Three-Dimensional Non-Planar Crack Growth by a Coupled Extended Finite Element and Fast Marching Method

N. Sukumar[1,*], D. L. Chopp[2], E. Béchet[3], N. Moës[4]

[1] *Department of Civil and Environmental Engineering, University of California, Davis, CA 95616, USA*

[2] *Department of Engg Sciences & Applied Mathematics, Northwestern University, Evanston, IL 60208, USA*

[3] *LPMM - Ile du Saulcy, Université de Metz, 57045 Metz Cedex 1, France*

[4] *Laboratoire de Mécanique et Matériaux, Ecole Centrale de Nantes, 1 Rue de la Noé, 44321 Nantes, France*

SUMMARY

A numerical technique for non-planar three-dimensional linear elastic crack growth simulations is proposed. This technique couples the extended finite element method and the fast marching method. In crack modeling using the extended finite element method, the framework of partition of unity is used to enrich the standard finite element approximation by a discontinuous function and the two-dimensional asymptotic crack-tip displacement fields. The initial crack geometry is represented by two level set functions, and subsequently signed distance functions are used to maintain the location of the crack and to compute the enrichment functions that appear in the displacement approximation. Crack modeling is performed without the need to mesh the crack, and crack propagation is simulated

*Correspondence to: N. Sukumar, Department of Civil and Environmental Engineering, University of California, One Shields Avenue, Davis, CA 95616, USA. E-mail: `nsukumar@ucdavis.edu`

without remeshing. Crack growth is conducted using the fast marching method; unlike a level set formulation for interface capturing, no iterations nor any time step restrictions are imposed in the fast marching method. Planar and non-planar quasi-static crack growth simulations are presented to demonstrate the robustness and versatility of the proposed technique. Copyright © 2007 John Wiley & Sons, Ltd.

KEY WORDS: partition of unity, enrichment function, level sets, signed distance function, fast marching method, stress intensity factor, crack propagation

1. INTRODUCTION

In linear elastic fracture mechanics, the accurate modeling of cracks and crack growth in three dimensions remains a challenging problem. This difficulty is especially pronounced for non-planar crack configurations with the finite element method due to a few factors: (1) accurate solution of the elastostatic boundary-value problem is required in the vicinity of the crack front, and hence mesh refinement around the crack front becomes a necessity; (2) the crack needs to conform to the mesh and with crack advance, remeshing algorithms are needed; and (3) a widely accepted crack growth law in three dimensions is still elusive.

In this paper, we propose a new non-planar crack growth model by coupling the fast marching method (FMM) [1, 2] to a three-dimensional implementation of the extended finite element method (X-FEM) [3]. The two methods form a natural partnership for capturing a monotonically advancing front whose front velocity is obtained via the solution of a system of coupled elliptic equations. In the coupled method, the fast marching method maintains the location and motion of the crack front via signed distance functions, whereas the X-FEM is used to compute the local front velocity. The use of signed distance functions for crack

growth modeling was introduced in Sukumar *et al.* [4, 5] for planar three-dimensional cracks, and subsequently level set algorithms for crack propagation in two dimensions [6] and for arbitrary three-dimensional non-planar cracks [7, 8] were proposed. Propagation of multiple planar cracks using the fast marching method is treated in Reference [9]. In this paper, we combine the implementation for non-planar cracks presented in Moës *et al.* [7] with a fast marching algorithm for crack advance.

In the proposed approach, a fixed Eulerian mesh is used, and no remeshing is required in crack propagation simulations, which are the principal points of departure when compared to existing finite element approaches for crack growth modeling. Furthermore, the meshes used for the mechanical model (extended finite element analysis) and the fast marching method are distinct, which was also the case in the implementation for three-dimensional planar cracks [5]. In References [7,8], the same unstructured tetrahedral mesh is used for the mechanical analysis and for the level set update, and recently, Prabel *et al.* [10] have implemented the level set update algorithm of Gravouil *et al.* [8] on a structured mesh. The use of a structured mesh eases the level set implementation and also facilitates faster convergence [10]. Duflot [11] has presented an overview of techniques used to represent and update level sets for two- and three-dimensional crack propagation.

Some of the prominent numerical methods used for planar and non-planar crack growth in three dimensions are: finite element methods [12, 13], boundary element-based techniques [14–19], and boundary integral equations [20, 21]. Gao and Rice [22] and Lai *et al.* [23] used perturbation analysis to study planar and non-planar cracks, whereas Lazarus and co-workers [24–26] conducted planar crack growth simulations. Apart from the earlier cited works, many recent investigators have also adopted the partition of unity framework for crack growth

simulations in three dimensions [27–31].

The remainder of this paper is organized as follows. In the next section, we provide a self-contained description of the FMM and the X-FEM. In Section 3, details on the non-planar crack growth algorithm using the fast marching method and its coupling to the X-FEM are presented. Numerical simulations for planar and non-planar crack problems appear in Section 4, and we close with some final remarks in Section 5.

2. NUMERICAL TECHNIQUES

We first present an overview of the two underlying numerical methods that are used in this study. The theoretical basis of the methods and a description of the numerical algorithms are outlined.

2.1. Fast Marching Method

The fast marching method [1, 2] represents an evolving interface as a level surface of a higher dimensional function ϕ, where the location of the interface at time T is given by the level set $\{\mathbf{x} : \phi(\mathbf{x}) = T\}$. It is said that ϕ is the time of crossing map for the motion of the interface. For this method, let the initial interface be given by the level set $\{\mathbf{x} : \phi(\mathbf{x}) = 0\}$. The construction of the time-independent function $\phi(\mathbf{x})$ is accomplished by computing the values of $\phi(\mathbf{x})$ on the mesh in an ordered fashion starting from the initial surface and then moving outwards by solving

$$G\|\nabla\phi\| = 1, \tag{1}$$

where G is the local normal velocity of the interface. Thus, the evolution of the interface is computed in one pass through the mesh in contrast to iterative type methods (e.g., level set
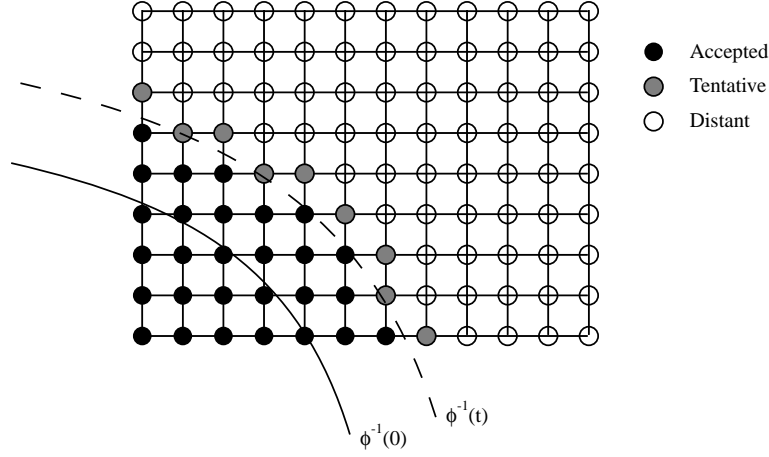
method) previously employed. This gives the fast marching method the advantage of significant speed.

The primary restriction to the fast marching method is that it is only applicable when the speed function is monotonic (G has the same sign everywhere), though efforts to extend this have been proposed in Reference [32]. Even though this is a significant restriction, there are still many applications (e.g., crack propagation as is the case in this investigation) where this method is useful.

The solution of Eq. (1) is constructed by using upwind finite differences to approximate $\nabla\phi$. We solve for the values of ϕ in a monotonically increasing fashion so that the upwind differences are always valid and all the mesh points are eventually computed. This sequential procession through the mesh points is maintained by a heap sort which controls the order in which the mesh points are computed.

To begin, the mesh points are separated into three disjoint sets, the set of *accepted* points A, the set of *tentative* points T, and the set of *distant* points D. The mesh points in the set A are considered computed and are always closer to the initial interface than any of the remaining mesh points. The mesh points in T are all potential candidates to be the next mesh point to be added to the set A. The mesh points in T are always kept sorted in a heap sort so that the best candidate is always easily found. The mesh points in D are considered too far from the initial interface to be possible candidates for inclusion in A. Thus, if $\mathbf{x} \in A$, $\mathbf{y} \in T$, and $\mathbf{z} \in D$, then $\phi(\mathbf{x}) < \phi(\mathbf{y}) < \phi(\mathbf{z})$. Figure 1 shows the relationship between the different sets of mesh points.

One of the key components in the fast marching method is the computation of the estimate of ϕ for points in T. Suppose, for example, mesh points $\mathbf{x}_{i-1,j,k}$, $\mathbf{x}_{i,j+1,k}$, $\mathbf{x}_{i,j,k-1} \in A$, and

Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nmeauth.cls

Int. J. Numer. Meth. Engng 2007; **00**:1–39

Figure 1. Illustration of the sets A, T, and D.

$\mathbf{x}_{i,j,k} \in T$. Given the values of $\phi_{i-1,j,k}$, $\phi_{i,j+1,k}$ and $\phi_{i,j,k-1}$, we must estimate the value of $\phi_{i,j,k}$. This is accomplished by looking at the discretization of Eq. (1) given by

$$\left(\frac{\phi_{i,j,k} - \phi_{i-1,j,k}}{\Delta x} \right)^2 + \left(\frac{\phi_{i,j+1,k} - \phi_{i,j,k}}{\Delta y} \right)^2 + \left(\frac{\phi_{i,j,k} - \phi_{i,j,k-1}}{\Delta z} \right)^2 = \frac{1}{G_{i,j,k}^2}. \qquad (2)$$

Eq. (2) reduces to a quadratic equation in the unknown value $\phi_{i,j,k}$. The new estimate for $\phi_{i,j,k}$ is given by the largest of the two roots of Eq. (2). The remaining configurations and the resulting quadratic equations can be derived in a similar fashion and results in the following formulation:

$$\begin{aligned}
\max & \left(\frac{\phi_{i,j,k} - \phi_{i-1,j,k}}{\Delta x}, -\frac{\phi_{i+1,j,k} - \phi_{i,j,k}}{\Delta x}, 0 \right)^2 \\
& + \max \left(\frac{\phi_{i,j,k} - \phi_{i,j-1,k}}{\Delta y}, -\frac{\phi_{i,j+1,k} - \phi_{i,j,k}}{\Delta y}, 0 \right)^2 \\
& + \max \left(\frac{\phi_{i,j,k} - \phi_{i,j,k-1}}{\Delta z}, -\frac{\phi_{i,j,k+1} - \phi_{i,j,k}}{\Delta z}, 0 \right)^2 = \frac{1}{G_{i,j,k}^2}. \quad (3)
\end{aligned}$$

Now the fast marching method can be assembled as an algorithm:

1. Initialize all the points adjacent to the initial interface with an initial value, put those

points in A. A discussion about initialization follows at the end of this section. All points $\mathbf{x}_{i,j,k} \notin A$, but are adjacent to a point in A are given initial estimates for $\phi_{i,j,k}$ by solving Eq. (3) for the given configuration of neighboring points in A. These points are tentative points and put in the set T. All remaining points are placed in D and given initial value of $\phi_{i,j,k} = +\infty$.

2. Choose the point $\mathbf{x}_{i,j,k} \in T$ that has the smallest value of $\phi_{i,j,k}$ and move it into A. Any point that is adjacent to $\mathbf{x}_{i,j,k}$ (i.e. the points $\mathbf{x}_{i-1,j,k}$, $\mathbf{x}_{i,j-1,k}$, $\mathbf{x}_{i+1,j,k}$, $\mathbf{x}_{i,j+1,k}$, $\mathbf{x}_{i,j,k-1}$, $\mathbf{x}_{i,j,k+1}$) that is in T has its value of ϕ recalculated using Eq. (3). Any point adjacent to $\mathbf{x}_{i,j,k}$ and in D has its value of ϕ computed using Eq. (3) and is moved into the set T.

3. If $T \neq \emptyset$, go to step 2.

For N nodes, the method has a total operation count of $O(N \log N)$. If $G \equiv 1$, then Eq. (1) becomes the Eikonal equation and the solution ϕ gives the distance from $\mathbf{x}$ to the zero contour $\phi^{-1}(0)$. Where Eq. (1) is used in the algorithm presented in this paper, we exclusively use $G \equiv 1$.

The fast marching method is often used in conjunction with the level set method [33]. In the level set method, the interface speed function must be defined at least in a neighborhood of the interface, not just on the interface itself. Therefore, the speed function F defined on the interface $\phi = 0$ must be extended to a function F_{ext} defined on the entire domain in such a way that F_{ext} is constant in the direction normal to the interface [34]. This leads to the equation:

$$\nabla F_{\text{ext}} \cdot \nabla \phi = 0, \tag{4}$$

$$F(x) = F_{\text{ext}}(x)\big|_{\phi(\mathbf{x})=0}.$$

Equation (4) is designed so that $\|\nabla\phi\| = 1$ is maintained in the level set method wherever ϕ is smooth. Eq. (4) is also an evolution equation for F_{ext} with its initial data located on the set $\phi = 0$, and can be discretized using upwind finite differences to get

$$
\left\{ \frac{F_{i,j,k} - F_{i-1,j,k}}{\Delta x} \ \middle| \ -\frac{F_{i+1,j,k} - F_{i,j,k}}{\Delta x} \right\} \max\left(\frac{\phi_{i,j,k} - \phi_{i-1,j,k}}{\Delta x}, -\frac{\phi_{i+1,j,k} - \phi_{i,j,k}}{\Delta x}, 0 \right)
$$

$$
+ \left\{ \frac{F_{i,j,k} - F_{i,j-1,k}}{\Delta y} \ \middle| \ -\frac{F_{i,j+1,k} - F_{i,j,k}}{\Delta y} \right\} \max\left(\frac{\phi_{i,j,k} - \phi_{i,j-1,k}}{\Delta y}, -\frac{\phi_{i,j+1,k} - \phi_{i,j,k}}{\Delta y}, 0 \right)
$$

$$
+ \left\{ \frac{F_{i,j,k} - F_{i,j,k-1}}{\Delta z} \ \middle| \ -\frac{F_{i,j,k+1} - F_{i,j,k}}{\Delta z} \right\} \max\left(\frac{\phi_{i,j,k} - \phi_{i,j,k-1}}{\Delta z}, -\frac{\phi_{i,j,k+1} - \phi_{i,j,k}}{\Delta z}, 0 \right) = 0,
$$

$$(5)$$

where only one of the finite differences of F is chosen to be in the same upwind direction as the corresponding upwind direction for ϕ.

By selecting nodes in the mesh in the same order as for a fast marching method, and using the same upwind finite difference discretization as used in Eq. (3) to discretize Eq. (4), the function F_{ext} can also be computed in a single pass over the mesh.

Finally, one very important, yet often overlooked part of the fast marching method and also its application to velocity extension is how it is initialized. Typically, the interface is given as a level surface, and the initial data is provided on that surface. However, the fast marching method requires initial data to be located on nodes of the mesh that are adjacent to the interface, but not necessarily on the interface. It was shown in Reference [2] that the accuracy of the fast marching method is significantly impacted by the accuracy of initializing the values on the nodes from the data given on the interface. It was also shown in Reference [2] how to improve the initialization process to obtain higher order accuracy for the overall computation using local tricubic interpolants.

Let $V_{ijk} = \{\mathbf{x}_{i,j,k}, \mathbf{x}_{i+1,j,k}, \mathbf{x}_{i,j+1,k}, \mathbf{x}_{i+1,j+1,k}, \mathbf{x}_{i,j,k+1}, \mathbf{x}_{i+1,j,k+1}, \mathbf{x}_{i,j+1,k+1}, \mathbf{x}_{i+1,j+1,k+1}\}$
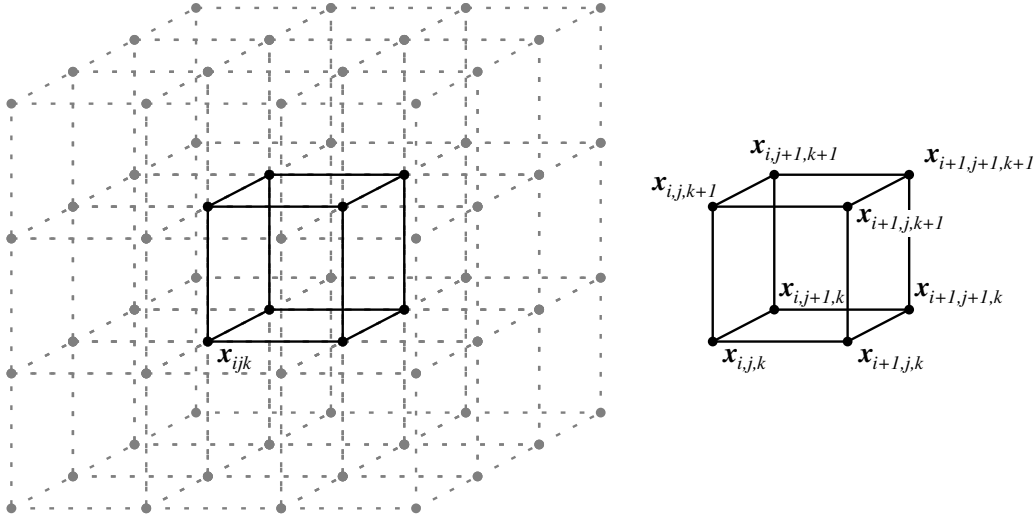
Figure 2. Illustration of a voxel (right) and the surrounding nodes required for generating a tricubic interpolant (left).

be the voxel illustrated in Fig. 2. A voxel contains a given surface $\phi(\mathbf{x}) = 0$ if the set of nodes in the voxel are not all of the same sign. To initialize the FMM, tricubic interpolants are constructed for each voxel that contains the initial surface using the nodes of the voxel and the surrounding nodes as shown in Fig. 2. Let $p(\mathbf{x})$ be the tricubic interpolant for the voxel V_{ijk}. Using this interpolant, the point on the initial surface $p(\mathbf{x}) = 0$, nearest to a given point $\bar{\mathbf{x}}$ is found by solving the pair of equations

$$p(\mathbf{x}) = 0 \tag{6}$$

$$\nabla p(\mathbf{x}) \times (\bar{\mathbf{x}} - \mathbf{x}) = 0.$$

These equations are solved using a modified Newton's method. Once the nearest point $\mathbf{x}$ is found, properties on the initial surface can be mapped from $\mathbf{x}$ to $\bar{\mathbf{x}}$ including the distance to the surface, $\|\mathbf{x} - \bar{\mathbf{x}}\|$, or the interface velocity. This method will be used in a few different places in the final algorithm.

2.2. Extended Finite Element Method

The partition of unity finite element method [35,36] is a generalization of the standard Galerkin finite element method. In the literature, numerical techniques such as the extended finite element method (X-FEM) [3,37] and the generalized finite element method [38] are particular instances of the partition of unity method. The promise of the X-FEM has been in the area of fracture mechanics, with emphasis on the modeling of crack (strong) discontinuities using minimal enrichment. Furthermore, on coupling to level set and fast marching methods, the need for remeshing in crack growth simulations has been alleviated.

Consider a body $\Omega \subset \mathbb{R}^3$, with boundary Γ. The boundary Γ consists of the sets Γ_u, Γ_t, and Γ_c^i, such that $\Gamma = \Gamma_u \cup \Gamma_t \cup_{i=1}^m \Gamma_c^i$. All the internal crack surfaces Γ_c^i are assumed to be traction-free. The field equations for small displacement elastostatics (zero body forces) are:

$$\boldsymbol{\nabla} \cdot \boldsymbol{\sigma} = \mathbf{0} \quad \text{in} \ \ \Omega, \tag{7a}$$

$$\boldsymbol{\sigma} = \mathbf{C} : \boldsymbol{\varepsilon}, \tag{7b}$$

$$\boldsymbol{\varepsilon} = \boldsymbol{\nabla}_s \mathbf{u}, \tag{7c}$$

where $\boldsymbol{\sigma}$ is the Cauchy stress tensor, $\boldsymbol{\varepsilon}$ is the small strain tensor, $\boldsymbol{\nabla}_s$ is the symmetric gradient operator, and $\mathbf{C}$ is the elastic moduli tensor for a homogeneous isotropic material. The essential and natural boundary conditions are:

$$\mathbf{u} = \bar{\mathbf{u}} \quad \text{on} \ \ \Gamma_u, \tag{8a}$$

$$\boldsymbol{\sigma} \cdot \mathbf{n} = \bar{\mathbf{t}} \quad \text{on} \ \ \Gamma_t, \tag{8b}$$

$$\boldsymbol{\sigma} \cdot \mathbf{n} = 0 \quad \text{on} \ \ \Gamma_c^i, \quad (i = 1, 2, \ldots, m), \tag{8c}$$

where $\mathbf{n}$ is the unit outward normal to Ω, $\bar{\mathbf{u}}$ and $\bar{\mathbf{t}}$ are prescribed displacements and tractions, respectively, and m is the number of cracks.

The weak form of the elastostatic boundary-value problem given in Eqs. (7) and (8) is: find $\mathbf{u} \in \mathbf{V}$ such that

$$\int_\Omega \boldsymbol{\sigma} : \delta\boldsymbol{\varepsilon}\, d\Omega = \int_{\Gamma_t} \bar{\mathbf{t}} \cdot \delta\mathbf{u}\, d\Gamma \quad \forall \delta\mathbf{u} \in \mathbf{V}_0. \tag{9}$$

In the X-FEM, trial functions $\mathbf{u}^h(\mathbf{x}) \in \mathbf{V}^h \subset \mathbf{V}$ and test functions $\delta\mathbf{u}^h(\mathbf{x}) \in \mathbf{V}_0^h \subset \mathbf{V}_0$ of the form given in Eq. (10) are used in the weak form. A standard Galerkin procedure is then used to obtain the discrete equations [3, 37].

For three-dimensional crack modeling in the X-FEM [3], the framework of partition of unity [35] is used to introduce a discontinuous function (Heaviside) that has a jump across the crack within the standard displacement-based approximation. Furthermore, for improved accuracy and to model crack fronts that terminate within the interior of a finite element, the plane strain asymptotic crack fields are also used as enrichment functions. The enriched displacement approximation for three-dimensional crack modeling is [3]:

$$\mathbf{u}^h(\mathbf{x}) = \underbrace{\sum_{i \in I} N_i(\mathbf{x})\mathbf{u}_i}_{\text{standard FE}} + \underbrace{\sum_{j \in J} N_j(\mathbf{x})H(\mathbf{x})\mathbf{a}_j}_{\text{Heaviside enrichment}} + \underbrace{\sum_{k \in K} N_k(\mathbf{x})\sum_{\alpha=1}^{4}\psi_\alpha(\mathbf{x})\mathbf{b}_{k\alpha}}_{\text{crack front enrichment}}, \tag{10}$$

where $N_i(\mathbf{x})$ is the finite element shape function of node i, $\mathbf{u}_i$ are the classical degrees of freedom associated with node i, $\mathbf{a}_j$ are the enriched degrees of freedom associated with node j and the Heaviside function $H(\mathbf{x})$ (discontinuous across the crack interior), and $\mathbf{b}_{k\alpha}$ are the enriched degrees of freedom associated with node k and the crack front enrichment functions $\psi_\alpha(\mathbf{x})$, which are defined as [39]

$$\left\{\psi_\alpha(\mathbf{x}),\ \alpha = 1\text{–}4\right\} = \left\{\sqrt{r}\sin\frac{\theta}{2},\ \sqrt{r}\cos\frac{\theta}{2},\ \sqrt{r}\sin\frac{\theta}{2}\sin\theta,\ \sqrt{r}\cos\frac{\theta}{2}\sin\theta\right\}, \tag{11}$$

where r and θ are local crack front polar coordinates of point $\mathbf{x}$ (Fig. 3a). Furthermore, in Eq. (10), I is the set of all nodes in the mesh, the set K consists of nodes that contain the

crack front within their shape function support closure, and J is the set of nodes whose shape functions supports are cut by the crack interior and do not belong to set K:

$$K = \{k : k \in I : \bar{\omega}_k \cap \Lambda_c \neq \emptyset\} \tag{12a}$$

$$J = \{j : j \in I : \omega_j \cap \Gamma_c \neq \emptyset, j \notin K\}, \tag{12b}$$

$$\omega_i = \{\mathbf{x} : N_i(\mathbf{x}) > 0\}, \tag{12c}$$

where ω_i is the support (open set) of $N_i(\mathbf{x})$, Γ_c denotes the crack surface and Λ_c is the crack front. Apart from the selection of nodes for enrichment, partitioning algorithms are used to perform numerical integration of the weak form integrals on either side of the crack interior [3].

3. THREE-DIMENSIONAL NON-PLANAR CRACK GROWTH MODEL

3.1. Level Set Description of Non-Planar Three-Dimensional Cracks

The surface and front of a three-dimensional crack are represented by a pair of functions ϕ_1, ϕ_2 (Fig. 3b). The crack surface will be represented by the zero level set of ϕ_1, while the location of the crack front is captured by a second level set function ϕ_2. Combining the two, the crack discontinuity (open set) and the crack front of the discontinuity are represented by the sets

$$\Gamma_c = \{\mathbf{x} : \phi_1(\mathbf{x}) = 0 \text{ and } \phi_2(\mathbf{x}) \leq 0\}, \tag{13}$$

$$\Lambda_c = \{\mathbf{x} : \phi_1(\mathbf{x}) = 0 \text{ and } \phi_2(\mathbf{x}) = 0\}.$$

Near the crack, ϕ_1, ϕ_2 provide the local coordinate information necessary for evaluating the enrichment functions used in the X-FEM.

The sign of ϕ_1 provides the value of $H(\mathbf{x})$. The function ϕ_1 will be maintained parallel to the crack surface so that x_1^s is tangent to $\phi_1^{-1}(0)$ at all points on the crack front. Similarly, ϕ_2
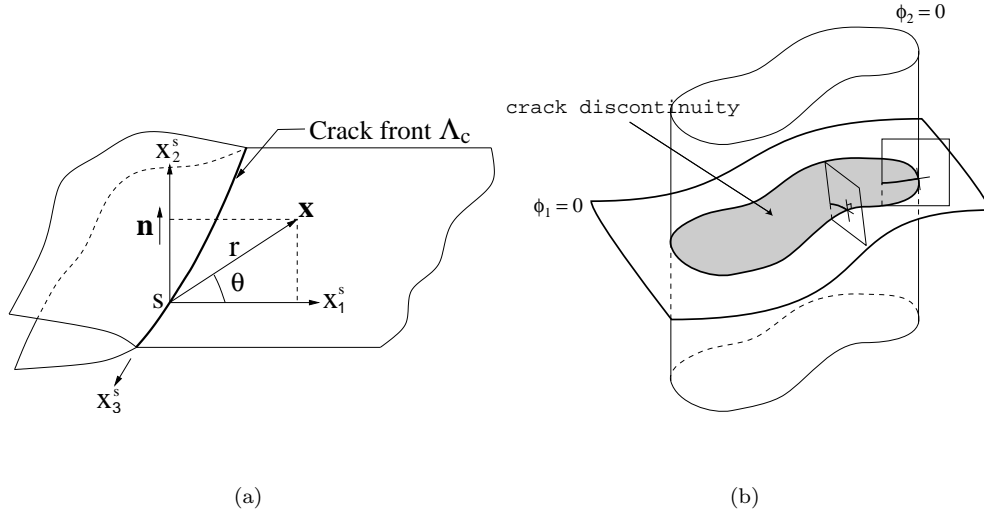
Figure 3. Enrichment and level-set representation of crack. (a) Coordinate configuration for crack front enrichment; and (b) Level set functions ϕ_1 and ϕ_2 to represent the crack surface and front, respectively.

will be maintained orthogonal to the crack plane so that x_2^s is tangent to $\phi_2^{-1}(0)$. For planar cracks, the signed distance functions are simplified: $\phi_1(\mathbf{x})$ is independent of t, and only $\phi_2(\mathbf{x})$ needs to be updated after every time step [4,5].

The advantage of this approach for capturing the crack surface is that important coordinate information necessary for evaluating the enrichment functions can be provided quickly and easily. For instance, the x_1^s coordinate at a point $\mathbf{x}$ is given by $\phi_2(\mathbf{x})$ and the x_2^s coordinate is given by $\phi_1(\mathbf{x})$ [4,5,7]. From this coordinate information, it is then trivial to compute the values of r and θ used in the enrichment functions (see Fig. 3a):

$$r = \sqrt{\phi_1^2 + \phi_2^2}, \quad \theta = \tan^{-1}(\phi_1/\phi_2). \tag{14}$$

The FMM is used in a number of different ways in order to update ϕ_1, ϕ_2, and hence advance the crack front. Before discussing the details, we present a basic overview of the algorithm,

which assumes that we are given ϕ_1^n, ϕ_2^n for the time t_n, and a crack front velocity vector $\mathbf{F} = F_k \mathbf{e}_k$ (Einstein summation convention) has been computed from the X-FEM.

1. The distance function to the crack front, ρ, is computed.

2. The velocity $\mathbf{F}$ is mapped onto grid points near the crack front, and then extended into the rest of the domain by solving

$$\nabla \mathbf{F} \cdot \nabla \rho = 0. \tag{15}$$

3. New functions ϕ_1^{n+1}, $\tilde{\phi}_2^{\ n}$ are constructed using $\mathbf{F}$ and ρ so that Γ_c is left unchanged, but $\phi_1^{n+1} = 0$ is aligned with $\mathbf{F}$, and $\tilde{\phi}_2^{\ n}$ is normal to ϕ_1^{n+1}.

4. $\tilde{\phi}_2^{\ n+1}$ is the result of advancing in time by Δt:

$$\tilde{\phi}_2^{\ n+1} = \tilde{\phi}_2^{\ n} - \|\mathbf{F}\| \Delta t. \tag{16}$$

5. The final ϕ_2^{n+1} is constructed by reinitializing $\tilde{\phi}_2^{n+1}$, i.e., recomputing the signed distance to the set $\tilde{\phi}_2^{n+1} = 0$.

These steps are illustrated in Fig. 4, and in Fig. 5, the distance functions ρ, ϕ_1 and ϕ_2 are illustrated for a penny crack.

In the following subsections we provide more detail for each of the steps. With the objective of making the description self-contained, where necessary, we have reproduced parts of the algorithms presented in Reference [2].

3.1.1. Computing the distance to the crack front   The distance function to the crack front is a critical step in the algorithm for updating the crack front. While Eq. (14) is a reasonable approximation for the purpose of determining enrichment functions, we found that this approximation was not quite accurate enough for a critical step in the advancement of the crack
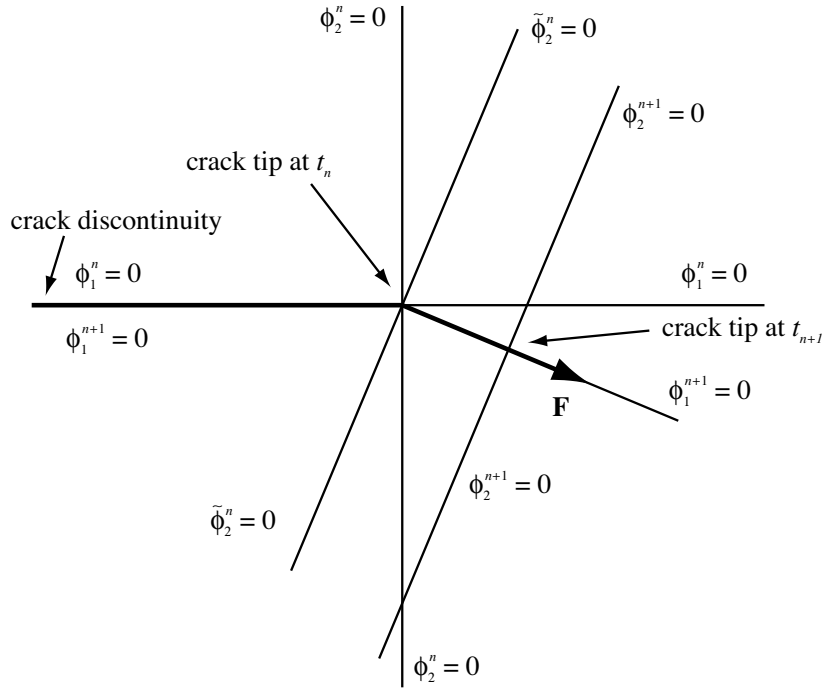
Figure 4. Illustration of the steps required to advance the crack front shown viewed on edge: (1) the initial crack discontinuity is given by $\phi_1^n = 0$ with the crack front at the intersection with $\phi_2^n = 0$, (2) ϕ_1^{n+1} and $\tilde{\phi_2}^n$ are computed so that they are in line with the front velocity vector $\mathbf{F}$, (3) the front is advanced by shifting $\tilde{\phi_2}^{n+1} = \tilde{\phi_2}^n - \|\mathbf{F}\|\Delta t$.

front. So instead, we compute the distance to the crack front using a combination of a local tricubic approximation, coupled with the FMM. The local tricubic approximation provides the initial conditions on grid points near the crack front, and the FMM then computes an approximation of ρ everywhere else.

The tricubic approximation is based on the work in Reference [2], and provides a smooth interpolant between grid point values of a function. The interpolants are computed using local data, but by construction form a globally C^1 function. The interpolant is only valid within each rectangular region bounded by grid points, as illustrated in Fig. 2.
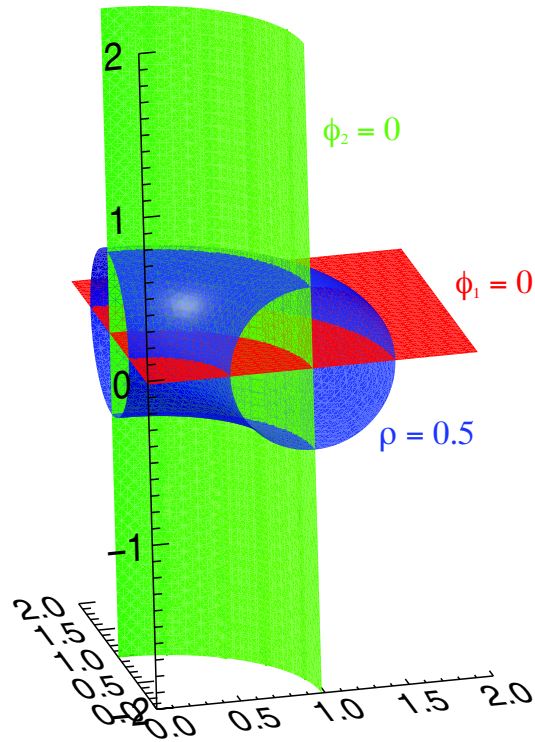
Figure 5. Distance functions in the FMM for a penny crack of unit radius located on $x_3 = 0$.

Let V_{ijk} be a voxel that contains both surfaces $\phi_1 = 0$ and $\phi_2 = 0$, and let p_{ϕ_1}, p_{ϕ_2} be the tricubic interpolants of ϕ_1, ϕ_2 respectively. To find the distance to the crack front from a given point $\bar{\mathbf{x}}$ we must find the point $\mathbf{x}$ that solves the following three equations:

$$p_{\phi_1}(\mathbf{x}) = 0, \tag{17}$$

$$p_{\phi_2}(\mathbf{x}) = 0, \tag{18}$$

$$(\mathbf{x} - \bar{\mathbf{x}}) \cdot (\nabla p_{\phi_1} \times \nabla p_{\phi_2}) = 0. \tag{19}$$

The solution is computed using a modified Newton iteration similar to that used in Reference [2]. Once $\mathbf{x}$ is found, the distance from the point $\bar{\mathbf{x}}$ to the crack front is then

$\rho(\bar{\mathbf{x}}) = \|\bar{\mathbf{x}} - \mathbf{x}\|$. Note that the nearest point to the crack front from a given point $\bar{\mathbf{x}}$ may not be within the given voxel, but because the construction of the tricubic polynomials is globally C^1, the algorithm is able to find at least one point within the crack front that will be closest to $\bar{\mathbf{x}}$, and that point will be contained in one of the voxels.

Let $\mathbf{X}(\bar{\mathbf{x}}, V_{ijk})$ be the point such that

$$p_{\phi_1}(\mathbf{X}(\bar{\mathbf{x}}, V_{ijk})) = 0, \tag{20}$$

$$p_{\phi_2}(\mathbf{X}(\bar{\mathbf{x}}, V_{ijk})) = 0,$$

$$\|\bar{\mathbf{x}} - \mathbf{X}(\bar{\mathbf{x}}, V_{ijk})\| = \min_{\substack{\mathbf{x} \in V_{ijk} \\ p_{\phi_1}(\mathbf{x})=0 \\ p_{\phi_2}(\mathbf{x})=0}} \|\bar{\mathbf{x}} - \mathbf{x}\|.$$

The value of $\mathbf{X}(\bar{\mathbf{x}}, V_{ijk})$ is computed using the tricubic interpolant described above. The initial distance to the crack front is then computed on each grid point near the crack front. A point is near the crack front if it is in the neighborhood of a voxel that contains the crack front as illustrated in the left panel of Fig. 2. At that point, the distance function is given by

$$\rho(\bar{\mathbf{x}}) = \min_{i,j,k} \|\bar{\mathbf{x}} - \mathbf{X}(\bar{\mathbf{x}}, V_{ijk})\|. \tag{21}$$

Once the distance has been computed on each of the grid points surrounding the crack front, these points are used as initial data for the FMM. In this case, because we want the distance function, the FMM is used to solve the Eikonal equation:

$$\|\nabla \rho\| = 1. \tag{22}$$

This results in computing the distance function from the crack front on all the grid points in a single pass through the grid.

3.1.2. Extending the tip velocity The extension of the crack front velocity on the grid is also accomplished using the FMM. As for the distance, the velocity data must first be mapped onto

the grid points surrounding the crack front. For simplicity, we use the same points initialized by the tricubic interpolant when computing the distance function.

Note that since the X-FEM and the FMM are not using the same meshes, the velocity data is not given at grid points and therefore must be interpolated onto the grid points. Suppose the velocity vector is to be computed at grid point $\mathbf{x}_{ijk}$, and suppose the front velocity data is provided as a list of sample coordinates, $\mathbf{x}_\ell$, and the corresponding front velocity vector, $\mathbf{F}_\ell$. We search for the two sample coordinates closest to $\mathbf{x}_{ijk}$, say $\mathbf{x}_{\ell_1}$, $\mathbf{x}_{\ell_2}$. Let $\bar{\mathbf{x}} = (1 - \alpha)\mathbf{x}_{\ell_1} + \alpha \mathbf{x}_{\ell_2}$, $0 \le \alpha \le 1$, be the point on the segment with ends $\mathbf{x}_{\ell_1}$, $\mathbf{x}_{\ell_2}$ nearest to $\mathbf{x}_{ijk}$. We then interpolate the value of the velocity vector at the same point, hence

$$\mathbf{F}_{ijk} = (1 - \alpha)\mathbf{F}_{\ell_1} + \alpha \mathbf{F}_{\ell_2}. \tag{23}$$

Once the velocity data is mapped onto the grid points near the crack front, the velocity is extended to the rest of the domain using the fast marching method as described in Reference [34]. In this case, since the distance to the crack front is already computed, the velocity extension is computed in the same order as ρ and using the upwind approximation from Eq. (5) for the equation

$$\nabla F_k \cdot \nabla \rho = 0, \quad (k = 1, 2, 3). \tag{24}$$

3.1.3. Computing ϕ_1^{n+1} and $\tilde{\phi}_2^{\ n}$ Now that both the distance to the crack front and the crack front velocity have been extended to the entire domain, the calculation of the new surface and orthogonal front functions are exercises in geometry. Points that are near the crack surface and behind the crack front are left alone, but points that are ahead of the crack front need to be recomputed. This leads to a set of different regions where the new surface and front values are to be computed. These regions are illustrated in Fig. 6.

Figure 6. The domain is broken into four regions. A point $\mathbf{x}$ is in region: (B) if $\phi_2^n(\mathbf{x}) \leq 0$ and $\tilde{\phi}_2^{\,n}(\mathbf{x}) \leq 0$; (A) if $\phi_2^n(\mathbf{x}) > 0$ and $\tilde{\phi}_2^{\,n}(\mathbf{x}) > 0$; (I) if $\phi_2^n(\mathbf{x}) \leq 0$ and $\tilde{\phi}_2^{\,n}(\mathbf{x}) > 0$; (O) if $\phi_2^n(\mathbf{x}) > 0$ and $\tilde{\phi}_2^{\,n}(\mathbf{x}) \leq 0$.

Clearly, the determination of which region $\mathbf{x}$ belongs relies on the value of $\tilde{\phi}_2^{\,n}(\mathbf{x})$. The value of $\tilde{\phi}_2^{\,n}(\mathbf{x})$ can be computed by using the gradient of the distance function coupled with the extended velocity $\mathbf{F}(\mathbf{x})$:

$$\tilde{\phi}_2^{\,n}(\mathbf{x}) = \rho(\mathbf{x}) \frac{\nabla \rho(\mathbf{x}) \cdot \mathbf{F}(\mathbf{x})}{\|\nabla \rho(\mathbf{x})\| \|\mathbf{F}(\mathbf{x})\|}, \tag{25}$$

which is geometrically depicted in Fig. 7b. If $\|\mathbf{F}(\mathbf{x})\| < \epsilon$ for some small threshold ϵ, then for stability we take $\tilde{\phi}_2^{\,n}(\mathbf{x}) = \phi_2^n(\mathbf{x})$. In our code, we used $\epsilon = 10^{-7}$.

We can now identify to which region $\mathbf{x}$ belongs, and then compute the value of $\phi_1^{n+1}(\mathbf{x})$. The goal is to get an extrapolation of the crack surface in the direction of the front velocity

F. The regions are:

B: If $\phi_2^n(\mathbf{x}) \leq 0$ and $\tilde{\phi}_2^{\,n}(\mathbf{x}) \leq 0$, then $\mathbf{x}$ is behind the crack front before and after the crack front direction has changed. The values of ϕ_1 and ϕ_2 will remain unchanged in this region to preserve the location of the crack surface. Therefore,

$$\phi_1^{n+1}(\mathbf{x}) = \phi_1^n(\mathbf{x}). \tag{26}$$

A: If $\phi_2^n(\mathbf{x}) > 0$ and $\tilde{\phi}_2^{\,n}(\mathbf{x}) > 0$, then $\mathbf{x}$ is ahead of the crack both before and after the current update. In this case, ϕ_1 will be determined by the orthogonal distance to a linear extrapolation of the crack front in the direction of the crack velocity $\mathbf{F}$. This leads to the formula

$$\phi_1^{n+1}(\mathbf{x}) = \rho(\mathbf{x}) \frac{\nabla\rho(\mathbf{x})}{\|\nabla\rho(\mathbf{x})\|} \cdot \frac{\big(\mathbf{F}(\mathbf{x}) \times \nabla\phi_1^n(\mathbf{x})\big) \times \mathbf{F}(\mathbf{x})}{\|\big(\mathbf{F}(\mathbf{x}) \times \nabla\phi_1^n(\mathbf{x})\big) \times \mathbf{F}(\mathbf{x})\|}, \tag{27}$$

which is shown in Fig. 7a.

I: If $\phi_2^n(\mathbf{x}) \leq 0$ and $\tilde{\phi}_2^{\,n}(\mathbf{x}) > 0$, then $\mathbf{x}$ is in the inside part of the turn. In this case, the proper distance to the crack surface will be the minimum of the distance to the crack surface behind the crack front and the extrapolated crack front in the direction of the crack velocity $\mathbf{F}$. Thus,

$$\phi_1^{n+1}(\mathbf{x}) = \text{minmod}\left\{\phi_1^n(\mathbf{x}), \rho(\mathbf{x}) \frac{\nabla\rho(\mathbf{x})}{\|\nabla\rho(\mathbf{x})\|} \cdot \frac{\big(\mathbf{F}(\mathbf{x}) \times \nabla\phi_1^n(\mathbf{x})\big) \times \mathbf{F}(\mathbf{x})}{\|\big(\mathbf{F}(\mathbf{x}) \times \nabla\phi_1^n(\mathbf{x})\big) \times \mathbf{F}(\mathbf{x})\|}\right\}, \tag{28a}$$
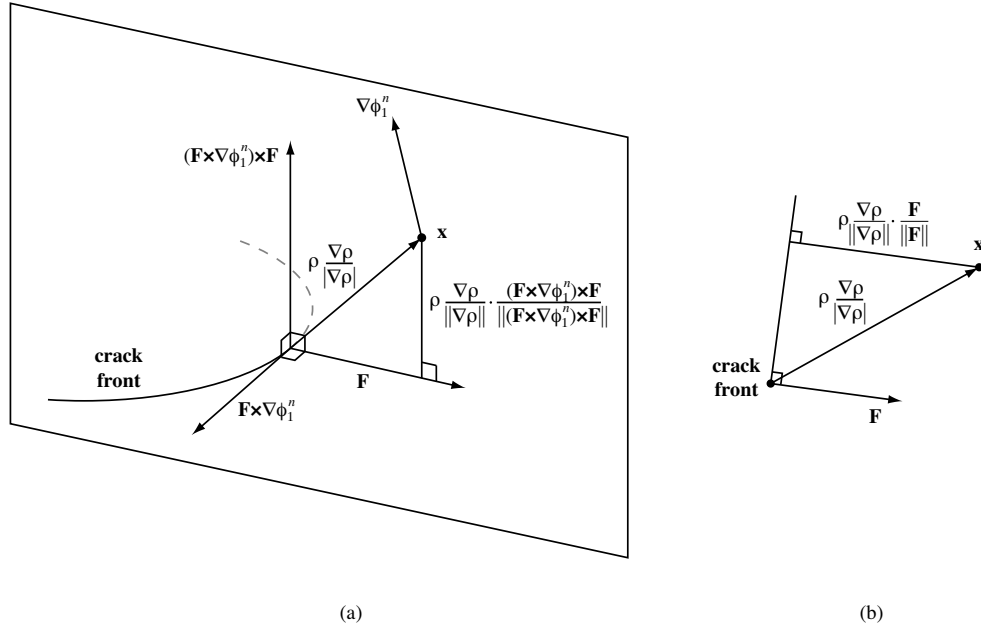
Figure 7. Update formulas for signed distance functions. (a) ϕ_1; and (b) ϕ_2.

where

$$
\mathrm{minmod}(a, b) = \begin{cases} a & |a| \leq |b| \\[2mm] b & |b| < |a| \end{cases} .
\tag{28b}
$$

O: If $\phi_2^n(\mathbf{x}) > 0$ and $\tilde{\phi}_2^{\,n}(\mathbf{x}) \leq 0$, then $\mathbf{x}$ is on the outside of the turn. In this case, the proper distance is simply the distance to the current crack front, which is given by

$$
\phi_1^{n+1}(\mathbf{x}) = \mathrm{sign}(\phi_1^n(\mathbf{x}))\rho(\mathbf{x}).
\tag{29}
$$

3.1.4. Advancing the front Finally, the front has been properly rotated to be normal to the extended crack surface, i.e. $\nabla\tilde{\phi}_2^{\,n} \cdot \nabla\phi_1^{n+1} = 0$, in the region beyond the crack front. The final

step to advance the crack front is to move it according to the magnitude of the velocity for time step Δt, therefore we have

$$\tilde{\phi}_2^{\,n+1}(\mathbf{x}) = \tilde{\phi}_2^{\,n}(\mathbf{x}) - \|\mathbf{F}(\mathbf{x})\|\Delta t. \tag{30}$$

Note, however, that the resulting $\tilde{\phi}_2^{\,n+1}$ is not a signed distance function. If this is not true, then this will cause problems when ϕ_1, ϕ_2 are used as enrichment functions in the X-FEM. Therefore, we must reconstruct the distance function to the crack front by using reinitialization. Reinitialization is a technique often employed in level set methods, and was first introduced by Chopp [40]. However, the reinitialization method we will use is that detailed in Reference [2], which is shown to be much faster and more accurate. In this case, we are computing the distance to the surface $\tilde{\phi}_2^{\,n+1} = 0$, and we use tricubic interpolants and the fast marching method in a manner similar to that detailed earlier, and as described in Reference [2]. Reinitializaton gives us the final $\phi_2^{n+1}(\mathbf{x})$. For good measure, we also reinitialize the function $\phi_1^{n+1}(\mathbf{x})$ even though it should already be a signed distance function by construction.

3.2. Computation of Fracture Parameters

The stress fields at the front of a crack in a homogeneous, isotropic elastic solid have the form:

$$\sigma_{ij} = \frac{K_I}{\sqrt{2\pi r}}\sigma_{ij}^I(\theta) + +\frac{K_{II}}{\sqrt{2\pi r}}\sigma_{ij}^{II}(\theta) + +\frac{K_{III}}{\sqrt{2\pi r}}\sigma_{ij}^{III}(\theta), \tag{31}$$

where δ_{ij} is the Kronecker delta, K_I, K_{II}, K_{III} are the stress intensity factors, and the θ-variations are given in many textbooks, for e.g., see Lawn [41]. Fracture parameters such as the stress intensity factors or the energy release rate are measures of the intensity of the crack front fields. The general crack-tip contour integral at point s along a three-dimensional crack

front is [42]:

$$I(s) = \lim_{\Gamma \to 0} \xi_l(s) \int\limits_{\Gamma(s)} P_{lj} n_j \, d\Gamma, \qquad (32)$$

where $\Gamma(s)$ is a contour that lies in a plane passing through point s and is perpendicular to

the crack front, $\xi_l(s)$ are the components of the local (virtual) crack extension, and n_i are

the components of the unit outward normal to the contour Γ (Fig. 8). A specific form of the

tensor $\mathbf{P}$, namely $P_{lj} = W\delta_{lj} - \sigma_{ij} u_{i,l}$, which is the energy-momentum tensor in continuum

mechanics [43], yields the energy release rate $\mathcal{G}(s)$:

$$\mathcal{G}(s) = \lim_{\Gamma \to 0} c_l(s) \int\limits_{\Gamma(s)} P_{lj} n_j \, d\Gamma, \qquad (33)$$

where $c_l(s)$ is the component of a unit vector that is perpendicular to the crack front and lies

in the local tangent plane to the crack surface at point s. In addition, W is the strain energy

density, σ_{ij} are the components of the Cauchy stress tensor, and u_i are the displacement

components. In general, the jump in the energy-momentum tensor denotes the driving force

acting on a defect (e.g., crack or interface).
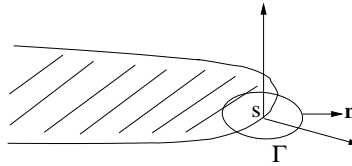
Figure 8. Three-dimensional crack front.

In three dimensions, the plane strain fields are only asymptotically approached at the crack

front, and hence the above contour integral can be used only very near the crack front. It is

precisely near sharp discontinuities (high gradients) such as cracks where the standard finite

element approximation is likely to be the least accurate. To circumvent this difficulty, the

use of domain integrals has become a standard post-processing tool for extracting SIFs in finite element computations [42, 44]. In the domain integral approach, the contour integral is converted to an equivalent domain form that is amenable to numerical computations [45]. In the interaction integral method [45], auxiliary fields (plane strain and anti-plane deformations) are introduced and superposed on the actual fields that arise from the solution of the boundary-value problem. By judicious choice of the auxiliary fields, the interaction integral is directly related to the mixed-mode stress intensity factors; a domain form of the contour interaction integral is used in the extraction of the SIFs. The computation of the SIFs along non-planar crack fronts in three dimensions is presented in Gosz and Moran [46]. In the X-FEM, domain integrals have been adopted for SIF computations in three dimensions for planar mode I cracks [3, 5, 9] as well as for non-planar cracks [7]. In this paper, we use the domain form of the interaction integral presented by Moës *et al.* [7].

3.3. Coupled Extended Finite Element and Fast Marching Method

A tetrahedral finite element mesh is used for the extended finite element analysis, whereas a regular hexahedral grid is used for the fast marching method. In the general coupled numerical method, two signed distance functions are used to maintain the location of the crack front. The signed distance functions from the FMM are used to populate the nodal values in the finite element tetrahedral mesh. Hence, a field description of these functions becomes available (linear interpolation), and the interpolated signed distance functions are used to identify which nodes are required to be enriched and the enrichment functions associated with the crack interior and the crack front are also evaluated using them. Since the crack surface is linear within any tetrahedral element, partitioning the elements for the purpose of numerical integration

is simplified [3, 8]. These steps are used in the extended finite element method to solve the elastostatic boundary-value problem. From the extended finite element solution, the stress intensity factors (K_I, K_{II}, and K_{III}) are extracted at different points s along the crack front, and the maximum hoop stress criterion [47] is invoked to determine the growth direction, $\mathbf{n}_s$, which lies in the tangent plane to the crack front. Assuming a fatigue crack growth law, the crack velocity $\mathbf{F}_s = |\mathcal{G}(s)|\mathbf{n}_s$ [8], where $\mathcal{G}(s)$ is the energy release rate. The time increment, $\Delta t = \ell/|\mathbf{F}_s^{\max}|$, where ℓ is taken to be of the order of the mesh spacing. Let t_{max} denote the terminal time of the evolution. The crack front velocity and the time increment are fed back to the fast marching method to update the location of the crack, and the above steps are repeated. Note that since the finite element mesh and the fast marching grid are distinct, the velocities obtained from X-FEM need to be appropriately mapped onto the fast marching grid (see Section 3.1.2). A summary of the crack growth algorithm using the fast marching method follows:

1. Step $t = 0$ (t_{max} is user-specified). Let ϕ_1 and ϕ_2 be the signed distance functions for the crack surface and the crack front, respectively. The crack surface is contained in the set $\phi_1^{-1}(0)$, and the the crack front is given by $\phi_1^{-1}(0) \bigcap \phi_2^{-1}(0)$, such that $\boldsymbol{\nabla}\phi_1 \cdot \boldsymbol{\nabla}\phi_2 = 0$.

2. Compute the distance function ρ using the FMM with $G = 1$ in Eq. (1).

3. Evaluate $\mathbf{F}$ on the front using the X-FEM, and extend the velocity using the FMM.

4. Modify ϕ_1 in the set $\{\phi_2(\mathbf{x}) > 0\}$ so that it is parallel to $\mathbf{F}$, and modify ϕ_2 so that $\phi_2^{-1}(0)$ is orthogonal to $\mathbf{F}$.

5. Expand the crack using the algorithm described in Section 3.1.

6. if $t < t_{max}$, then increment t ($t \leftarrow t + \Delta t$) and goto step 2.

4. NUMERICAL EXAMPLES

Numerical examples are presented to establish the accuracy of the fast marching algorithm and
to demonstrate the robustness of the X-FEM/FMM coupling for three-dimensional non-planar
crack growth simulations. For the purpose of comparison, our main reference is Lai *et al.* [23],
who developed asymptotic solutions for perturbation of planar and non-planar crack shapes.
In all the crack growth simulations, the problem domain $\Omega = [-0.5, 0.5]^3$. A $50 \times 50 \times 50$
hexahedral fast marching grid is used, whereas the finite element mesh consists of tetrahedral
elements (14K, 17K, and 26K nodes) with nodes that are quasi-uniformly spaced. On average,
the time taken to compute a single crack growth step is between 30 to 45 minutes. Of this, on
the order of a minute is spent to update the signed distance functions using the fast marching
method. Since the number of evaluation points on the crack front ranges from 50 to over 200, a
significant fraction of the total time is spent in the computation of the stress intensity factors
(front velocity) along the crack front. Finite element mesh generation and post-processing is
carried out using the open-source free software `Gmsh` [48].

4.1. Planar Wavy Cracks

Gao and Rice [22] and Lai *et al.* [23] have presented first-order perturbation analysis for a
planar crack that departs from a penny-shaped configuration. Consider a planar crack that is
located on the $x_3 = 0$ plane. The location of the crack front is $a(\theta) = a_0(1 + \epsilon \cos n\theta)$, where
ϵ is the magnitude of the perturbation and n is the wave number. Under uniaxial tension, the
mode I stress intensity factor (SIF) is given by [23]:

$$\frac{K_I}{K_I^0(a(\theta))} = 1 - \frac{\epsilon n}{2} \frac{a_0}{a(\theta)} \cos n\theta, \tag{34}$$
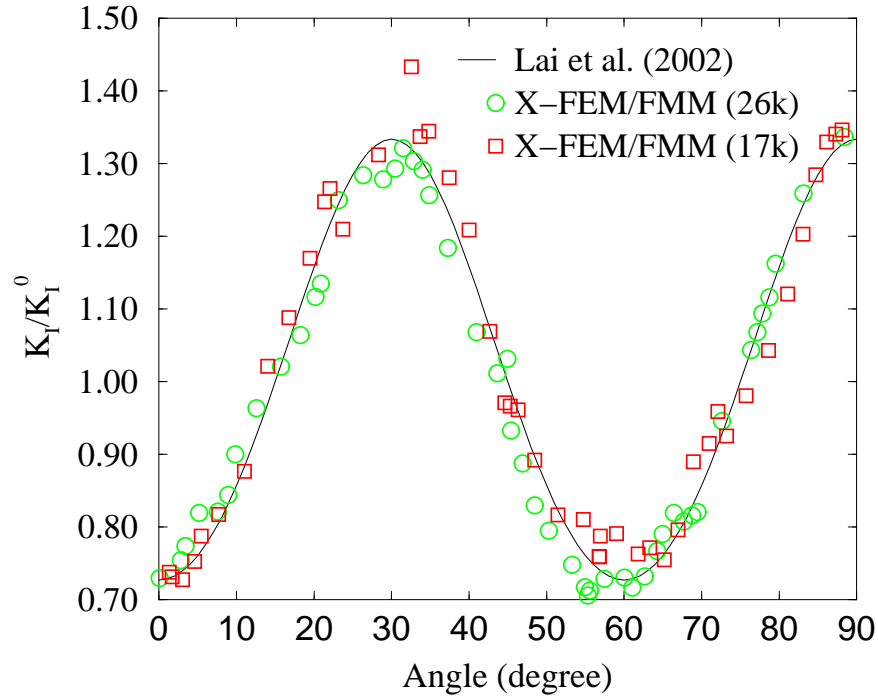
Figure 9. Stress intensity factors for the planar wavy crack problem.

where $K_I^0(a) = 2\sigma_0\sqrt{a/\pi}$ with σ_0 the applied traction. The above is an asymptotic reference solution, which we adopt; the asymptotic solution is within a percent or two of the reference (boundary element) solution [23]. In the numerical computations, we use $a_0 = 0.25$, $n = 6$, and $\epsilon = 0.1$. The analyses are carried out for two meshes with about 17K and 26K nodes. The asymptotic solution and the X-FEM solution is compared in Fig. 9. Results are reported for only one quadrant; the numerical results are proximal to the asymptotic solution for all values of θ. From Fig. 9, we observe that $K_I(\theta)$ varies such that it attains its maximum at $a_{\min} = a_0(1 - \epsilon)$ and its minimum at $a_{\max} = a_0(1 + \epsilon)$. Hence, under quasi-static growth conditions, the crack should attain a circular shape. The 17K node mesh is used in the crack growth analysis, and $\ell = 0.015$ is used to determine Δt. In Fig. 10, the crack locations at three

different steps ($t = 0$, $5\Delta t$, $10\Delta t$) are depicted. In accordance with theory, the wavy crack grows to become circular in shape.

4.2. Testing the FMM Algorithm

To assess the robustness and accuracy of the fast marching algorithm, a manufactured solution for an ordinary differential equation is used. In this test, a simple exponential growth/decay ordinary differential equation is built to go in the radial direction with initial data $x_3 = 0$ at radius $r = \sqrt{x_1^2 + x_2^2} = 1$. For this basic construction, the corresponding front velocity, $\mathbf{V}(\mathbf{x})$, is given by

$$\mathbf{V}(\mathbf{x}) = \frac{x_1\mathbf{e}_1 + x_2\mathbf{e}_2 + x_1(1 + x_3)\mathbf{e}_3}{\sqrt{x_1^2 + x_2^2}}, \tag{35}$$

and the exact solution for the surface is given by

$$x_3(x_1, x_2) = -1 + \exp\left(x_1\left[1 - \frac{1}{\sqrt{x_1^2 + x_2^2}} \right] \right). \tag{36}$$

A $100 \times 100 \times 100$ fast marching grid is used, and simulations are performed for 10 time steps. In Fig. 11, the exact and FMM solutions are compared, and the FMM solution is observed to be in good agreement with the exact solution inside the computed crack surface. The deviation outside the location of the crack front surface $\phi_2 = 0$ shows the difference between the exact solution and the linear extrapolation used when constructing ϕ_1 away from the crack surface.

Occasionally, points that are in region O or I as depicted in Fig. 6 may generate a sign error. These errors always occur far from the crack surface and are a result of the long extrapolation distances being calculated. These points are not of concern because they remain far from the crack surface, $\phi_1 = 0$, and hence do not affect the location of the crack or the crack front velocity.
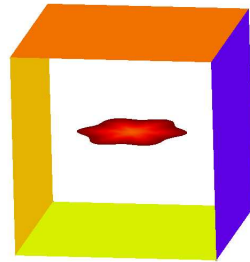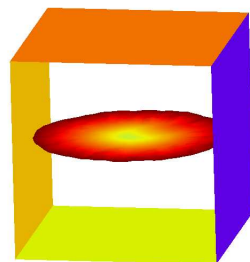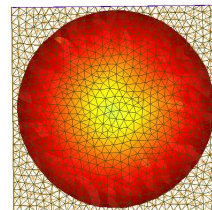
(a)
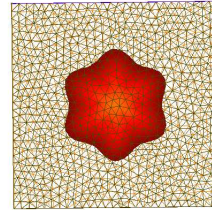
(b)
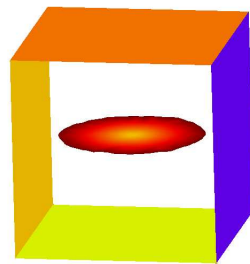
(c)
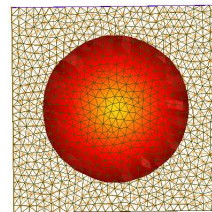
(d)

(e)

(f)

Figure 10. Propagation of a planar wavy crack. (a),(b) $t = 0$; (c),(d) $t = 5\Delta t$; and (e),(f) $t = 10\Delta t$.

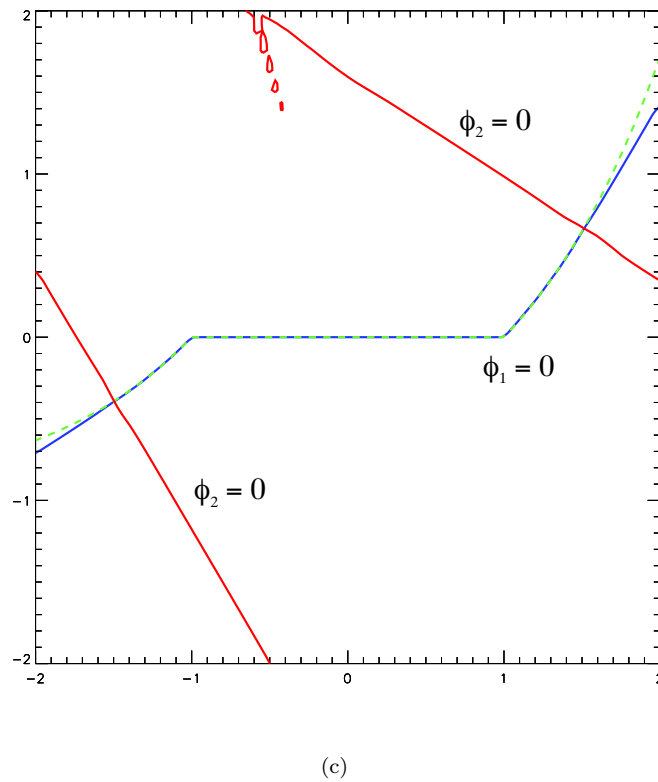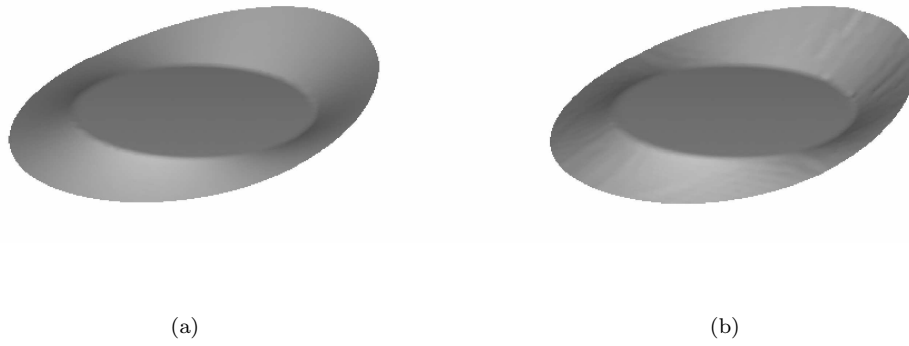(a)   (b)

(c)

Figure 11. Benchmarking the fast marching algorithm. (a),(b) Exact surface and surface obtained
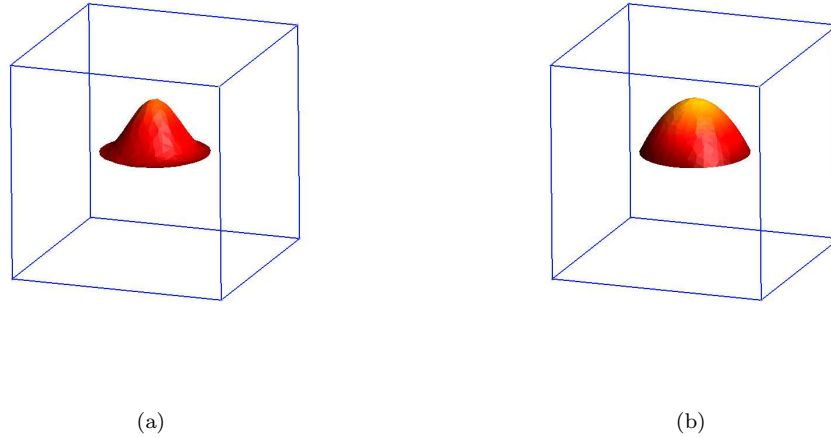using the FMM algorithm; and (c) Comparison of exact solution and FMM along a cross-section.

<center>(a)   (b)</center>

Figure 12. Axisymmetric non-planar cracks. (a) Regular perturbation; and (b) Singular (cap-shaped crack) perturbation.

4.3. Non-Planar Axisymmetric Cracks

We consider two particular axisymmetric crack shapes. The cracks are defined by [23]

$$\frac{h}{a} = \frac{1}{2}\left\{\cos\left(\frac{\pi r}{a}\right) + 1\right\}, \tag{37a}$$

$$\frac{h}{a} = 1 - \left(\frac{r}{a}\right)^2, \tag{37b}$$

where the first crack is a regular perturbation and the latter is a singular (cap-shaped crack) perturbation. The initial level sets for these cracks are shown in Fig. 12. Crack propagation simulations are conducted for the cap-shaped crack under uniaxial tension. A 14K node mesh is used. In Fig. 13, the crack shapes at $t = 0$, $5\Delta t$, $9\Delta t$ are shown. Even though the growth is planar, it is noteworthy to observe that the wake of the crack is accurately captured. This demonstrates that the signed distance functions are not changed in the wake of the crack, as described in Section 3.1.
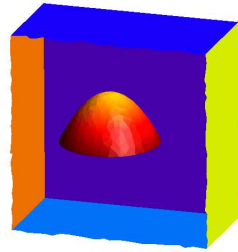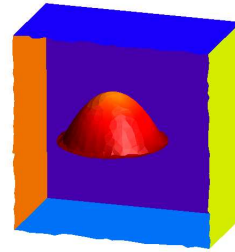
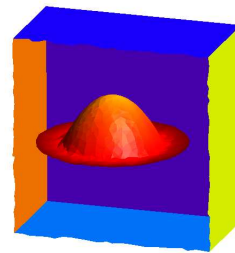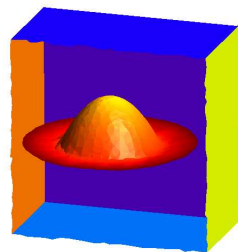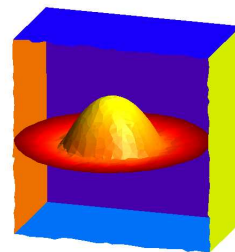(a)   (b)
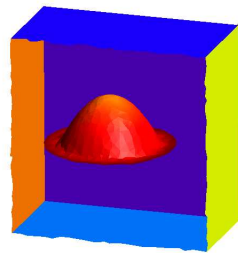
(c)   (d)

(e)   (f)

Figure 13. Propagation of a cap-shaped crack. (a) $t = 0$; (b) $t = \Delta t$; (c) $t = 3\Delta t$; (d) $t = 5\Delta t$; (e)

$t = 7\Delta t$; and (f) $t = 9\Delta t$.

4.4. Non-Planar Growth of an Inclined Penny Crack

As the last example, we consider an inclined penny crack ($\theta = 45°$) that is under uniaxial

tension. Since the loading at the crack front is mixed-mode, the crack kinks. For these

simulations, the 17K node mesh is used; Δt is computed using $\ell = 0.015$. Crack propagation

simulations are conducted, and the different stages ($t = 0$, $4\Delta t$, $8\Delta t$, $12\Delta t$, $16\Delta t$, $20\Delta t$) in

the crack growth are depicted in Fig. 14. The simulations reveal the non-planar character of

the growth and the results evince that the fast marching algorithm is able to capture the

representation of the crack at each step.

5. CONCLUDING REMARKS

In this paper, a fast marching algorithm for crack propagation simulations was proposed.

Unlike a level set implementation for non-planar cracks [8], the new method is a single-pass

algorithm (no iterations are required) with no time step restrictions. The fast marching method

was coupled to the extended finite element for quasi-static three-dimensional crack growth

simulations. In the X-FEM, cracks are represented by enriching the standard finite element

displacement approximation through the framework of partition of unity. A discontinuous

function and the asymptotic crack-tip fields were used as enrichment functions to model a

crack. The fast marching method was coupled to the X-FEM to simulate crack propagation

without the need for remeshing. Two distinct meshes were used for the X-FEM and FMM.

An unstructured tetrahedral mesh with linear C^0 finite elements was used for the extended

finite element analysis. A structured mesh with local C^1 tricubic interpolation was adopted

for the fast marching method; development of a fast marching crack growth algorithm on
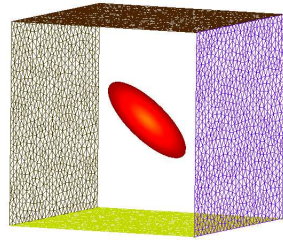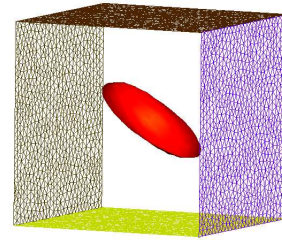
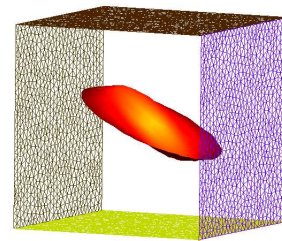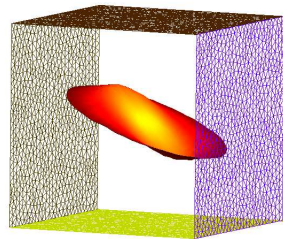(a)   (b)
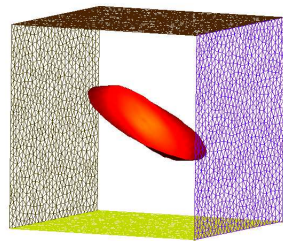
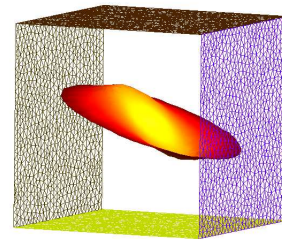(c)   (d)

(e)   (f)

Figure 14. Non-planar propagation of an inclined penny crack. (a) $t = 0$; (b) $t = 4\Delta t$; (c) $t = 8\Delta t$;

(d) $t = 12\Delta t$; (e) $t = 16\Delta t$; and (f) $t = 20\Delta t$.

unstructured meshes is part of on-going research. Two signed distance functions, which were used to represent the crack, were also used to compute the enrichment functions. The crack front velocity was computed using the X-FEM and fed to the fast marching method. To extend the crack, algorithms to update the signed distance functions were presented.

The principal advantages offered by coupling these methods are the geometric and topological flexibility of the level set method, the speed of the fast marching method, and the greater subgrid resolution and singularity capturing of the extended finite element method. The resulting coupled method runs entirely on a fixed Eulerian mesh without the need for conformity to the crack front, while achieving much greater accuracy in the neighborhood of the front with minimal additional cost. In the analyses, benchmark planar and non-planar crack solutions from Lai *et al.* [23] were considered. The numerical simulations that were presented revealed the accuracy and robustness of the new fast marching algorithm, and its efficient coupling to the X-FEM for planar and non-planar crack propagation problems.

6. ACKNOWLEDGMENTS

REFERENCES

1. J. A. Sethian. Fast marching methods. *SIAM Review*, 41(2):199–235, 1999.

2. D. L. Chopp. Some improvements of the fast marching method. *SIAM Journal of Scientific Computing*, 23(1):230–244, 2001.

3. N. Sukumar, N. Moës, B. Moran, and T. Belytschko. Extended finite element method for three-dimensional crack modelling. *International Journal for Numerical Methods in Engineering*, 48(11):1549–1570, 2000.

4. N. Sukumar, D. L. Chopp, and B. Moran. Extended finite element method and fast marching method for three dimensional fatigue crack propagation. unpublished, 2000.

5. N. Sukumar, D. L. Chopp, and B. Moran. Extended finite element method and fast marching method for three dimensional fatigue crack propagation. *Engineering Fracture Mechanics*, 70(1):29–48, 2003.

6. M. Stolarska, D. L. Chopp, N. Möes, and T. Belytschko. Modelling crack growth by level sets in the extended finite element method. *International Journal for Numerical Methods in Engineering*, 51(8):943–960, 2001.

7. N. Moës, A. Gravouil, and T. Belytschko. Non-planar 3D crack growth by the extended finite element and level sets. Part I: Mechanical model. *International Journal for Numerical Methods in Engineering*, 53(11):2549–2568, 2002.

8. A. Gravouil, N. Moës, and T. Belytschko. Non-planar 3D crack growth by the extended finite element and the level sets. Part II: Level set update. *International Journal for Numerical Methods in Engineering*, 53(11):2569–2586, 2002.

9. D. L. Chopp and N. Sukumar. Fatigue crack propagation of multiple coplanar cracks with the coupled extended finite element/fast marching method. *International Journal of Engineering Science*, 41(8):845–869, 2003.

10. B. Prabel, A. Combescure, A. Gravouil, and S. Marie. Level set X-FEM non-matching meshes: Application to dynamic crack propagation in elastic-plastic media. *International Journal for Numerical Methods in Engineering*, 69(8):1553–1569, 2007.

11. M. Duflot. A study on the representation of cracks with level sets. *International Journal for Numerical Methods in Engineering*, 70(11):1261–1302, 2007.

12. G. Dhondt. Automatic 3-D mode I crack propagation calculations with finite elements. *International Journal for Numerical Methods in Engineering*, 41(4):739–757, 1998.

13. B. J. Carter, P. A. Wawrzynek, and A. R. Ingraffea. Automated 3-d crack growth simulation. *International Journal for Numerical Methods in Engineering*, 47:229–253, 2000.

14. Y. Mi and M. H. Aliabadi. Three-dimensional crack growth simulations using BEM. *Computers and Structures*, 52(5):871–878, 1994.

15. S. C. Forth and W. D. Keat. Three dimensional nonplanar fracture model using the surface integral method. *International Journal of Fracture*, 77:243–262, 1996.

16. M. Bonnet, G. Maier, and G. Polizzotto. Symmetric Galerkin boundary element method. *Applied Mechanics Review*, 51:669–704, 1998.

17. G. P. Nikishkov and S. N. Atluri. SGBEM-FEM alternating method for analyzing 3D non-planar cracks and their growth in structural components. *CMES-Computer Modeling in Engineering & Sciences*, 2(3):401–422, 2001.

18. K. Yoshida, N. Nishimura, and S. Kobayashi. Application of new fast multipole boundary integral equation method to crack problems in 3D. *Engineering Analysis with Boundary Elements*, 25(4–5):239–247, 2001.

19. K. Kolk, W. Weber, and G. Kuhn. Investigation of 3D crack propagation problems via fast BEM formulations. *Computational Mechanics*, 37(1):32–40, 2005.

20. G. Xu and M. Ortiz. A variational boundary integral equation method for the analysis of 3d cracks of arbitrary geometry modelled as continuous distribution of dislocation loops. *International Journal for Numerical Methods in Engineering*, 31:3675–3701, 1993.

21. G. Xu, A. F. Bower, and M. Ortiz. An analysis of non-planar crack growth under mixed mode loading. *International Journal of Solids and Structures*, 31(16):2167–2193, 1994.

22. H. Gao and J. R. Rice. Somewhat circular tensile cracks. *International Journal of Fracture*, 33(3):155–174, 1987.

23. Y.-S. Lai, A. B. Movchan, and G. J. Rodin. A study of quasi-circular cracks. *International Journal of Fracture*, 113:1–25, 2002.

24. V. Lazarus. Brittle fracture and fatigue propagation paths of 3D plane cracks under uniform remote tensile loading. *International Journal of Fracture*, 122(1–2):23–46, 2003.

25. E. Favier, V. Lazarus, and J.-B. Leblond. Coplanar propagation paths of 3D cracks in infinite bodies loaded in shear. *International Journal of Solids and Structures*, 43(7–8):2091–2109, 2006.

26. V. Lazarus. Mixed mode stress intensity factors for deflected and inclined corner cracks in finite-thickness plates. *International Journal of Fatigue*, 29(2):305–317, 2007.

27. P. M. A. Areias and T. Belytschko. Analysis of three-dimensional crack initiation and propagation using the extended finite element method. *International Journal for Numerical Methods in Engineering*, 63(5):760–788, 2005.

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

29. T. C. Gasser and G. A. Holzapfel. 3D crack propagation in unreinforced concrete. A two-step algorithm for tracking 3D crack paths. *Computer Methods in Applied Mechanics and Engineering*, 195(37–40):5198–

5219, 2006.

30. J. Mergheim, E. Kuhl, and P. Steinmann. Towards the algorithmic treatment of 3d strong discontinuities. *Communications in Numerical Methods in Engineering*, 23(2):97–108, 2007.

31. T. Rabczuk and T. Belytschko. A three-dimensional large deformation meshfree method for arbitrary evolving cracks. *Computer Methods in Applied Mechanics and Engineering*, 196(29-30):2777–2799, 2007.

32. D. L. Chopp. Another look at velocity extensions in the level set method. in review, 2007.

33. S. Osher and J. A. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics*, 79(1):12–49, November 1988.

34. D. Adalsteinsson and J. A. Sethian. The fast construction of extension velocities in level set methods. *Journal of Computational Physics*, 48(1):2–22, 1999.

35. 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.

36. I. Babuška and I. Melenk. Partition of unity method. *International Journal for Numerical Methods in Engineering*, 40:727–758, 1997.

37. N. Möes, 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.

38. T. Strouboulis, K. Copps, and I. Babuška. The generalized finite element method. *Computer Methods in Applied Mechanics and Engineering*, 190(32–33):4081–4193, 2001.

39. T. Belytschko and T. Black. Elastic crack growth in finite elements with minimal remeshing. *International Journal for Numerical Methods in Engineering*, 45(5):601–620, 1999.

40. D. L. Chopp. Computing minimal surfaces via level set curvature flow. *Journal of Computational Physics*, 106(1):77–91, May 1993.

41. B. Lawn. *Fracture of Brittle Solids*. Cambridge University Press, Cambridge, England, second edition, 1993.

42. B. Moran and C. F. Shih. Crack tip and associated domain integrals from momentum and energy balance. *Engineering Fracture Mechanics*, 27(6):615–641, 1987.

43. J. D. Eshelby. Energy relations and the energy momentum tensor in continuum mechanics. In M. F. Kanninen, W. F. Adler, A. R. Rosenfeld, and R. T. Jafee, editors, *Inelastic Behavior of Solids*, pages 77–114, New York, 1970. McGraw-Hill, Inc.

44. G. P. Nikishkov and S. N. Atluri. Calculation of fracture mechanics parameters for an arbitrary three-dimensional crack by the 'equivalent domain integral method'. *International Journal for Numerical*

Methods in Engineering, 24:1801–1821, 1987.

45. C. F. Shih, B. Moran, and T. Nakamura. Energy release rate along a three-dimensional crack front in a thermally stressed body. *International Journal of Fracture*, 30:79–102, 1986.

46. M. Gosz and B. Moran. An interaction energy integral method for computation of mixed-mode stress intensity factors along non-planar crack fronts in three dimensions. *Engineering Fracture Mechanics*, 69(3):299–319, 2002.

47. F. Erdogan and G. C. Sih. On the crack extension in plates under plane loading and transverse shear. *Journal of Basic Engineering*, 85:519–527, 1963.

48. C. Geuzaine and J.-F. Remacle. *Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities*. Available at `http://www.geuz.org/gmsh`, 2007.