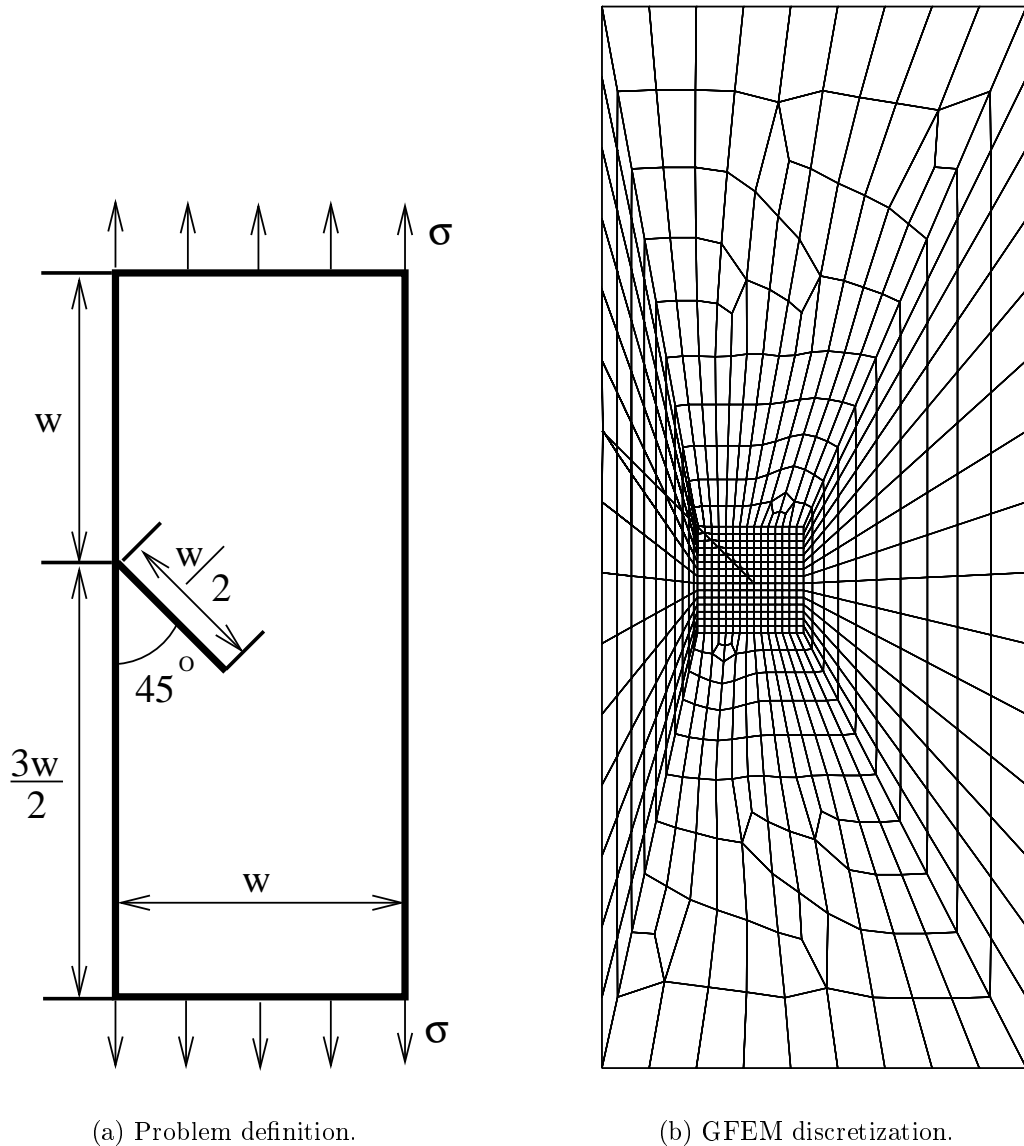
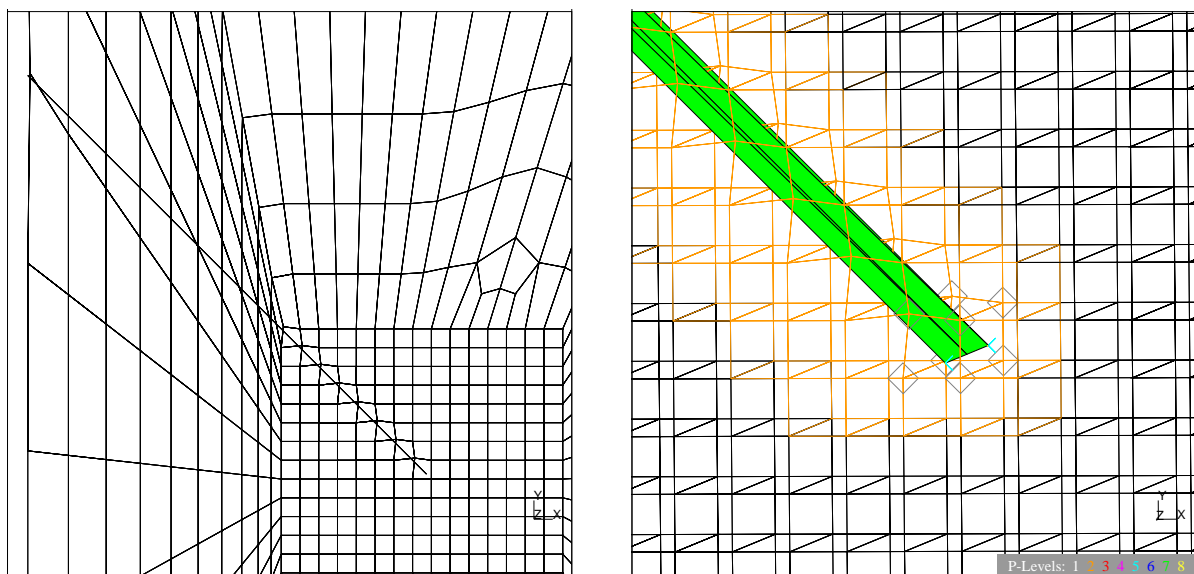


Figure 12: Mixed-mode crack problem.

The inclined crack is also shown in the figure. Figure 13(a) shows a closer look near the crack. It can be observed that the crack surface cuts the elements in the mesh in a quite arbitrary manner. In fact, the meshing of the domain is done as if there is no crack at all. The only consideration used during the meshing of the domain was to use a more refined mesh near the location of the crack front. The crack representation is created and passed to the geometric engine as input data. The geometric engine uses no information whatsoever about the mesh. Nodes in the mesh that are too close to the crack surface are then automatically moved a small distance away from the crack surface (this can be observed in Fig. 13(a) near the crack front).

This is required because an approximation node must be located at one or another side of the crack surface. The representation of the crack surface used here is topologically identical to the one used in the previous example (see Fig. 11(a)). Figure 13(b) shows a closer look at the mesh and crack surface near the crack front. The nodes carrying singular degrees of freedom are represented by diamond-shaped dots. The singular functions used at these nodes are those presented at Section 2.5. As in the previous example, whenever they are used, only the nodes of the elements that contain the crack front are enriched with these singular shape functions.



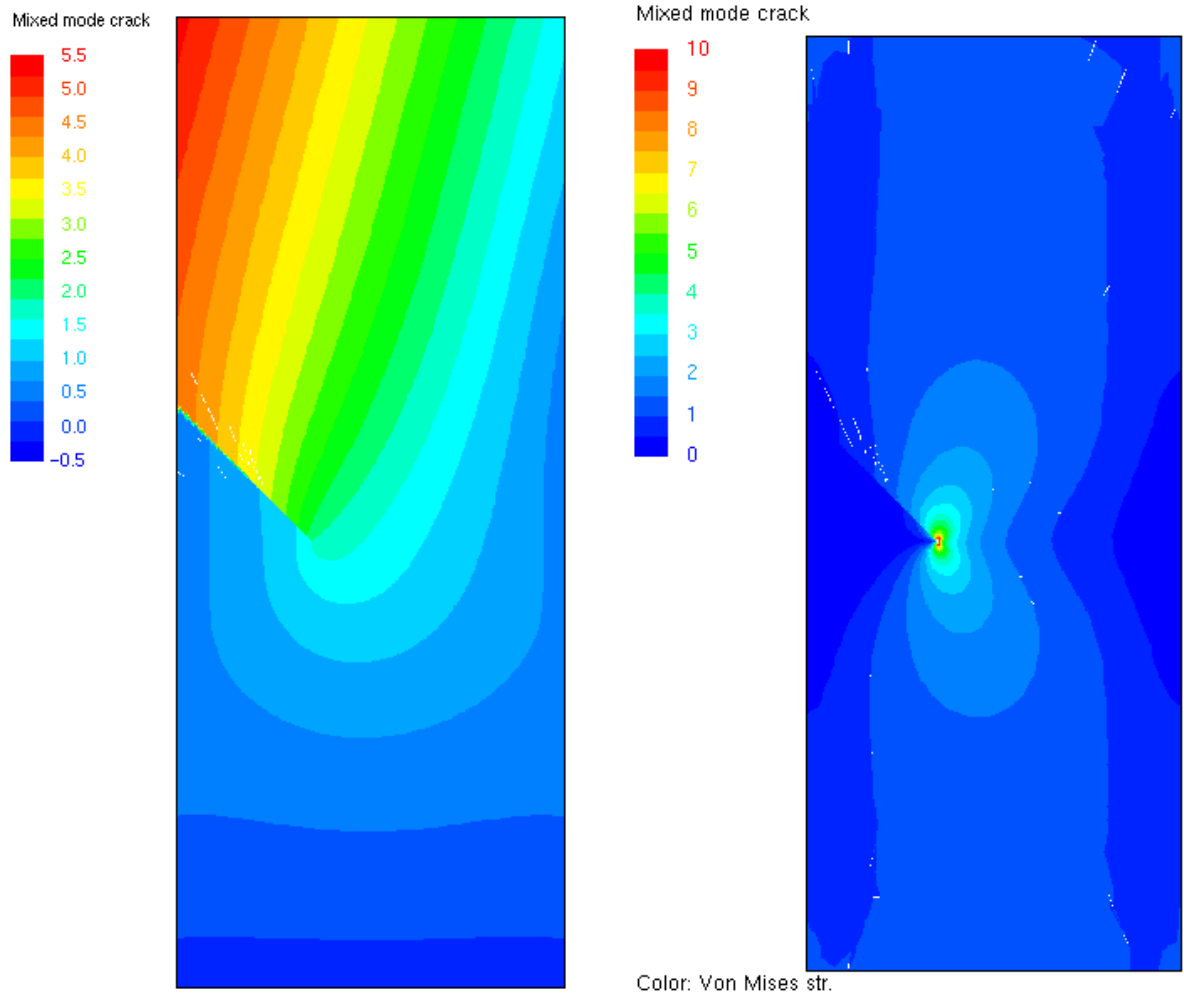
(a) Zoom showing the elements cut by the crack surface.

(b) Three-dimensional view of the crack front.

Figure 13: Crack representation.

The notation used to describe the various discretizations (Vis- i , $i=1,6$ and WA- j , $j=1,3$) is the same as in the previous example. The stress intensity factors are computed at a point in the crack front located at the middle surface of the body. The following parameters are used for extracting the stress intensity factors using the least squares method:

- Dimensions of the extraction domain: $\mathbf{d} = (0.2, 1.0, 0.05)$. Which represents a cylinder of radius 0.4 and length 0.05.
- Number of integration points in the r, θ and z directions: $\mathbf{n} = (10, 40, 1)$, respectively.
- Type of weighting matrix \mathbf{D} : identity matrix
- Power of the weighting functions: 6.

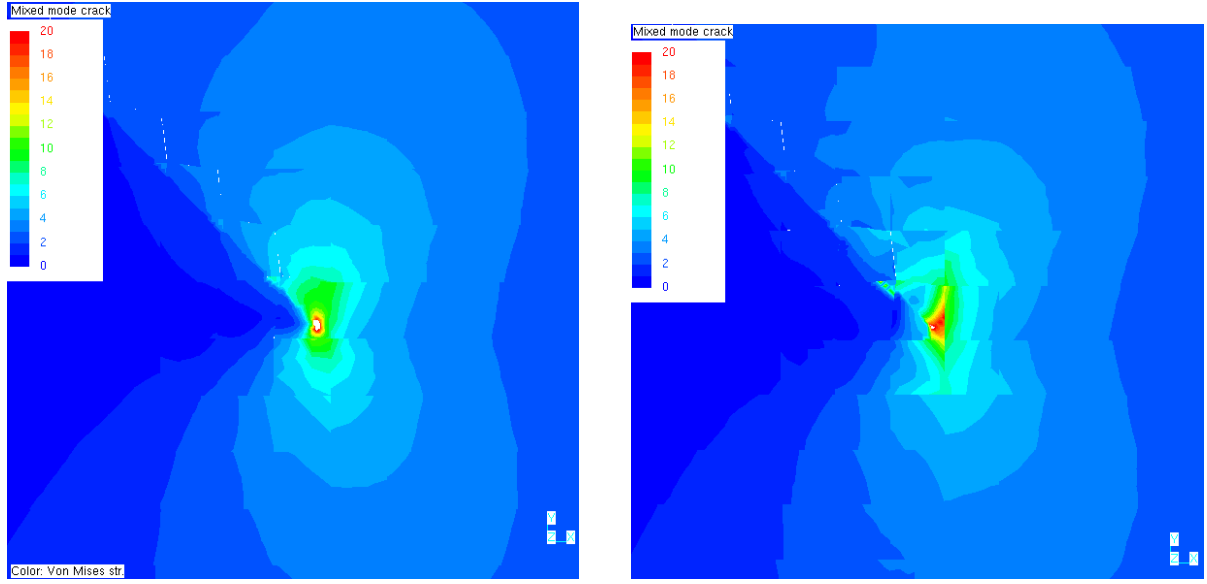


(a) Displacement in the vertical direction.

(b) Von Mises stress field.

Figure 14: Displacement and stress computed using the discretization WA-3.

Figure 14(a) shows a contour plot of the displacement in the vertical direction near the crack computed using the discretization WA-3. The discontinuity in the displacement field constructed using the technique presented in Section 2.7 is clearly observed. Figure 14(b) shows a contour plot for the von Mises stress computed with this discretization and Fig. 15(a) shows a closer look near the crack front. The computed stresses are all raw stresses computed at arbitrary points inside each element. Figure 15(b) shows the same quantity computed using the discretization Vis-5. It can be observed that the stress field is quite disturbed near the crack front. This is caused by the spurious discontinuities created by the visibility approach near the crack front.



(a) Stress computed with discretization WA-3.

(b) Stress computed with discretization Vis-5.

Figure 15: Zoom at the crack front showing von Mises stress.

A summary of the results is presented on Tables 3 and 4. Table 3 shows the computed strain energy for the various discretizations using the PU as defined in Section 2.3 (visibility approach with or without singular functions). It can be observed that the p -enrichment of the approximation has a more significant effect on the strain energy values than the addition of singular functions at the crack front. Nonetheless, as in the previous example, the addition of singular functions improves considerably the computed stress intensity factors. In the case of the discretization Vis-3, for example, the enrichment with singular functions adds only 1.8% more degrees of freedom while the error on the computed value of K_I decreases from 14.0% to only 3.5% and the error on the computed K_{II} decreases from 10.8% to only 1.4%. That is, the error on the computed K_I and K_{II} decrease by 75.0% and 87.0%, respectively.

The results for the discretizations that use the wrap-around approach (Section 2.7) are presented in Table 4. The discretization WA-3 has a relative error in energy, $(U^{Ref} - U)/U^{Ref}$, of only 0.02% which corresponds to a relative error in the energy norm of only 1.41%. This same problem was also solved using the classical hp finite element method with the hp adaptation driven by error indicators based on the element residual method (see, for example, [1]). The results obtained after seven adaptive cycles are shown on Table 5. The relative error in energy and in the energy norm for this discretization are 0.3% and 5.5%, respectively. Note that this discretization has more degrees of freedom (18585) than the discretization WA-3 (16632) but an error in the energy norm almost four times bigger. The reason for this is that the GFEM discretization can capture the singular field near the crack front more efficiently by

using customized singular functions.

Table 3: GFEM using visibility to build the PU over cracked elements. In the table, “Discr” stands for discretization, “Sing Fn” stands for singular functions at the crack front, “Neq” stands for number of equations and “U” stands for strain energy.

Discr	$\mathbf{p} = \mathbf{1}+$	$\mathbf{p}_c = \mathbf{0}+$	Sing Fn	Neq	U/U^{Ref}	K_I/K_I^{Ref}	K_{II}/K_{II}^{Ref}
Vis-1	(0,0,0)	(1,1,1)	No	5874	0.9271	0.7286	0.8052
Vis-2	(0,0,0)	(1,1,1)	Yes	6018	0.9310	0.8391	0.9447
Vis-3	(0,0,0)	(2,2,1)	No	7764	0.9651	0.8595	0.8921
Vis-4	(0,0,0)	(2,2,1)	Yes	7908	0.9686	0.9647	0.9864
Vis-5	(1,1,0)	(2,2,1)	No	16488	1.0044	0.9881	1.0076
Vis-6	(1,1,0)	(2,2,1)	Yes	16632	1.0068	1.1002	1.0589

Table 4: GFEM using wrap-around and visibility to build the PU over cracked elements.

Discr	$\mathbf{p} = \mathbf{1}+$	$\mathbf{p}_c = \mathbf{0}+$	Sing Fn	Neq	U/U^{Ref}	K_I/K_I^{Ref}	K_{II}/K_{II}^{Ref}
WA-1	(0,0,0)	(1,1,1)	Yes	6018	0.9271	0.8735	0.8782
WA-2	(0,0,0)	(2,2,1)	Yes	7908	0.9635	0.9722	0.9472
WA-3	(1,1,0)	(2,2,1)	Yes	16632	0.9998	1.0800	0.9979

Table 5: Results using the hp finite element method and seven adaptive cycles.

Discr	Neq	U	U/U^{Ref}	K_I	K_{II}	K_I/K_I^{Ref}	K_{II}/K_{II}^{Ref}
hp FEM	18585	0.16988	0.9969	1.4680	-0.7041	0.9733	0.9649

6.3 Plate Under Impact Load

In this section, we investigate the performance of the GFEM in modeling propagating cracks in a body subjected to impact loads. The test problem is illustrated on Fig. 16. This problem was analyzed by Lu et al. [29], Krysl and Belytschko [22], Organ [39] and Belytschko and Tabbara [7] using the element free Galerkin method, by Gallego and Dominguez [19] using a boundary element method, among others. A state of plane strain and the following parameters are adopted

- Dimensions: $b = 10.0$, $h = 2.0$, $a = 5.0$ and uniform thickness $t = 0.1$,
- Loading: $\sigma(t) = \hat{\sigma} H(t) = 63750.0 H(t)$, $t \geq 0$. Here, $H(t)$ is the Heaviside step function.

- Material Properties: Linear elastic material with $E = 2.0 \times 10^{11}$, $\nu = 0.3$ and $\rho = 7833.0$.
- Time Step: $\Delta t = 10^{-5}$.

A state of plane strain is modeled by constraining the displacement in the z-direction at $z = 0$ and $z = t$. Two uniform hexahedral meshes are used. The first one has 125, 49 and 1 element in the x-, y- and z-directions, respectively. This same mesh was used in the computations of Krysl and Belytschko [22]. We denote this as the *fine mesh*. The second mesh has 65, 25 and 1 element in the x-, y- and z-directions, respectively. This mesh is denoted as the *coarse mesh*. The representation of the crack surface and of the outer skin of the body are shown in Figs. 17(a) and 17(b). It is composed of five triangles and four edge elements. There are five vertex nodes uniformly spaced at the crack front. The stress intensity factors reported for this problem are computed at $\mathbf{x} = (5.0, 2.0, 0.05)$ and is located at the middle plane of the body. This vertex node is denoted *vertex 3* hereafter.

The stress intensity factors are computed using the least squares fit method. The following parameters are used in all computations presented in this section:

- Dimensions of the extraction domain: $\mathbf{d} = (1.0, 1.0, 0.05)$. Which represents a cylinder of radius 1.0 and length 0.05.
- Number of integration points in the r, θ and z directions: $\mathbf{n} = (5, 30, 1)$, respectively.
- Type of \mathbf{D} matrix equal material matrix
- Power of the weighting function: 3.

Four GFEM discretizations are used (the notation used for \mathbf{p} and \mathbf{p}_c is defined in Section 6.1) :

- Discretization 1:
 - Fine mesh ($125 \times 49 \times 1$ elements),
 - Degree of approximation over non cracked elements $\mathbf{p} = \mathbf{1} + \mathbf{0}$.
 - Degree of approximation over cracked elements $\mathbf{p}_c = \mathbf{0} + \mathbf{1}$.
 - Crack modeled using a FE-Shepard partition of unity and visibility approach as defined in Section 2.3. No singular functions are used.
- Discretization 2: Same as Discretization 1 except that here the crack is modeled using the wrap-around approach and singular functions as defined in Section 2.7.
- Discretization 3: Same as Discretization 1 but using the coarse mesh ($65 \times 25 \times 1$ elements) instead.

- Discretization 4: Same as Discretization 3 but here the degree of approximation over cracked elements is $\mathbf{p}_c = \mathbf{0} + (2, 2, 1)$.

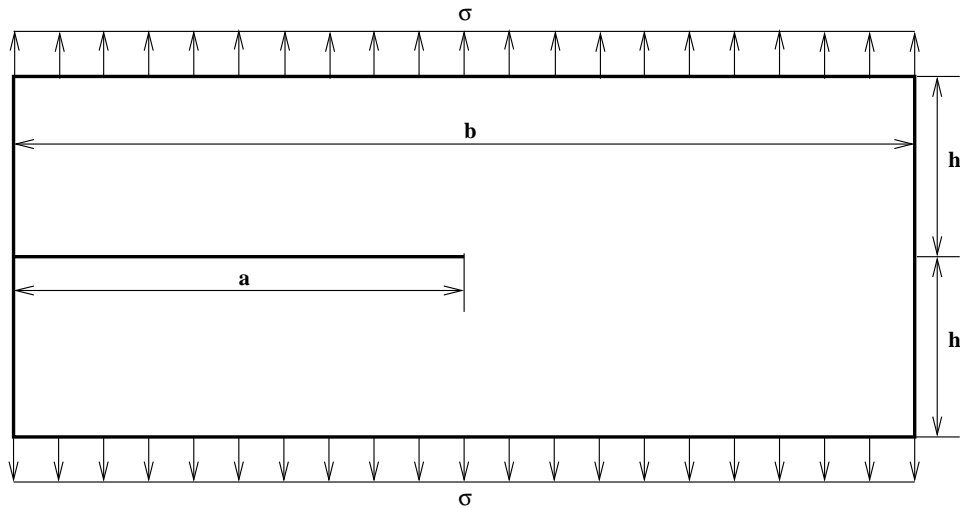
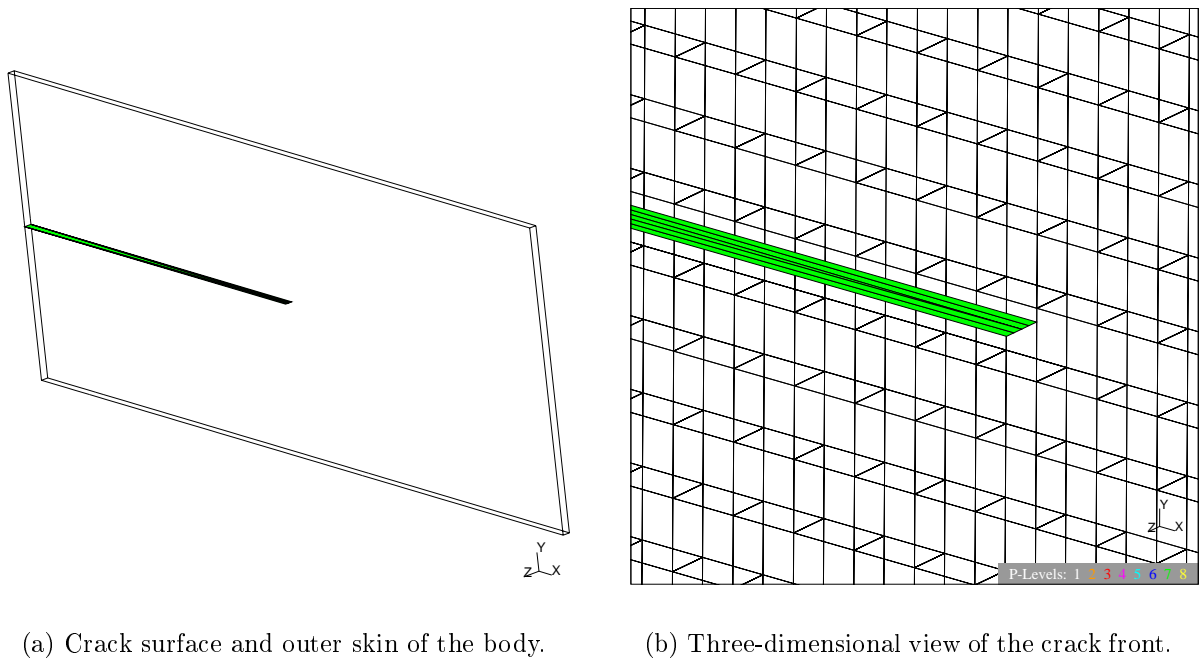


Figure 16: Model problem used to analyze the performance of the GFEM in modeling propagating cracks in a body subjected to impact loads.



(a) Crack surface and outer skin of the body.

(b) Three-dimensional view of the crack front.

Figure 17: Crack representation.

6.3.1 Reference Solution

The GFEM results are compared to the analytic solution for a semi-infinite crack in the plane proposed by Freund [18]. The problem solved by Freund is represented in Fig. 18. The two-dimensional domain has a straight semi-infinite crack, is in a state of plane strain and is loaded by uniformly distributed tractions applied at time $t = 0$. The mode-I stress intensity factor, as a function of time and crack speed C is given by [18].

$$K_I(t, C) = \frac{4\hat{\sigma}H(t - \hat{t})k(C)}{1 - \nu} \sqrt{\frac{(1 - 2\nu)(t - \hat{t})C_d}{\pi}}$$

where,

- $\hat{\sigma}$ is the magnitude of the tensile tractions,
- C_d is the pressure wave speed in the body which is given by

$$C_d = \sqrt{\frac{\mu(\kappa + 1)}{\rho(\kappa - 1)}}$$

where $\mu = \frac{E}{2(1+\nu)}$ and $\kappa = 3 - 4\nu$ (for plane strain state). For the material properties given previously, we get $C_d = 5862.7$.

- \hat{t} is the time the elastic wave hits the crack. For the problem represented in Fig. 16 and $C_d = 5862.7$

$$\hat{t} = 0.000341$$

- $k(C)$ is scaling factor that takes into account that the crack front is advancing with speed C and is given by

$$k(C) = \frac{1.0 - C/C_R}{1.0 - 0.5 * C/C_R}$$

where C_R is the Rayleigh wave speed. For this test problem, with the material properties given above, $C_R = 3030$ (see Section 4).

Due to symmetries, $K_{II} = 0$. The magnitude of the energy release rate is given by

$$G(C, t) = \left(1 - \frac{C}{C_R}\right) G(C = 0, t)$$

where

$$G(C = 0, t) = \frac{K_I^2(C = 0, t)}{E^*}$$

and, for plane strain,

$$E^* = \frac{E}{1 - \nu^2}$$

It should be noted that due to the finite dimensions of the domain modeled here there will be waves reflected by the boundary. This reflected waves will eventually reach the crack front and a comparison of the numerical solution with the above reference solution will no longer be valid. The first reflected wave to reach the crack is a pressure wave after traveling from a loaded edge to the opposite edge and then back to the crack front [22]. This happens at

$$\bar{t} = \frac{3h}{C_d} = \frac{6}{5862.7} = 0.00102$$

A more detailed discussion on the wave patterns that reach the crack front can be found in [22].

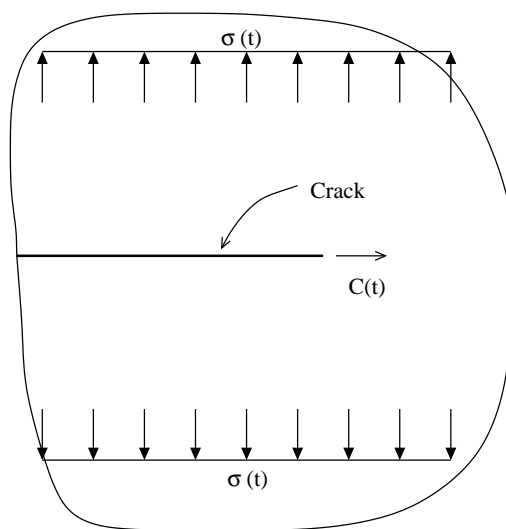


Figure 18: Definition of reference problem.

6.3.2 Stationary Crack

As a first test, we consider the case of a stationary crack. Discretizations 1 and 2 as described above are used. The computed mode-I and mode-II stress intensity factors and the energy release rate G are plotted on Figs. 19 and 20, respectively. It can be observed a good agreement between the computed and reference values for $t < \bar{t}$. The finite dimensions of the extraction domain for the stress intensity factors and of the support of the shape functions are responsible for the non-zero values computed before the pressure wave hits the crack front (at $t = \hat{t}$). It can also be observed that both discretizations give basically identical results.

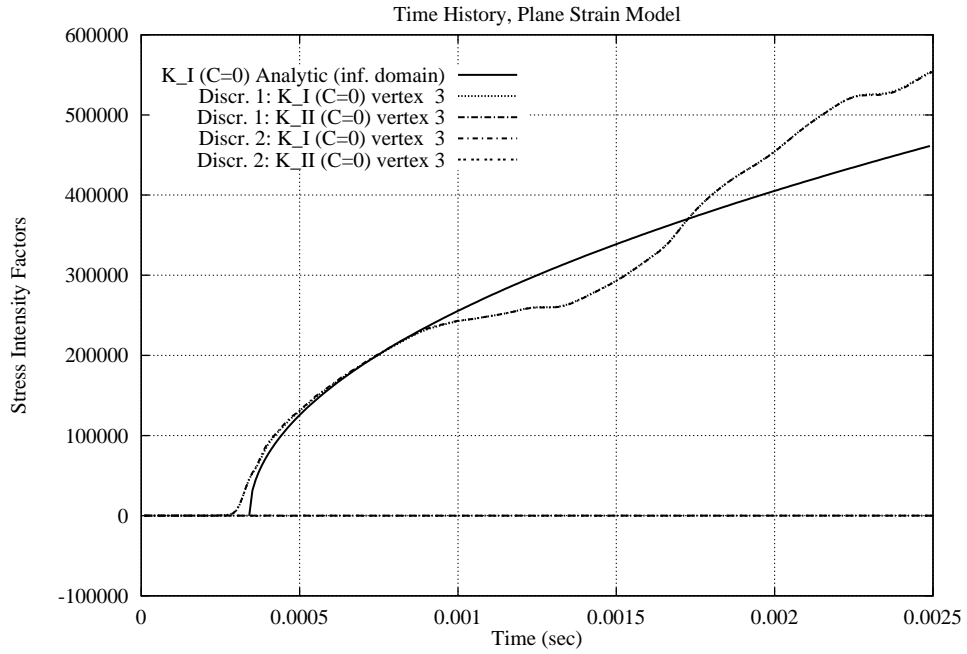


Figure 19: Time history for mode-I and mode-II stress intensity factors K_I and K_{II} using Discretizations 1 and 2. Stationary crack.

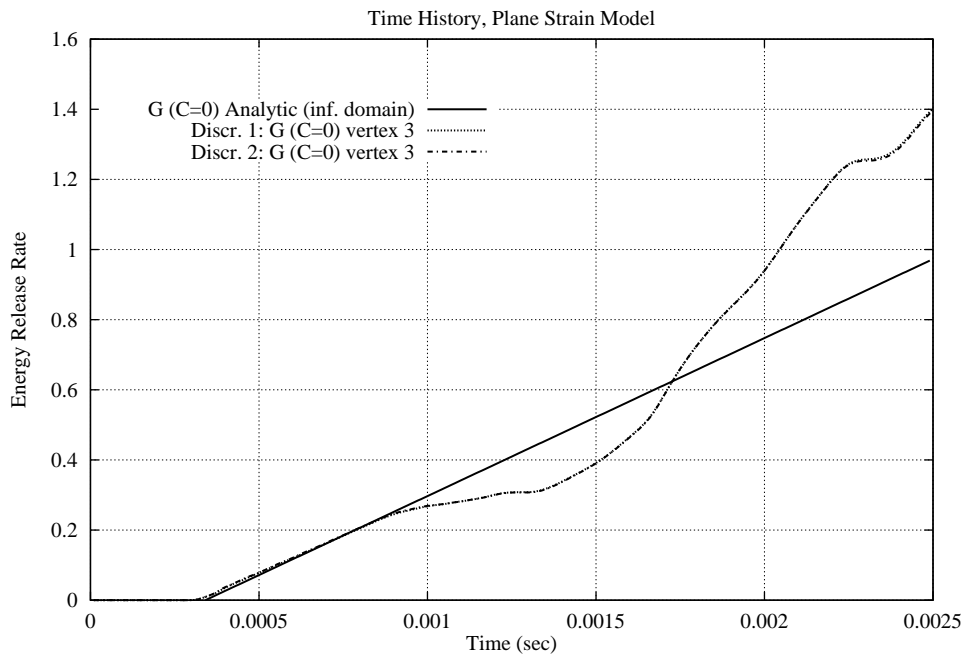


Figure 20: Time history for the energy release rate G using Discretizations 1 and 2. Stationary Crack.

6.3.3 Moving Crack with Prescribed Speed

Here, the crack speed is given by

$$C(t) = H(t - 0.00044) \frac{C_R}{3} = H(t - 0.00044) 1010.0$$

and the direction of the crack advancement is given by (18). Discretization 1 (fine mesh) and 3 (coarse mesh) are used. Figures 21 and 22 show the time history for K_I , K_{II} and G , respectively. While the computed values present some oscillation, they are in good agreement with the reference solution before $t = \hat{t}$ (when reflected waves hits the crack surface). Organ [39], Krysl and Belytschko [22] and Belytschko and Tabbara [7] also reported oscillations on their results obtained with the element free Galerkin method. It can be observed that the GFEM solution was able to capture very well the slop of the reference solution. Figure 23 shows the crack surface at time $t = 0.00168s$.

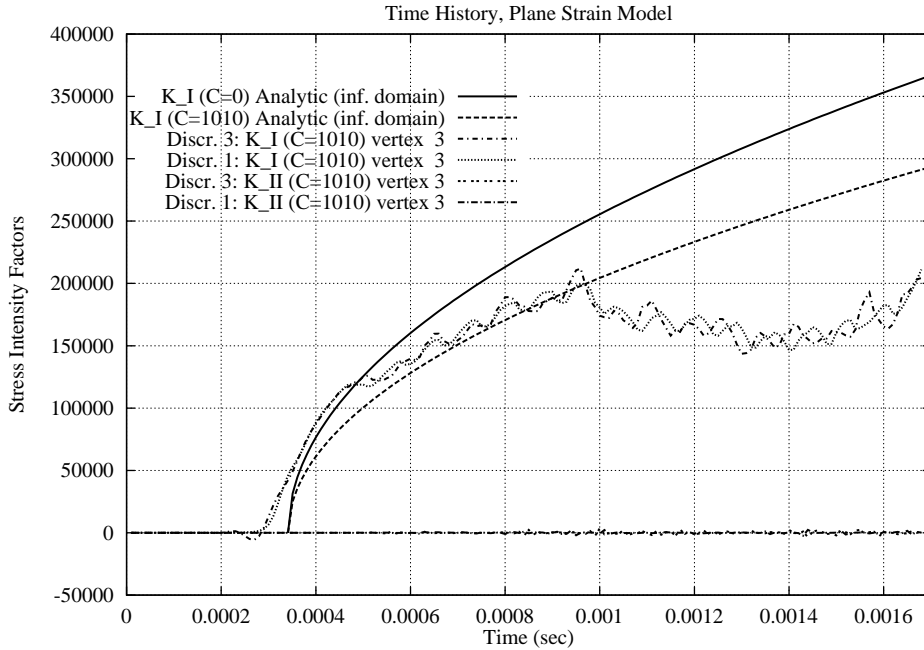


Figure 21: Time history for K_I and K_{II} computed with Discretizations 1 and 3. The crack advancement direction is computed using (18).

The effect of p enrichment of the cracked elements is investigated by using Discretization 4. The results for this discretization are shown on Figs. 24 and 25. It can be observed the the p enrichment of the cracked elements increases the amplitude of the oscillations of the computed quantities. Figure 26 is identical to Fig. 25 but here the tic marks of the x-axis are placed exactly at the times when the crack front crossed the boundary between two elements. It can be observed that the peaks in the oscillations occur just before those instants. Note

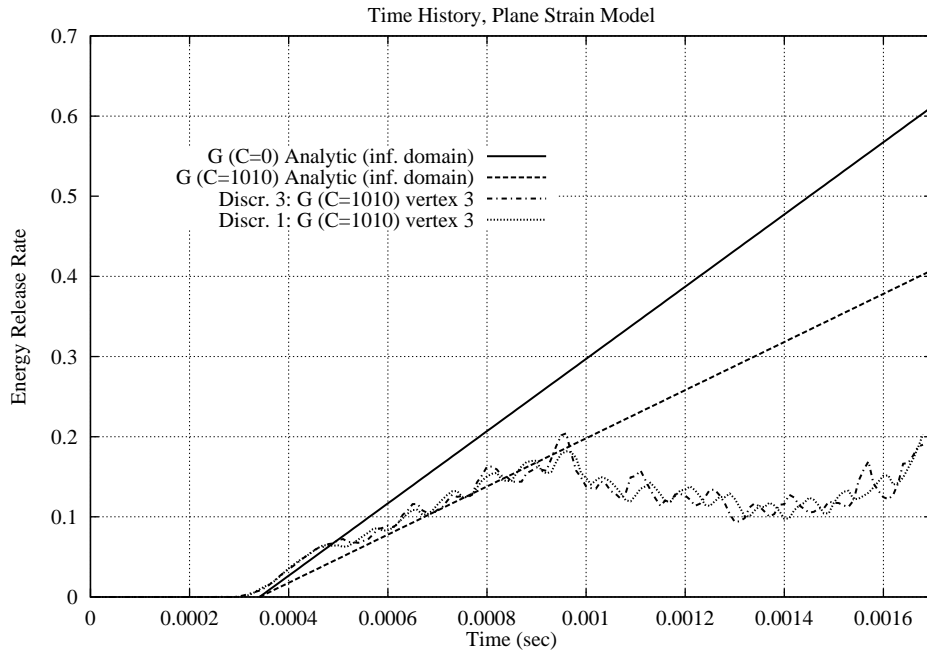


Figure 22: Time history for G computed with Discretizations 1 and 3. The crack advancement direction is computed using (18).

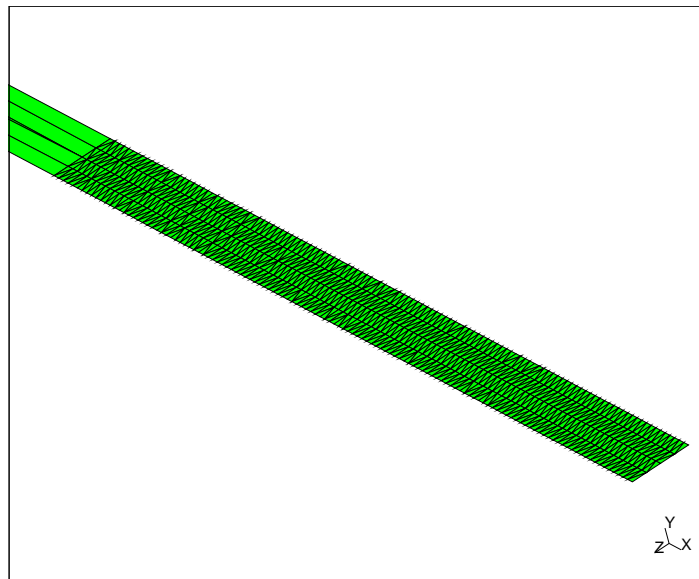


Figure 23: Crack surface at $t = 0.00168s$. Moving crack with advancement direction given by (18).

that as the crack crosses the boundary between two elements it passes close to the nodes. This same phenomena was observed by Organ [39] using the element free Galerkin method.

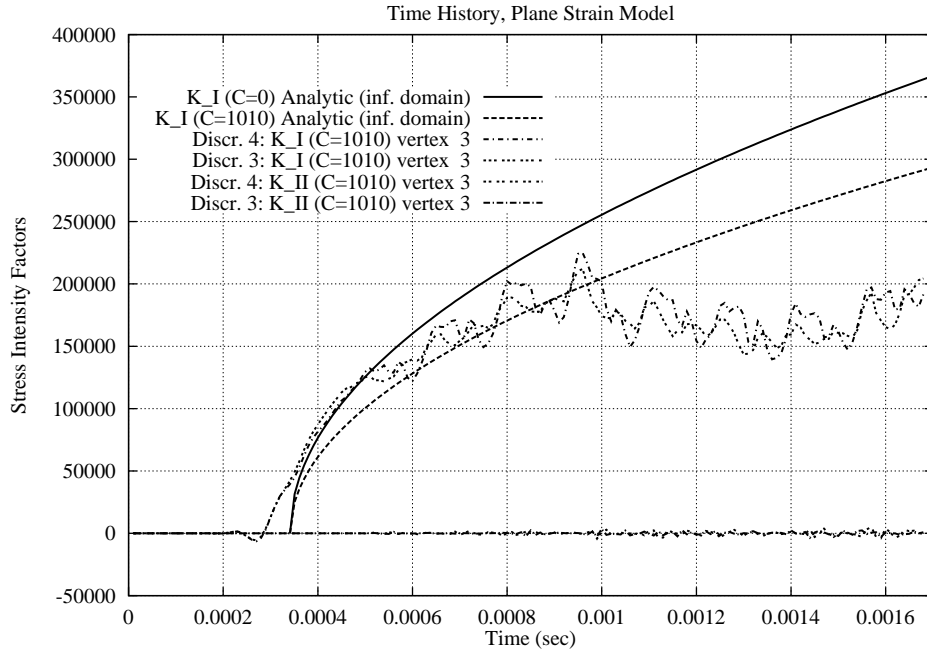


Figure 24: Time history for K_I and K_{II} using coarse mesh and $\mathbf{p}_c = \mathbf{0} + \mathbf{1}$, $\mathbf{p}_c = \mathbf{0} + (2, 2, 1)$ (Discretizations 3 and 4).

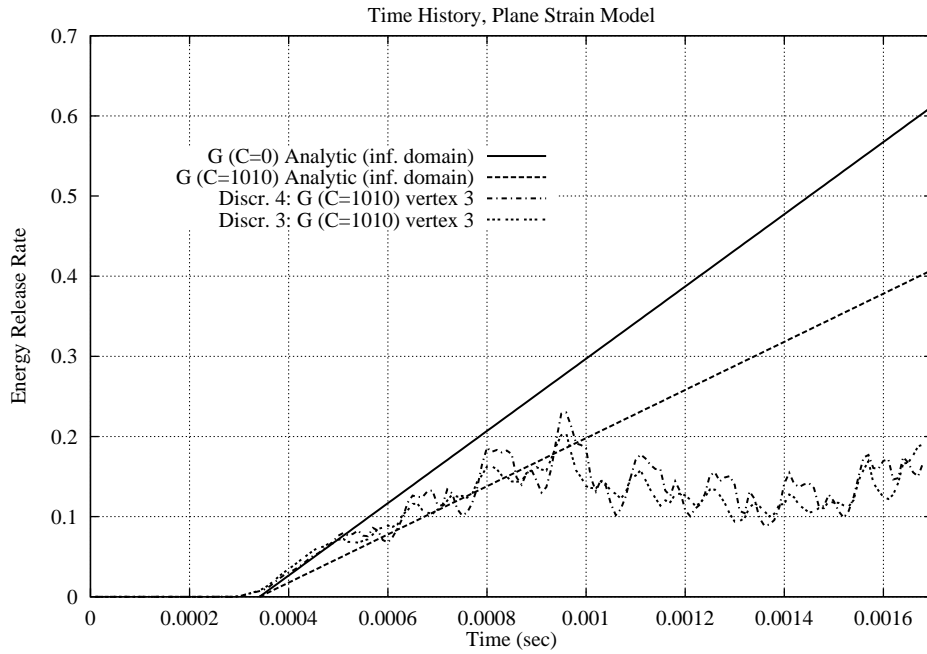


Figure 25: Time history for G using coarse mesh and $\mathbf{p}_c = \mathbf{0} + \mathbf{1}$, $\mathbf{p}_c = \mathbf{0} + (2, 2, 1)$ (Discretizations 3 and 4).

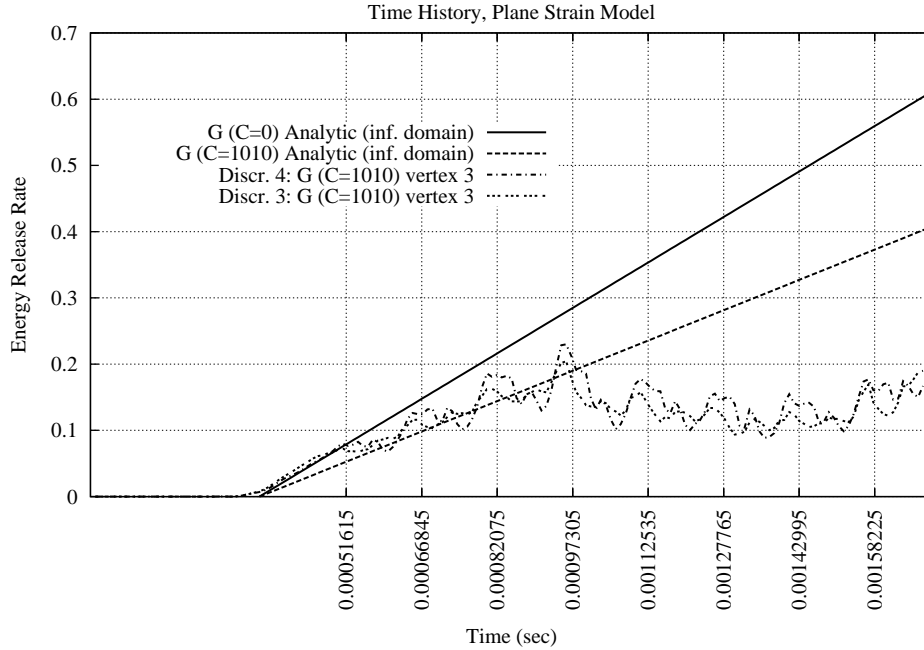


Figure 26: Time history for G using coarse mesh and $\mathbf{p}_c = \mathbf{0} + \mathbf{1}$, $\mathbf{p}_c = \mathbf{0} + (2, 2, 1)$ (Discretizations 3 and 4). The tic marks of the x-axis are placed exactly when the crack front crosses the boundary between two elements.

6.4 Welded T-Joint with a Crack

In this section we present a truly 3D example on using GFEM for crack propagation. The example, as shown in Figure 27, is a beam with welded cross-section (T-section). An initial half-penny crack is placed longitudinally between the weld and the web as shown in the figure. The crack propagates due to an impact couple loading applied at time=0 at the end of the beam. Dynamic waves travel through the body. Once they reach the crack area, stresses increase sharply so that crack is propagated. It is assumed that both domain and loading are symmetric with respect to the other end of the beam and, therefore, only half of the domain is analyzed.

Both the web and the flange are made of the same linear elastic material with the following properties:

- Young's modulus = 200×10^9 .
- Poisson's ratio = 0.3.
- Mass density = 7833.

The weld material is assumed to be ten times stiffer than the latter (*i.e.*, Young's modulus

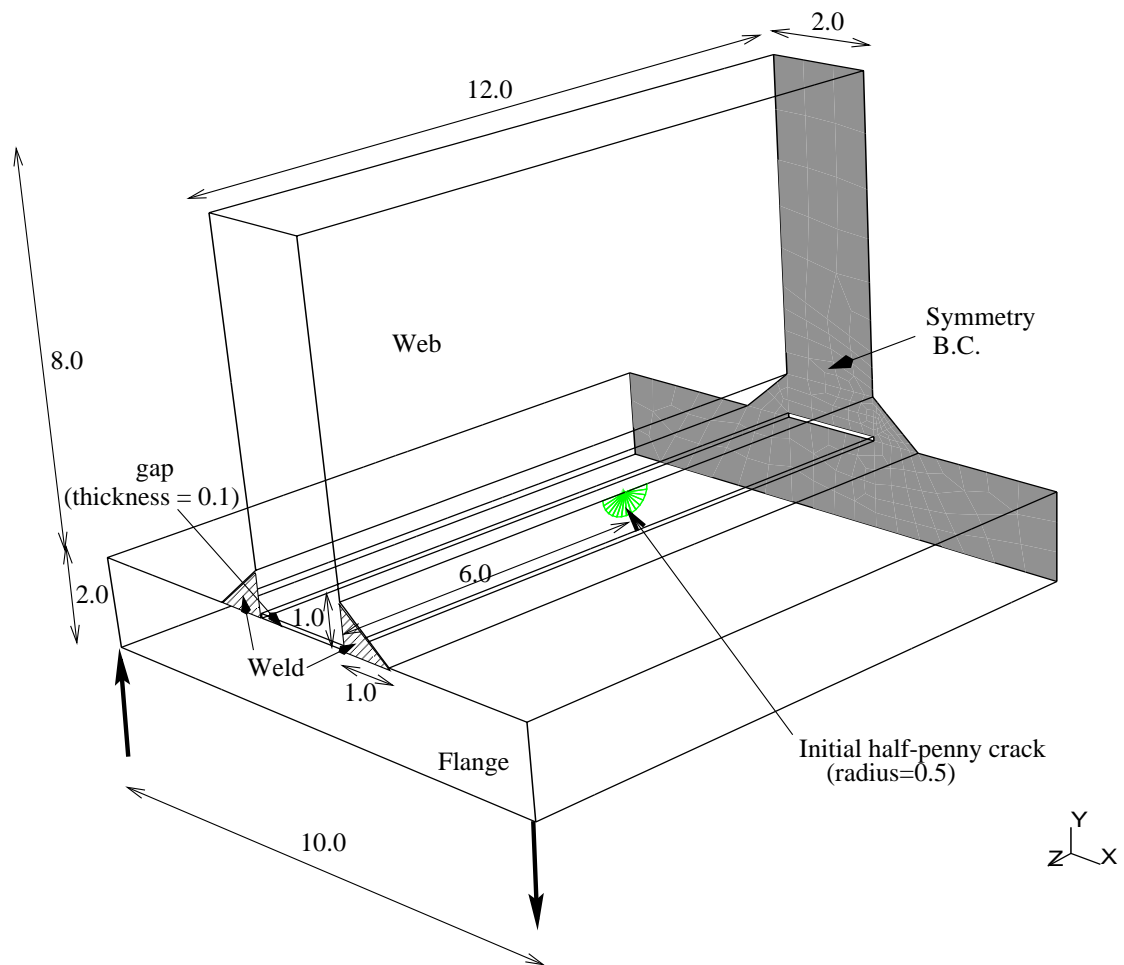


Figure 27: Welded T-section beam with a crack. The initial crack surface is represented using 14 flat triangles. The representation of the outer skin of the body is also shown. The crack surface and the outer skin of the body are used by the geometric engine as described in Section 5.

$= 200 \times 10^{10}$). The Freund propagation model is used to advance the crack (see section 4.1). The dynamic fracture properties and parameters used to propagate the crack are as follows:

- Least squares fit extraction domain: Cylinder of radius 0.5 and length 0.25.
- Number of integration points in the r, θ and z directions: $\mathbf{n} = (5, 10, 3)$, respectively.
- \mathbf{D} is the material matrix.
- Power for the weighting function = 3.
- Dynamic fracture toughness in a pure mode I, $\hat{K}_{ID} = 75000$.
- Crack speed limit = 1212.

The couple is applied as two equal and opposite impact forces at the two corners of the edge cross section with a value of 5×10^6 each.

The initial crack surface definition is shown in Figure 27. Note that the web is not in contact with the flange, but rather a small gap exists between the two. The domain is discretized for GFEM as shown in Figure 28. Note that the grid is made finer around the crack area. Linear approximation is used over all the domain.

A transient dynamic analysis is performed on the model described above. Newmark method is used to march the solution over time. Figures 29, 30, and 31 represent the crack surface at times 0.0015, 0.0020, and 0.0030 respectively. A solution for this problem is not available in the literature. Presentation of this example serves the purpose of demonstrating the capabilities and potential of the proposed methodology to solve three dimensional crack propagation problems in geometrically complicated domains.

7 Conclusions

A partition of unity method for the simulation of three-dimensional dynamic crack propagation is proposed in this paper. The discontinuity in the displacement field across the crack surface is modeled by using a discontinuous Shepard partition of unity to build the shape functions. The PU is computed using Shepard formula, the visibility or wrap-around criteria and FE shape functions as weighting functions. This so-called FE-Shepard partition of unity has several powerful properties. It allows arbitrary cut of the finite element mesh by the crack surface. In fact, the finite element mesh generation can be done as if there is no crack at all in the domain. The crack surface representation is independently created of the finite element mesh and passed to the geometric engine which provides basic functionality like distance from a point to the crack surface, intersection of segments with the crack surface, etc. This high level

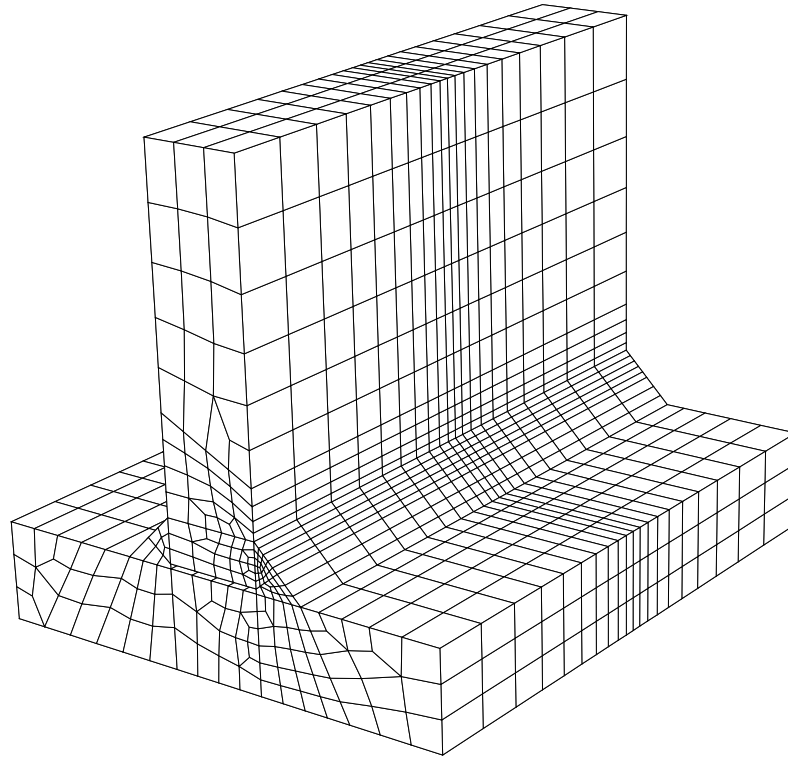


Figure 28: GFEM discretization for welded T-section beam.

of modeling flexibility avoids the continuous remeshing of the domain during the simulation of propagating cracks as done in standard finite element methods. The examples presented in Section 6 illustrate this feature.

The computational cost of the FE-Shepard PU over cracked elements is only marginally higher than usual finite element shape functions and much smaller than, for example, moving least square functions. The numerical integration of the shape functions can also be done as efficiently as in the finite element method since the intersections of these functions coincide with the integration domains.

The FE-Shepard partition of unity degenerates to a standard finite element PU along the boundary between cracked and non-cracked elements and over non-cracked elements. Therefore, it is not necessary to use any special transition element.

Customized shape functions that can reproduce, for example, the asymptotic expansion of the elasticity solution near the crack front can easily and naturally be constructed using the partition of unity framework. The examples presented in Section 6 demonstrate the effectiveness of using customized functions near the crack front.

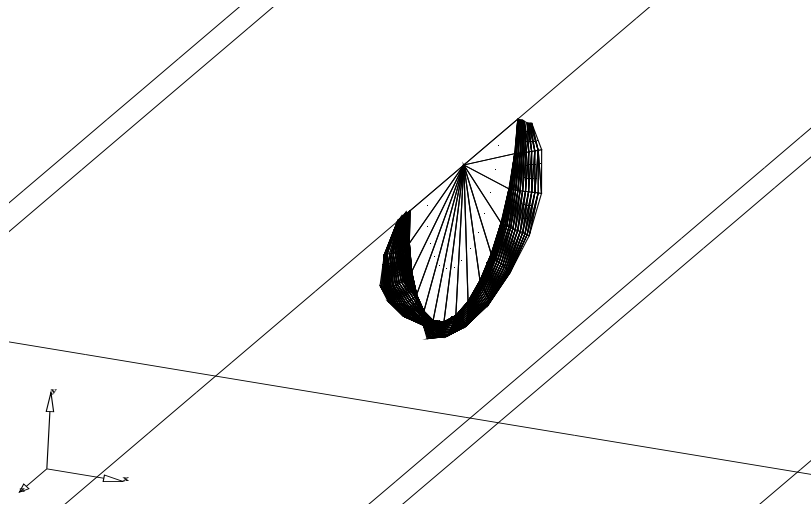


Figure 29: Crack surface of welded T-section example at time 0.0015.

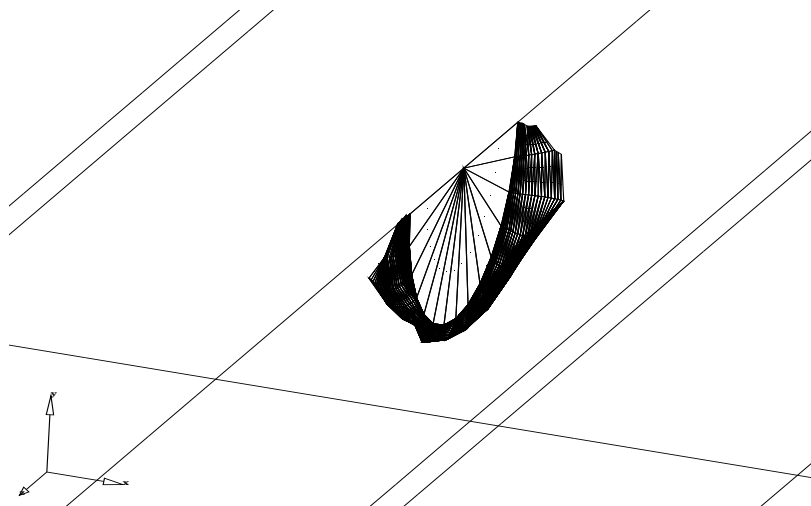


Figure 30: Crack surface of welded T-section example at time 0.0020.

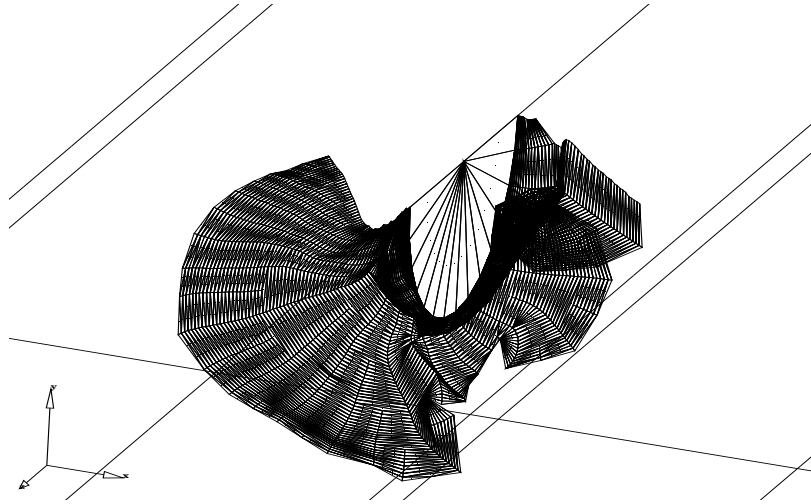


Figure 31: Crack surface of welded T-section example at time 0.0030.

The present method can accommodate very general physics to predict crack direction and speed of propagation since it imposes no restriction on the geometry of the crack surface(s). Most importantly, the implementation of the proposed method is quite straightforward. It is essentially the same as in standard finite element codes, the main difference being the definition of the shape functions. The proposed method can be implemented into most legacy finite element data bases.

Acknowledgements The support of the Office of Naval Research to this project under grant SBIR-ONR-N00014-96-C-0329 is gratefully acknowledged.

References

- [1] M. Ainsworth and J. T. Oden. A posteriori error estimation in finite element analysis. *Computational Mechanics Advances. Special Issue of Computer Methods in Applied Mechanics and Engineering*, 142:1–88, 1997.
- [2] I. Babuška and J. M. Melenk. The partition of unity finite element method. *International Journal for Numerical Methods in Engineering*, 40:727–758, 1997.
- [3] T. Belytschko and T. Black. Elastic crack growth in finite elements with minimal remeshing. *International Journal for Numerical Methods in Engineering*, 45:601–620, 1999.
- [4] T. Belytschko, Y. Krongauz, D. Organ, and M. Fleming. Meshless methods: An overview and recent developments. *Computer Methods in Applied Mechanics and Engineering*, 139:3–47, 1996.
- [5] T. Belytschko, Y. Y. Lu, and L. Gu. Crack propagation by element free Galerkin methods. In *Advanced Computational Methods for Material Modeling*, pages 191–205, 1993. AMD-Vol. 180/PVP-Vol. 268, ASME 1993.
- [6] T. Belytschko, Y. Y. Lu, and L. Gu. Element-free Galerkin methods. *International Journal for Numerical Methods in Engineering*, 37:229–256, 1994.
- [7] T. Belytschko and M. Tabbara. Dynamic fracture using element-free Galerkin methods. *International Journal for Numerical Methods in Engineering*, 39:923–938, 1996.
- [8] B. J. Carter, C.-S. Chen, A. R. Ingraffea, and P. A. Wawrzynek. A topology based system for modeling 3d crack growth in solid and shell structures. In *Proc. 9th Int. Congr. on Fracture*, Sydney, Australia, 1997. Elsevier Science, Amsterdam.
- [9] Guido Dhondt. Automatic 3-D mode I crack propagation calculations with finite elements. *International Journal for Numerical Methods in Engineering*, 41:739–757, 1998.
- [10] J. Dolbow, N. Moes, and T. Belytschko. Discontinuous enrichment in finite elements with a partition of unity method. To appear in *Finite Elements Analysis and Design*.
- [11] C. A. Duarte, I. Babuška, and J. T. Oden. Generalized finite element methods for three dimensional structural mechanics problems. In S. N. Atluri and P. E. O’Donoghue, editors, *Modeling and Simulation Based Engineering*, volume I, pages 53–58. Tech Science Press, October 1998. Proceedings of the International Conference on Computational Engineering Science, Atlanta, GA, October 5-9, 1998.
- [12] C. A. Duarte, I. Babuška, and J. T. Oden. Generalized finite element methods for three dimensional structural mechanics problems. To appear in *Computers and Structures*, 2000.

- [13] C. A. Duarte, O. N. Hamzeh, T. J. Liszka, and W. W. Tworzydło. The element partition method for three-dimensional dynamic crack propagation. In *Fifth US National Congress on Computational Mechanics*, Boulder, CO, Aug. 1999.
- [14] C. A. Duarte, O. N. Hamzeh, T. J. Liszka, and W. W. Tworzydło. Simulation of three-dimensional crack growth under impact loads using a partition of unity method. In *36th Annual Meeting of the Society of Engineering Science*, Austin, TX, Oct 1999.
- [15] C. A. M. Duarte and J. T. Oden. An *hp* adaptive method using clouds. *Computer Methods in Applied Mechanics and Engineering*, 139:237–262, 1996.
- [16] C. A. M. Duarte and J. T. Oden. *Hp* clouds—an *hp* meshless method. *Numerical Methods for Partial Differential Equations*, 12:673–705, 1996.
- [17] C. Armando Duarte. *The hp Cloud Method*. PhD dissertation, The University of Texas at Austin, December 1996. Austin, TX, USA.
- [18] L. B. Freund. *Dynamic Fracture Mechanics*. Cambridge University Press, Cambridge, 1990.
- [19] R. Gallego and J. Dominguez. Dynamic crack propagation analysis by moving singular boundary elements. *J. Appl. Mech, Trans. ASME*, 59:S158–S162, 1992.
- [20] W. H. Gerstle, A. R. Ingraffea, and Perucchio R. Three-dimensional fatigue crack propagation analysis using the boundary element method. *Int. J. Fatigue*, 10:187–192, 1988.
- [21] W. H. Gerstle, L. Martha, and A. R. Ingraffea. Boundary element modeling of crack propagation in three dimensions. *Engng. Comput.*, 2:167–183, 1987.
- [22] P. Krysl and T. Belytschko. A three-dimensional crack growing under a tensile pulse: Simulation by the element free Galerkin method. *Int. J. Solids and Structures*, to appear.
- [23] P. Krysl and T. Belytschko. The element free Galerkin method for dynamic propagation of arbitrary 3-D cracks. *International Journal for Numerical Methods in Engineering*, 44:767–800, 1999.
- [24] P. Lancaster and K. Salkauskas. Surfaces generated by moving least squares methods. *Mathematics of Computation*, 37(155):141–158, 1981.
- [25] P. Lancaster and K. Salkauskas. *Curve and Surface Fitting, an Introduction*. Academic Press, San Diego, 1986.
- [26] S. Li, M. E. Mear, and L. Xiao. Symmetric weak-form integral equation method for three-dimensional fracture analysis. *Computer Methods in Applied Mechanics and Engineering*, 151:435–459, 1998.

- [27] W. K. Liu, Y. Chen, S. Jun, T. Belytschko, C. Pan, R. A. Uras, and C. T. Chang. *Overview and Applications of the Reproducing Kernel Particle Methods*. CIMNE, 1996.
- [28] W. K. Liu, S. Jun, and Y. F. Zhang. Reproducing kernel particle methods. *International Journal for Numerical Methods in Engineering*, 20:1081–1106, 1995.
- [29] Y. Y. Lu, T. Belytschko, and M. Tabbara. Element-free Galerkin method for wave propagation and dynamic fracture. *Computer Methods in Applied Mechanics and Engineering*, 126:131–153, 1995.
- [30] L. F. Martha, P. A. Wawrzynek, and A. R. Ingraffea. Arbitrary crack representation using solid modeling. *Engng. Computers*, 9:63–82, 1993.
- [31] J. M. Melenk and I. Babuška. The partition of unity finite element method: Basic theory and applications. *Computer Methods in Applied Mechanics and Engineering*, 139:289–314, 1996.
- [32] N. Moes, J. Dolbow, and T. Belytschko. A finite element method for crack growth without remeshing. *International Journal for Numerical Methods in Engineering*, 46(1):131–150, 1999.
- [33] B. Nayroles, G. Touzot, and P. Villon. Generalizing the finite element method: Diffuse approximation and diffuse elements. *Computational Mechanics*, 10:307–318, 1992.
- [34] T. Nishioka. The state of the art in computational dynamic fracture mechanics. *JSME Int. J. Ser. A*, 37:313–333, 1994.
- [35] J. T. Oden and C. A. Duarte. Clouds, cracks and fem’s. In B. D. Reddy, editor, *Recent Developments in Computational and Applied Mechanics*, pages 302–321, Barcelona, Spain, 1997. International Center for Numerical Methods in Engineering, CIMNE.
- [36] J. T. Oden, C. A. Duarte, and O. C. Zienkiewicz. A new cloud-based *hp* finite element method. *Computer Methods in Applied Mechanics and Engineering*, 153:117–126, 1998.
- [37] J. T. Oden and C. A. M. Duarte. Solution of singular problems using *hp* clouds. In J. R. Whiteman, editor, *The Mathematics of Finite Elements and Applications 96*, pages 35–54, New York, NY, 1997. John Wiley & Sons.
- [38] D. Organ, M. Fleming, T. Terry, and T. Belytschko. Continuous meshless approximations for nonconvex bodies by diffraction and transparency. *Computational Mechanics*, 18:225–235, 1996.
- [39] D. J. Organ. *Numerical Solutions to Dynamic Fracture Problems Using the Element-Free Galerkin Method*. PhD dissertation, Northwestern University, 1996.
- [40] D. Shepard. A two-dimensional function for irregularly spaced data. In *ACM National Conference*, pages 517–524, 1968.

- [41] T. Strouboulis, I. Babuška, and K. Copps. The design and analysis of the generalized finite element method. *Computer Methods in Applied Mechanics and Engineering*, 81(1–3):43–69, 2000.
- [42] T. Strouboulis, K. Copps, and Babuška I. The generalized finite element method: An example of its implementation and illustration of its performance. *International Journal for Numerical Methods in Engineering*, 47(8):1401–1417, 2000.
- [43] D. Swenson and A. Ingraffea. Modeling mixed-mode dynamic crack propagation using finite elements. *Computational Mechanics*, 3:381–397, 1988.
- [44] B. A. Szabo and I. Babuska. Computation of the amplitude of stress singular terms for cracks and reentrant corners. In T. A. Cruse, editor, *Fracture Mechanics: Nineteenth Symposium, ASTM STP 969*, pages 101–124, 1988.
- [45] Barna Szabo and Ivo Babuska. *Finite Element Analysis*. John Wiley and Sons, New York, 1991.
- [46] H. Tada, P. Paris, and G. Irwin. *The Stress Analysis of Cracks Handbook*. Del Research Corporation, St. Louis, Missouri, 1973.
- [47] G. Xu and M. Ortiz. A variational boundary integral method for the analysis of 3d cracks of arbitrary geometry modeled as continuous distributions of dislocation loops. *International Journal for Numerical Methods in Engineering*, 36:3675–3701, 1993.