INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2014; **100**:555–576
Published online 11 August 2014 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/nme.4744

Reduction in mesh bias for dynamic fracture using adaptive splitting of polygonal finite elements

S. E. Leon‡, D. W. Spring‡ and G. H. Paulino*,†

Department of Civil and Environment Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA

SUMMARY

We present a method to reduce mesh bias in dynamic fracture simulations using the finite element method with adaptive insertion of extrinsic cohesive zone elements along element boundaries. The geometry of the domain discretization is important in this setting because cracks are only allowed to propagate along element facets and can potentially bias the crack paths. To reduce mesh bias, we consider unstructured polygonal finite elements in this work. The meshes are generated with centroidal Voronoi tessellations to ensure element quality. However, the possible crack directions at each node are limited, making this discretization a poor candidate for dynamic fracture simulation. To overcome this problem, and significantly improve crack patterns, we propose adaptive element splitting, whereby the number of potential crack directions is increased at each crack tip. *Thus, the crack is allowed to propagate through the polygonal element.* Geometric studies illustrate the benefits of polygonal element discretizations employed with element splitting over other structured and unstructured discretizations for crack propagation applications. Numerical examples are performed and demonstrate good agreement with previous experimental and numerical results in the literature. Copyright © 2014 John Wiley & Sons, Ltd.

Received 8 July 2013; Revised 5 March 2014; Accepted 21 June 2014

KEY WORDS: polygonal finite elements; cohesive fracture; mesh dependency; element splitting; random meshes

1. INTRODUCTION

In numerical dynamic fracture simulations, researchers aim to predict real world failure phenomenon, especially for cases where experiments would not be feasible because of the size or cost of the experimental set up. A simulation ought to produce physically meaningful results, without bias or influence from the numerical method itself. In this work, we employ the finite element method (FEM) and inter-element cohesive zone modeling to simulate dynamic fracture. These methods, however, are known to suffer from mesh bias, as cracks are forced to propagate along element boundaries [1]. To greatly reduce mesh bias, we use polygonal finite elements and propose an element splitting technique to improve crack patterns.

The concept of restricting the cracks to element boundaries was studied by Camacho and Ortiz [2], who proposed an adaptive strategy to insert cohesive elements at the facets where softening occurs. By limiting the location of crack propagation to the facets, issues of crack branching, crack coalescence, and fragmentation are naturally handled by the mesh topology. The primary drawback of this method is that the initial mesh, if chosen poorly, can adversely influence the path(s) of crack propagation. Several meshing strategies have been proposed to circumvent this issue.

*Correspondence to: G. H. Paulino, Department of Civil and Environment Engineering, University of Illinois at Urbana-Champaign, 205 N. Matthews Ave. Urbana, IL 61801, USA.
†E-mail: paulino@illinois.edu
‡Equally contributing authors.

Unstructured meshes are preferred for fracture applications because they possess no preferential path directions (i.e., isotropic), which can result in more realistic crack patterns than their structured counterparts [3, 4]. Methods to create arbitrary meshes for crack propagation problems have been studied extensively. Ingraffea, Wawrzynek, and coworkers, for example, have developed an algorithm to generate meshes on arbitrary domains for fracture applications [5] and have recently presented a parallel technique for mesh generation [6]. However, structured meshes have their benefits too: they can be systematically generated and may posses a hierarchical subdivision, making them suitable for refinement schemes. Hence, researchers have used structured meshes with special properties or additional adaptivity operators in an attempt to gain the benefits of a structured mesh with the isotropy of an unstructured mesh. The pinwheel mesh possesses the isoperimetric property, meaning that, as the element sizes tend to zero, it is able to represent any arbitrary crack path [7, 8]. However, applications of the pinwheel mesh are limited because the convergence of crack lengths and angles exists only in the limit sense and may not be achieved with practical ranges of mesh size. Furthermore, construction of the pinwheel mesh is not straightforward. The 4k mesh with adaptivity operators proposed in [1] is a practical alternative to the pinwheel mesh. Hence, we will use it for comparison with the present approach.

The structured 4k mesh has been widely used in finite element applications. It can easily be generated, even for arbitrary domains [9], and its structure and multi-level representation have made it a popular candidate for mesh refinement and coarsening strategies [9, 10]. A weakness of the structured 4k mesh is that it suffers from an inherent anisotropy and introduces a significant error in the crack length for angles other than those at 45° increments. Recently, nodal perturbation and edge-swap operators have been proposed to unstructure the geometry and topology of the 4k mesh, respectively, and to reduce the error in crack path convergence [1]. These operators significantly reduce the error in the crack path; however, the mesh still introduces unwarranted anisotropies into the problem domain [11, 12].

The unstructured, conjugate-directions mesh is presented by Rimoli and coworkers [11, 12]. They use a barycentric subdivision of the k-means mesh to construct a new mesh that has the same isotropic property of the k-means mesh, but has more potential crack directions. The scheme proposed in the present study aims to achieve similar results to that of the conjugate-directions mesh in geometric studies; however, because the element splitting is performed adaptively (i.e., as needed), the computational storage costs are less than what they would be to store an entire conjugate-directions mesh. Additionally, unlike the conjugate-directions mesh, the method of splitting the elements in the proposed approach is controlled to limit the presence of elements with small edges, which may not be well suited for finite element applications.

Alternatively, unstructured, isotropic meshes may be achieved by the use of polygonal elements constructed by means of Voronoi tessellations. Many researchers have used polygonal discretizations for fracture applications. Li and Ghosh model multiple crack growth with the extended Voronoi cell FEM [13] in brittle materials [14, 15]. Ooi *et al.* [16–18] develop a hybrid finite element-scaled boundary finite element technique to propagate cracks. For a more detailed discussion of Voronoi tessellations applied to fracture problems, we direct the reader to [19]. Bishop and coworkers use a framework similar to that studied in this work to model pervasive fracture phenomena. The finite element meshes are generated with randomly close-packed Voronoi tessellations, and cohesive elements are adaptively inserted at element boundaries [20, 21]. Other than the dynamic insertion of cohesive elements, however, the mesh remains fixed throughout the simulations. Therefore, to obtain more realistic crack patterns in the present study, the mesh is adaptively modified when needed.

The remainder of the paper is organized as follows. The numerical framework for our study is outlined in Section 2. In Section 3, we present the element splitting algorithm and discuss the path convergence characteristics of the polygonal mesh compared to the 4k mesh. Numerical examples are discussed in Section 4. Finally, in Section 5, we present our conclusions and extensions for future work.

DOI: 10.1002/nme

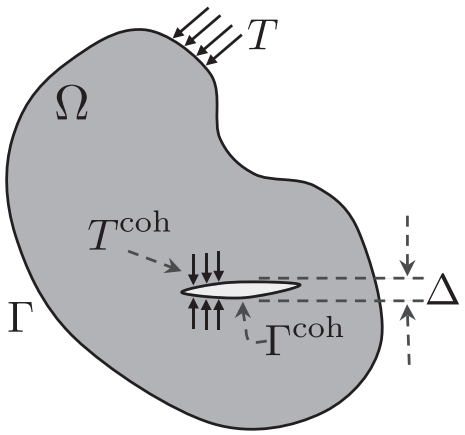

Figure 1. Arbitrary domain with applied boundary conditions.

2. MODELING FRAMEWORK

This work uses arbitrary polygonal finite elements and inter-element cohesive zone elements to perform dynamic fracture simulations. In this section, we present an overview of the finite element formulation for dynamic fracture problems with polygonal elements, the cohesive zone model to govern the separation of surfaces, and the topological data structure to manage on-the-fly mesh modification.

2.1. Finite element formulation

Consider the case of an arbitrary domain, Ω, that is subjected to surface tractions, $\boldsymbol{T}$, along the boundary, Γ, and cohesive tractions, $\boldsymbol{T}^{\text{coh}}$, along the fractured surfaces, Γ^{coh}, illustrated in Figure 1. The equations of motion are derived using the principle of virtual work, which states

$$\int_{\Omega} \delta\mathbf{u}^T \rho\ddot{\mathbf{u}} + \delta\boldsymbol{\varepsilon}^T \boldsymbol{\sigma}\, d\Omega = \int_{\Gamma} \delta\mathbf{u}^T \mathbf{T} d\Gamma + \int_{\Gamma^{\text{coh}}} \delta\boldsymbol{\Delta}^T \boldsymbol{T}^{\text{coh}} d\Gamma \tag{1}$$

where ρ is the mass density, $\boldsymbol{\sigma}$ is the stress, $\mathbf{u}$ is the displacement, $\ddot{\mathbf{u}}$ is the second time derivative of the displacement (i.e., acceleration), $\delta\boldsymbol{\Delta}$ is the virtual separation across the fractured surfaces, $\delta\mathbf{u}$ is the virtual displacement, and $\delta\boldsymbol{\varepsilon}$ is the virtual strain. The stress and strain measures used in this work are the 2nd Piola–Kirchhoff stress tensor and the Green–Lagrange strain tensor, respectively. The effects of body forces and damping are neglected. The continuous problem is converted to a discrete problem via the standard Galerkin approximation. Finite deformations are taken into account by means of the total Lagrangian formulation, in which all quantities are measured with respect to the initial configuration [22].

2.1.1. Mesh generation. The domain, Ω, is discretized into a collection of arbitrary convex polygonal elements. All of the meshes presented in this work are constructed using PolyMesher [23], a MATLAB code for general-purpose polygonal mesh generation on arbitrary domains. We employ the centroidal Voronoi tessellation (CVT) method, whereby the generating point of the Voronoi cell is located at the center of mass of the cell.

The CVT results in a discretization in which there is little variation in the range of element sizes, when compared to a polygonal mesh generated using randomly placed seeds. We refer to a mesh with more uniform edge lengths as higher quality than one with a large variation in edge length, in the sense that they are better suited for finite element analysis applications [23] (see [24] for a statistical analysis of CVT quality). We also investigated randomly close packing the element generating seeds, but found little difference in results when compared to the CVT (Section 3.2).

DOI: 10.1002/nme

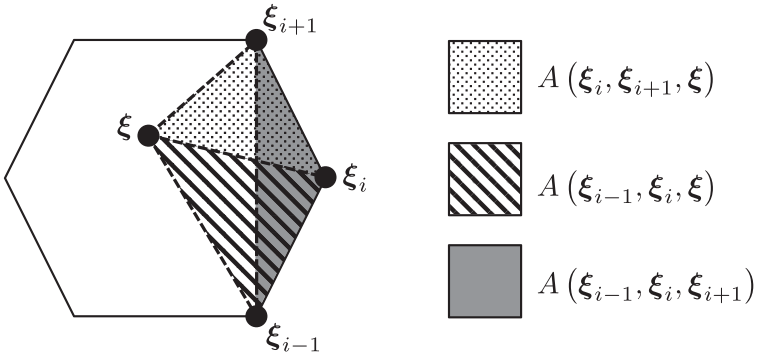

Figure 2. Triangular areas used to define α_i.

2.1.2. Shape functions for polygonal elements. By the virtue of the Galerkin approximation, the same functions are used for the test and trial functions. Field quantities are determined at any point in the domain via interpolation from nodal quantities. The separation across the cohesive elements is interpolated using linearly varying shape functions. On the volumetric elements, we use Wachspress shape functions [25] that satisfy the desirable properties of shape functions [26]. Isoparametric mapping is employed to map quantities from a regular n-gon in the reference coordinate system to the arbitrary convex polygon in the physical coordinate system. The Wachspress shape functions [26] for a reference n-gon are expressed as

$$N_i\left(\boldsymbol{\xi}\right) = \frac{\alpha_i\left(\boldsymbol{\xi}\right)}{\sum_{j=1}^{n} \alpha_j\left(\boldsymbol{\xi}\right)} \tag{2}$$

where the interpolants α_i are of the form

$$\alpha_i\left(\xi\right) = \frac{A\left(\boldsymbol{\xi}_{i-1}, \boldsymbol{\xi}_i, \boldsymbol{\xi}_{i+1}\right)}{A\left(\boldsymbol{\xi}_{i-1}, \boldsymbol{\xi}_i, \boldsymbol{\xi}\right) A\left(\boldsymbol{\xi}_i, \boldsymbol{\xi}_{i+1}, \boldsymbol{\xi}\right)} \tag{3}$$

and A denotes the area of the triangle made up of the points in its arguments. For example, Figure 2 shows a reference hexagon with shaded triangular regions to indicate the areas used in the computation of α_i in the shape function of the node at $\boldsymbol{\xi}_i$ evaluated at point $\boldsymbol{\xi}$. For a detailed discussion of the construction and implementation of the Wachspress shape functions for finite element applications, the reader is directed to [27]. The Wachspress shape functions are generated once, before the fracture simulation begins, and the necessary quantities (i.e., shape functions and their derivatives evaluated at the integration points) are stored and accessed on an as-needed basis.

2.1.3. Numerical integration. After discretization of Equation (1), integration is performed via numerical quadrature, for which there are many schemes available. One scheme is to use quadrature rules that have been designed specifically for polygonal domains [28, 29]. In this work, we opt to subdivide the reference n-gon into n non-overlapping triangles meeting at the centroid of the polygon. We then use a 3-point quadrature rule on each triangle. The contribution from each triangle is summed up to obtain the integral over the entire element. Alternatively, the reference n-gon could be subdivided into quadrilaterals and a 2×2 quadrature rule used on each quadrilateral.

To achieve a continuous stress field across the domain (except across cracks), we first extrapolate stresses from the quadrature points to the nodes then conduct a least squares minimization over each element [30]. Finally, the nodal stress value is computed as the average of that node's stress contribution from each adjacent element.

To evaluate the dynamic response of a system, we use an explicit time integration scheme: the well known explicit central difference method, which is a subset of the classical Newmark method [31]. The mass matrix is diagonal in order for the central difference method to be explicit; hence we apply a standard mass lumping technique in which the diagonals of the consistent mass matrix are calculated and scaled [32, 33].

DOI: 10.1002/nme

2.2. Park–Paulino–Roesler potential-based extrinsic cohesive zone model

In the dynamic simulations performed in this work, the creation of fractured faces, that is, crack advancement, is achieved by means of cohesive zone elements, which are inserted at facets between finite elements. The cohesive zone elements represent the inelastic zone in front of the crack tip. The finite element formulation of the cohesive elements (i.e., construction of the element force vector) is not described here, instead the readers are referred to the following [34–36]. Either an intrinsic or extrinsic approach can be employed. In the former, the cohesive zone elements are inserted into the domain *a priori*, and their activation criterion is internal to the formulation. This approach is known to produce artificial compliance in the system and can lead to unrealistic results in certain settings [37]. Therefore, we use the extrinsic approach, in which the criterion to insert the traction-separation relationship is external to the formulation. A cohesive element is inserted once the averaged normal or tangential stress along an edge exceeds the normal or tangential cohesive strength, respectively. Once the extrinsic cohesive element is inserted, its behavior is governed by the traction-separation relationship of the extrinsic Park–Paulino–Roesler (PPR) cohesive model [38]. Next, we describe the PPR softening model, and also comment on the unloading/reloading and contact formulations used in this work.

2.2.1. Park–Paulino–Roesler softening model. The PPR cohesive model is chosen to represent the nonlinear softening behavior in the inelastic zone in front of the crack tip [38]. The PPR cohesive model is potential based, meaning that the traction-separation relationship is derived from a potential and the unloading relationship and contact formulation are independent of the model. The potential is given by

$$\Psi(\Delta_n, \Delta_t) = \min(\phi_n, \phi_t) + \left[\Gamma_n \left(1 - \frac{\Delta_n}{\delta_n}\right)^{\alpha} + \langle \phi_n - \phi_t \rangle\right]\left[\Gamma_t \left(1 - \frac{|\Delta_t|}{\delta_t}\right)^{\beta} + \langle \phi_t - \phi_n \rangle\right] \tag{4}$$

where the Macaulay bracket $\langle \cdot \rangle$ is defined as

$$\langle x \rangle = \begin{cases} 0 & x < 0 \\ x & x \geq 0 \end{cases} \tag{5}$$

The normal, T_n, and tangential, T_t, tractions are determined by taking the derivative of the potential with respect to the normal opening, Δ_n, and tangential opening, Δ_t, respectively. Hence,

$$T_n(\Delta_n, \Delta_t) = -\alpha \frac{\Gamma_n}{\delta_n}\left(1 - \frac{\Delta_n}{\delta_n}\right)^{\alpha-1}\left[\Gamma_t \left(1 - \frac{|\Delta_t|}{\delta_t}\right)^{\beta} + \langle \phi_t - \phi_n \rangle\right] \tag{6}$$

$$T_t(\Delta_n, \Delta_t) = -\beta \frac{\Gamma_t}{\delta_t}\left(1 - \frac{|\Delta_t|}{\delta_t}\right)^{\beta-1}\left[\Gamma_n \left(1 - \frac{\Delta_n}{\delta_n}\right)^{\alpha} + \langle \phi_n - \phi_t \rangle\right]\frac{\Delta_t}{|\Delta_t|} \tag{7}$$

where ϕ_n and ϕ_t are mode I and mode II fracture energies, and α and β are mode I and mode II shape parameters, respectively. The energy constants Γ_n and Γ_t are defined as

$$\Gamma_n = (-\phi_n)^{\langle \phi_n - \phi_t \rangle/(\phi_n - \phi_t)}, \qquad \Gamma_t = (-\phi_t)^{\langle \phi_t - \phi_n \rangle/(\phi_t - \phi_n)} \qquad (\phi_n \neq \phi_t) \tag{8}$$

for different fracture energies and as

$$\Gamma_n = -\phi_n, \qquad \Gamma_t = 1 \qquad (\phi_n = \phi_t) \tag{9}$$

if the fracture energies are the same for both mode I and mode II separation.

The cohesive tractions are defined within a rectangular domain of dependence that is defined by final crack opening widths and conjugate final crack opening widths, as illustrated in Figure 3. The normal and tangential final crack opening widths are

$$\delta_n = \alpha \phi_n / \sigma_{\max}, \qquad \delta_t = \beta \phi_t / \tau_{\max} \tag{10}$$

DOI: 10.1002/nme

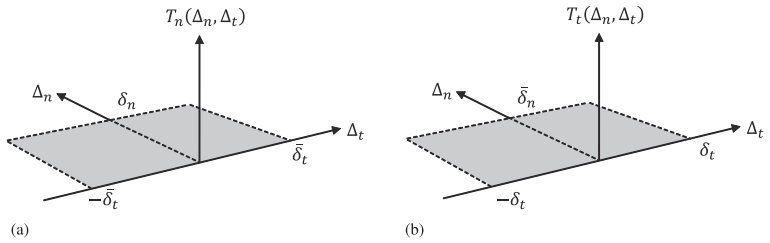

Figure 3. Domain of dependance for the (a) normal T_n and (b) tangential T_t interactions: defined by the final crack opening widths (δ_n, δ_t) and the conjugate final crack opening widths ($\bar{\delta}_n$,$\bar{\delta}_t$).

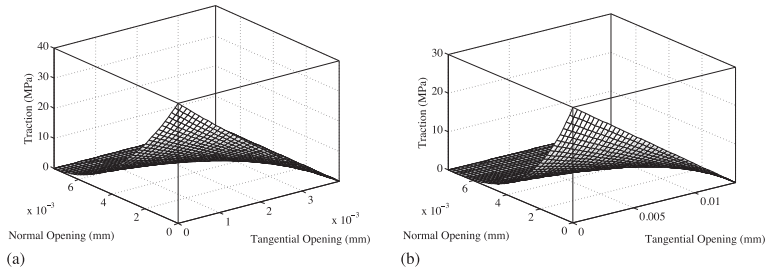

Figure 4. Traction-separation relations for (a) normal opening ($\phi_n = 100N/m, \sigma_{max} = 40MPa, \alpha = 3.0$) and (b) tangential opening ($\phi_t = 200N/m, \tau_{max} = 30MPa, \beta = 5.0$).

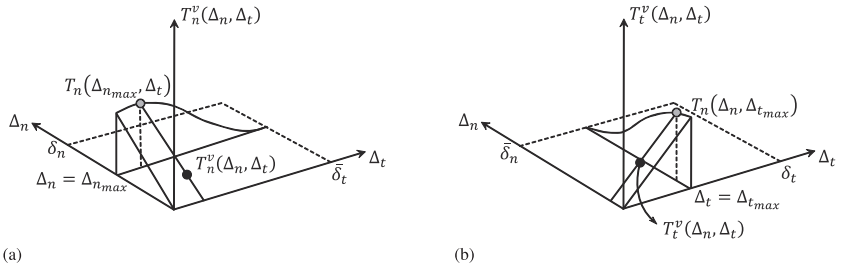

Figure 5. Uncoupled unloading schemes for (a) normal and (b) tangential interactions [39].

respectively, and the normal and tangential conjugate final crack opening widths are

$$\bar{\delta}_n = \delta_n - \delta_n \left(\frac{\langle \phi_n - \phi_t \rangle}{\phi_n} \right)^{1/\alpha}, \qquad \bar{\delta}_t = \delta_t - \delta_t \left(\frac{\langle \phi_t - \phi_n \rangle}{\phi_t} \right)^{1/\beta} \tag{11}$$

respectively, where $\sigma_{\max}$ and $\tau_{\max}$ are the mode I and mode II cohesive strengths, respectively. In total, there are six user inputs to the extrinsic PPR cohesive model: ϕ_n, ϕ_t, α, β, σ_{max}, τ_{max}. A set of sample traction-separations relationships are illustrated in Figure 4, for $\phi_n = 100$N/m, $\sigma_{max} = 40$MPa, $\alpha = 3.0$, $\phi_t = 200$N/m, $\tau_{max} = 30$MPa, and $\beta = 5.0$.

2.2.2. Unloading/reloading. For this study, the unloading/reloading relationship is uncoupled, meaning that the unloading in the normal direction is independent from the unloading in the tangential direction. Unloading is assumed to occur linearly back to the origin, as seen in Figure 5.

2.2.3. Contact. Contact is assumed to occur when the normal separation within a cohesive element becomes negative. We use the penalty stiffness approach, in which a high stiffness counteracts the

DOI: 10.1002/nme

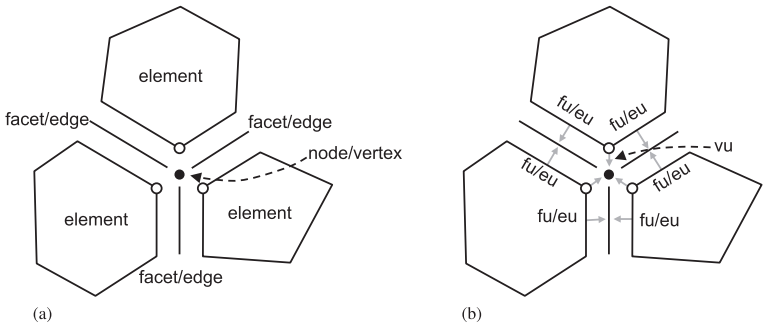

Figure 6. Topological entities: (a) unoriented entities include nodes and elements (explicitly stored) and edges, vertices, and facets (implicitly stored); (b) oriented entities include edge-uses (eu), facet-uses (fu), and vertex-uses (vu).

interpenetration of elements. The stiffness is assumed to be linear, with a relation of $T_n = \alpha^p \Delta_n$, where α^p is the slope of the penalty stiffness:

$$\alpha^p = 10\sigma_{max}/\delta_n. \tag{12}$$

2.3. Consistent topological data structure

The use of an extrinsic cohesive zone model requires on-the-fly mesh modification during the simulation. In order to handle these mesh modifications, the present work uses the consistent topological data structure (TopS) [40, 41], which allows for efficient modifications to the adjacency relations on an as-needed basis. Adjacency information is retrieved, and cohesive elements are dynamically inserted in time proportional to the number of entities retrieved/inserted.

Node and element entities are explicitly stored in memory, while edges, vertices, and facets are implicit entities. From the application point of view, there is no difference between implicit and explicit entities as they are accessed in the same way. TopS is a complete topological data structure, meaning that from any entity, the adjacency of all entities can be determined (e.g., from a given node, all adjacent nodes, elements, facets, edges, and vertices can be found). Additional entities, called uses, are oriented and indicate the use of an edge, vertex, or facet by an element. The entity definition and storage used in the TopS data structure makes it applicable for both 2D and 3D mesh representation. Several finite elements are supported by TopS, including classic elements (e.g., linear and quadratic triangles and quads, linear and quadratic tetrahedron and bricks, etc.) and the 2D linear polygonal elements that may contain any number of nodes used in this work. A schematic of the TopS entities for a polygonal element is illustrated in Figure 6. In 2D, edges and facets are equivalent, as are nodes and vertices. Cohesive elements are also supported by TopS; in this work, we use 2D linear cohesive elements to be compatible with the linear polygonal elements.

3. INNER-ELEMENT CRACKING BY ELEMENT SPLITTING

Crack patterns on geometrically structured meshes often suffer from mesh bias [7]. To reduce this bias, unstructured meshes, such as those employed in this work with polygonal elements, are desirable. However, crack patterns on polygonal element meshes are quite restricted because each node typically only has three to four edges emanating from it; therefore, there are only two or three possible directions for a crack to propagate from a crack tip node. To circumvent this problem, and improve crack patterns on polygonal element meshes, we implement an element splitting operator that increases the potential crack directions at each crack tip, as illustrated in Figure 7.

3.1. Methodology

Cracks are advanced by inserting extrinsic cohesive zone elements when and where they are needed during the dynamic fracture simulation. At the same time that the insertion of cohesive elements is

DOI: 10.1002/nme

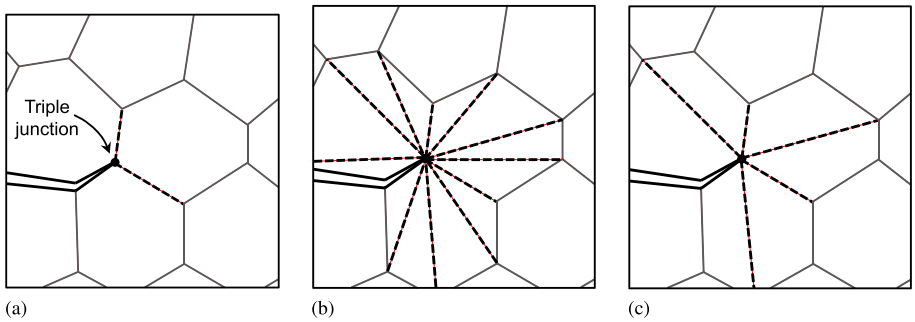

Figure 7. Schematic of a crack in a polygonal mesh, where crack faces are illustrated with solid black lines, the crack tip is indicated by a black circle, and potential crack paths are shown as dashed lines. (a) Crack directions are limited to existing facets on a plain mesh. (b) The crack may propagate to any adjacent node, with no restrictions, when element splitting is enabled. (c) Element splitting is restricted to certain nodes, in order to preserve element quality.

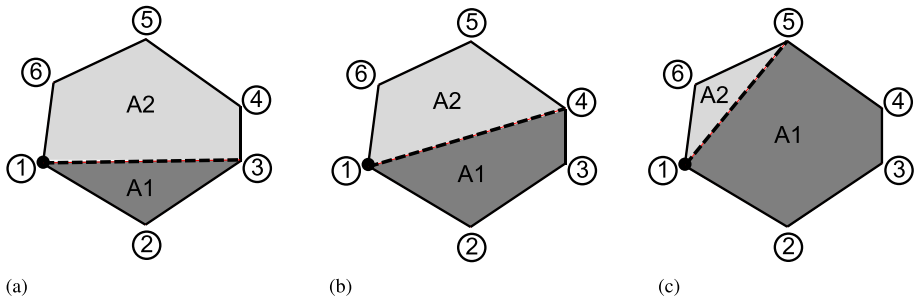

Figure 8. Schematic of potential new elements that would result from splitting element between nodes 1 and (a) 3 and (b) nodes 4 and (c) 5. The configuration shown in (b), where the element is split with node 4, minimizes the difference between the areas, A1 and A2, of the resulting new elements. Once an element is split, the resulting two elements cannot be split again.

checked, bulk elements are also evaluated for splitting. To start the procedure, stresses are computed at nodes, and nodes with principal stress greater than 90% of the cohesive strength are flagged for further inspection. The flagged nodes are used to identify (1) potential elements to be split and (2) potential facets along which to insert cohesive elements. Although a stress-based criterion has been adopted, the present methodology can be used in conjunction with other criteria, as used in references [42–45].

Each element adjacent to a flagged node is checked for splitting. Starting from the flagged node, the element may be split using the node that minimizes the difference in the areas between the two newly created elements. The element edge size is directly related to the critical time step for the explicit dynamic time integrator. Thus, the approach of limiting the nodes on which an element can be split is taken to avoid additional reduction of the already small time step. Furthermore, once an element is split, the resulting two elements cannot be split again. Thus, recurring element splitting is not allowed in the present work. For example, in Figure 8, node 1 is flagged and the element could be split with either node 3, 4, or 5, as shown in parts (a), (b), and (c), respectively. The resulting areas are indicted by the shaded regions and labeled, A1 and A2, in Figure 8. The element is only allowed to be split with node 4 because it produces two new elements whose difference in area is minimized when compared to splitting with node 3 or 5. To determine if the element should actually be split, the traction along the facet is computed by averaging the facet's nodal stresses. If the average normal traction along the facet that would connect the splitting nodes is greater than the normal cohesive strength, then the element is split. In the example in Figure 8, the two new elements connected by nodes 1-4-3-2 and 1-6-5-4, respectively, (Figure 8(b)), will not be split again. The insertion of cohesive elements through the polygonal element by means of the adaptive splitting proposed here is illustrated in Figure 9.

DOI: 10.1002/nme

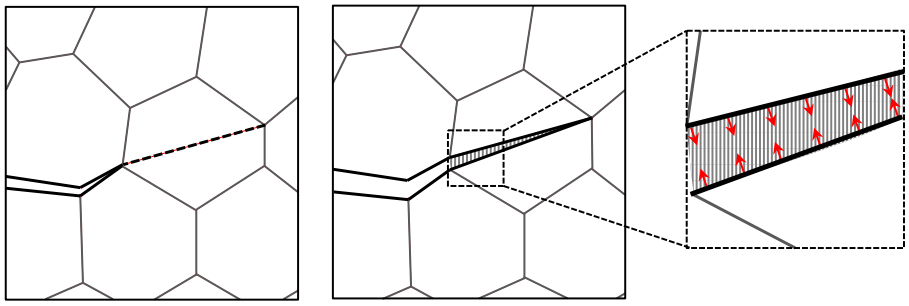

Figure 9. Schematic of a cohesive element being inserted at the new facets of split polygonal element. (a) The element ahead of the crack tip meets the criteria to be split, indicated by the dashed line. (b) The polygonal element is split, and a cohesive element is inserted at the new facets. The zoomed-in region shows a schematic of the cohesive tractions given by Equations (6)–(7).

Use of the TopS data structure, discussed in Section 2.3, ensures that the splitting operation is computationally efficient. The original polygonal element is removed then two new polygonal elements are inserted using appropriate functions available in the TopS API. Linear polygons are used in this work, so no new nodes are created when elements are split; therefore, it is not necessary to interpolate new nodal values. If higher order elements were used, mid-side nodal quantities would need to be interpolated from existing nodes. Moreover, because all of the polygons in the original mesh are convex, the polygons that result from splitting the element along its nodes are also convex. Thus, we do not encounter issues of degenerate finite elements.

Numerical integration of the split elements is accomplished by the same means as in the intact elements: the n-sided element is subdivided into n non-overlapping triangles that meet at the centroid of the polygon. Then, a three-point quadrature rule is employed on each triangle, and its contribution is added to the element. Thus, new integration points are created in the element splitting procedure. In the current work, we use a bulk material model in which the state of the material is completely characterized by the displacement field, and cohesive elements are never split. Thus, there is no need to map history-dependent internal state variables from the original set of integration points to the new set. However, if a more complex model were used, a technique to map internal state variables, such as that proposed in [46], would be necessary.

The methodology presented here is effective because all of the elements (whether they have been split or not) are polygons, so the shape functions for the n-gons are always computed in the same fashion. This would not be the case, however, if element splitting were employed on a mesh of alternative linear elements. For example, a split linear quad element would result in two linear triangles, which have different shape functions.

During a time step, element splitting is performed before cohesive elements are inserted. The criteria for inserting cohesive elements is the same as that to split an element, so all split elements will have a cohesive element inserted along the new facet.

3.2. Geometric studies on crack length and angle

To investigate the quality of the crack paths produced by finite element meshes, we perform comparison studies on crack lengths and angles between 4k meshes and the polygonal element meshes proposed here. 4k structured meshes are widely used in finite element analysis as they are easily generated and lend themselves well to mesh refinement and coarsening. To reduce bias in the 4k mesh, Paulino *et al.* [1] proposed the nodal perturbation and edge-swap topological operators. Nodal perturbation, shown in Figure 10, results in an unstructured geometry by randomly perturbing the internal nodes. The edge-swap operator reduces undesirable crack patterns by supplying all internal element vertices with the same number of potential crack directions, shown in Figure 11. We compare the polygonal element meshes with and without element splitting to 4k meshes with and without nodal perturbation and with and without edge swapping.

DOI: 10.1002/nme

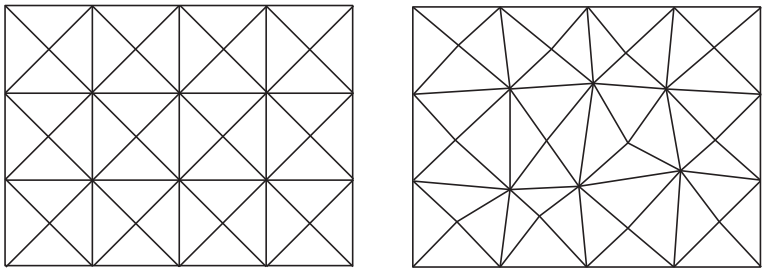

Figure 10. 4k mesh (a) without nodal perturbation and (b) with nodal perturbation.

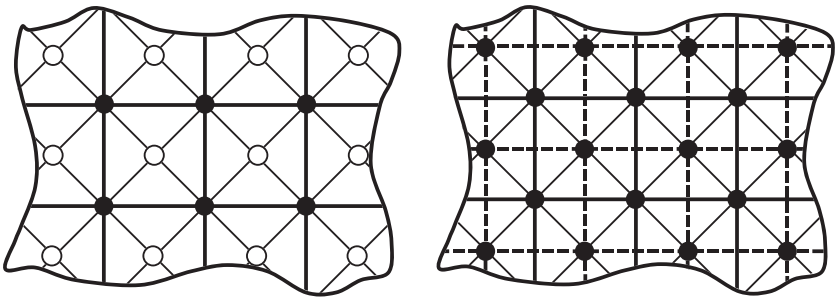

Figure 11. Potential crack path directions on a 4k mesh where the solid nodes have eight potential crack directions and the white nodes have only four directions. (a) Bold edges can be swapped. (b) After the edge-swap operator is applied, the dashed edges become available crack directions.

The polygonal meshes used throughout this work are constructed using the CVT. To investigate possible bias on the CVT mesh, the geometric studies in the succeeding text were also performed on meshes whose seeds were randomly close packed [20] to achieve a near maximal Poisson-disk sampling (near-MPS) [4].

First, the mesh influence on crack length is investigated. The fracture energy, E_f, required to generate a crack is directly proportional to the crack length, that is, $E_f = L_c G_c$, where L_c is the crack length and G_c is the energy release rate. Hence, the length of the crack should not be altered by the finite element representation such that the fracture energy artificially increases [7]. We perform studies similar to those of Rimoli and Rojas [12]. The shortest distance measured along the finite element edges between the start point and the end point is computed using Dijkstra's algorithm [47] then compared to the Euclidean distance. Square domains of 2 units by 2 units, where the center node is at point (0, 0) are used for all studies. The start point is (0, 0), and the end point is chosen as the closest node to the point given by $(r\cos\theta, r\sin\theta)$, for $\theta = 0°$ to 180° where $r = 1$. The domain was selected to be a square in order to fairly compare the 4k and polygonal element meshes. However, to ensure all paths from $\theta = 0°$ to 180° are approximately the same, the shortest path algorithm is terminated when the path reaches a node nearest to the edge of a circle of radius 1.

We compare an unperturbed 4k mesh with and without edge swapping, a perturbed 4k mesh with and without edge swapping, and a polygonal mesh with and without element splitting. A value of 30% perturbation was applied to the 4k meshes in accordance with the recommendation in Paulino *et al.* [1]. Figure 12 shows the paths of minimum distance for $\theta = 71.6°$ on a polygonal mesh of a circular domain with and without element splitting as measured with Dijkstra's algorithm. The distance without element splitting is over 23% longer than the Euclidean distance, while the distance with splitting is less than 4% longer. It should be emphasized that the metric of this study is only the length of the path and not the distance *between* the finite element path and a straight line. Later in this section, we perform a study on Hausdorff distance to address this issue. We also note that the

DOI: 10.1002/nme

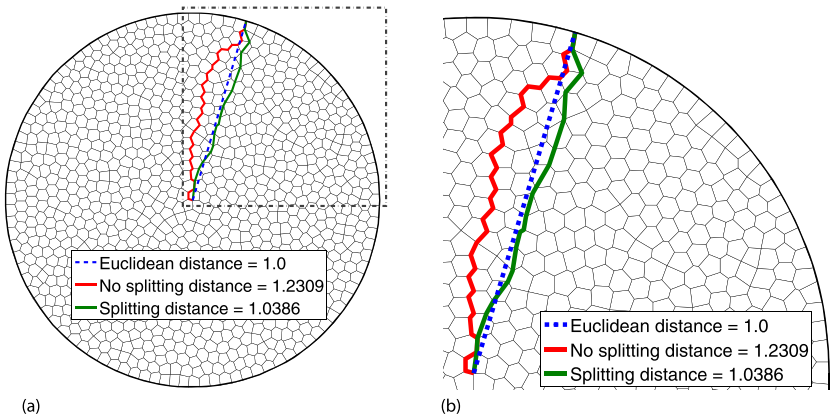

Figure 12. The shortest distance between the start point (0, 0) and the end point (cos (71.6°), sin (71.6°)), measured along the finite element edges versus the Euclidean distance for a polygonal mesh with and without element splitting. (a) Full domain and (b) zoom in of region inside the dotted, gray box in (a). The circular domain shown here is for illustrative purposes only; the studies were conducted on rectangular domains with an inscribed circular boundary, as described in the text.

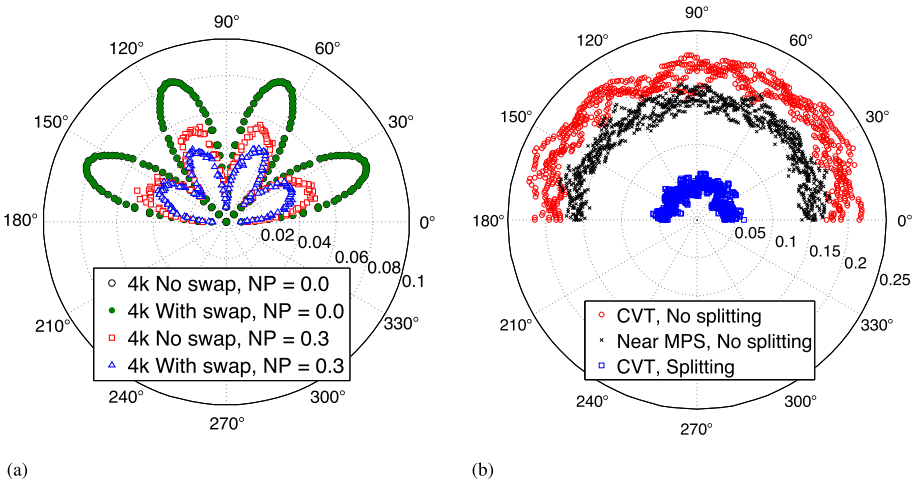

Figure 13. Comparison of mesh deviation, η, for meshes with $\lambda \approx \frac{1}{80}$. (a) Deviations of 4k meshes nodal perturbation factors of 0 and 0.3, with and without edge swap, and (b) deviations of polygonal meshes with and without element splitting.

example shown in Figure 12 is for illustrative purposes only; the domain used for the studies was rectangular, and the mesh is finer than that shown in the figure.

The deviation, η, is quantified as the percent difference between the length of the path created by the finite element edges and the Euclidean distance. The deviations for angles $\theta = 0°$ to 180° are shown in Figure 13. A study of mesh refinement revealed that the deviation did not decrease significantly as the element size decreases; hence, we chose a refinement level that allowed for computationally efficient calculations. Meshes of 10,000 elements are used for all studies. We also report mesh size according to a non-dimensional metric, λ, defined as [12]

$$\lambda = \frac{\frac{1}{N}\sum_{i=1}^{N} h_i}{L_e} \tag{13}$$

where L_e is the Euclidean distance between the start and end points, N is the total number of edges in the mesh, and h_i is the length of edge i. For the meshes used in these studies, $\lambda \approx \frac{1}{80}$.

DOI: 10.1002/nme

Table I. Deviations in crack length for meshes with $\lambda \approx \frac{1}{80}$.

	Average	Standard deviation
4k, no swap, NP = 0.0	0.0543	0.0250
4k, with swap, NP = 0.0	0.0543	0.0250
4k, no swap, NP = 0.3	0.0372	0.0120
4k, with swap, NP = 0.3	0.0302	0.0100
Near-MPS, no split	0.1602	0.0098
Polygonal, no split	0.1997	0.0125
Polygonal, with split	0.044	0.0078

MPS, maximal Poisson-disk sampling.

Because the polygonal meshes are random, results from four meshes are shown in Figure 13(b). As illustrated in Figure 13(a), even with the nodal perturbation and edge-swap operators, all of the 4k meshes suffer from mesh-induced anisotropy [12]. Meshes that do not suffer from mesh anisotropy have approximately equal deviations for all crack mesh angles. Furthermore, it is desirable for the deviation to also tend to zero such that the energy is not larger than expected. The near-MPS meshes do not appear to suffer from mesh anisotropy, as expected, but the deviation is significantly higher than it is in any of the 4k mesh variations, as illustrated in Figure 13(b) and shown in Table I. Based on the results shown in Figure 13(b), the CVT polygonal meshes also appear to be isotropic, but the deviation is higher than that of the near-MPS mesh. While the near-MPS mesh has a lower deviation than the CVT mesh, the near-MPS mesh is not well suited for element splitting because the elements may contain very small edges, which we attempt to avoid because of the restrictions upon the critical time step. Therefore, we employ element splitting on the CVT meshes in an attempt to reduce the deviation. In this case, isotropy appears to be achieved, and the deviation is reduced to an average value in the range of that of the 4k mesh with nodal perturbation and edge swap. We have demonstrated that the near-MPS mesh and the CVT mesh with and without splitting do not exhibit mesh anisotropy, but we direct our focus to the CVT meshes in the reminder of this work because they are better suited for element splitting.

The CVT mesh with splitting achieves a mean deviation near that of the k-means mesh studied by Rimoli and Rojas [12] (see Figure 12 in [12]). The proposed approach is not sufficient to reach the deviation of the conjugate-directions mesh, whose average deviation is less than 0.02. However, as stated earlier, because the element splitting is performed adaptively, the computational storage costs are less than what they would be for the conjugate-directions mesh.

Next, the mesh influence on crack angles is investigated, and results are compared between the 4k and polygonal meshes. We measure the difference between an arbitrary angle θ and the corresponding geometric angle, θ_{FE}. Again, the studies are performed on square domains of 2 units by 2 units with 10,000 elements, where the starting node is at point (0, 0). The procedure to determine the geometric angle, θ_{FE}, is as follows: from the starting location, a local search is performed over the edges adjacent to that node. The angle formed by the start node and the node at the opposite end of the edge is computed. The edge that results in the angle closest to the target angle is selected, and the current location is updated to the node at the opposite end of the selected edge. Then the same procedure is applied until the end point is reached; the end point is the first node whose distance to the starting point is greater than or equal to the target radius of one. The series of nodes and edges from the starting point to the boundary represents the path from which the geometric angle is determined. A least squares linear fit is applied to the path, and the angle is obtained. For illustrative purposes, Figure 14 shows the paths obtained for the case of $\theta = 34°$ for a coarse, unperturbed 4k mesh without edge swapping.

The absolute value in the deviations from target angles 0° to 359° is plotted in Figure 15(a) for the unperturbed 4k mesh without swapping, the perturbed 4k meshes without swapping, and the polygonal meshes without splitting; and 15(b) for the unperturbed 4k mesh with swapping, the perturbed 4k meshes with swapping, and the polygonal meshes with splitting. A single mesh was used to calculate the deviations for the unperturbed 4k mesh, while the absolute values in the angle differences in the randomly perturbed 4k and polygonal mesh results are averaged over 100 meshes.

DOI: 10.1002/nme

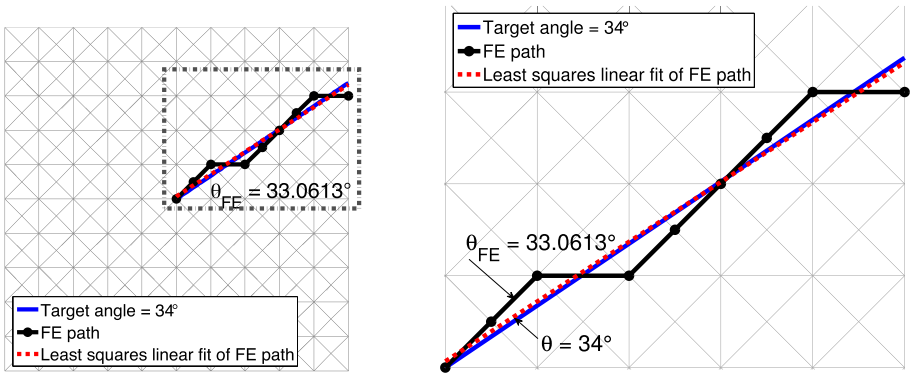

Figure 14. Comparison of an arbitrary angle and geometric angle, $\theta = 34°$, for (a) unperturbed 4k mesh with and without edge swapping and (b) perturbed 4k meshes with and without edge swapping.

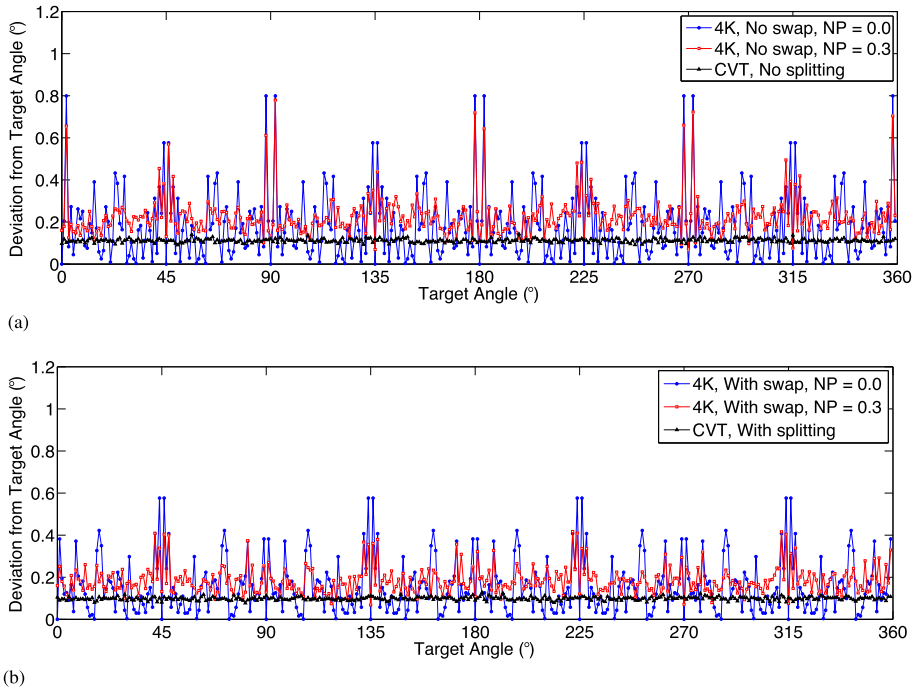

Figure 15. Absolute value of difference between geometric angle and target angle for radial paths from 0° to 359° at 1° increments for (a) 4k without swapping, perturbed 4k without swapping and polygonal without splitting, (b) 4k with swapping, perturbed 4k with swapping and polygonal with splitting. Results for the 4k discretization are obtained from one mesh, while the results from 100 meshes are averaged for each the perturbed 4k and polygonal discretizations.

All of the 4k meshes display mesh bias, where the lowest deviation is for angles of 0°, 45°, 90°, and so on, and highest for angles of $0° \pm 1°$, $45° \pm 1°$, $90° \pm 1°$, and so on. The benefit of the edge-swap operator is clear when comparing Figures 15(a) and 15(b) at 0°, 90°, 180°, and 270°; however, bias still exists. Conversely, the polygonal meshes with and without splitting display no mesh bias, and their average deviations are much lower than those in the 4k meshes. The minimum, maximum, and average deviations from the target angle are shown in Table II. While the 4k meshes have minimum deviations as low as or lower than the polygonal meshes, the range of deviations (i.e., difference between maximum and minimum) for the polygonal meshes is smaller than that of the 4k meshes.

DOI: 10.1002/nme

Table II. Deviations from target angle.

	Max	Min	Mean
4k, no swap, NP = 0.0	0.80°	0°	0.18°
4k, no swap, NP = 0.3	0.78°	0.07°	0.23°
Polygonal, no split	0.14°	0.09°	0.11°
4k, with swap, NP = 0.0	0.58°	0°	0.16°
4k, with swap, NP = 0.3	0.42°	0.07°	0.18°
Polygonal, with split	0.13°	0.07°	0.10°

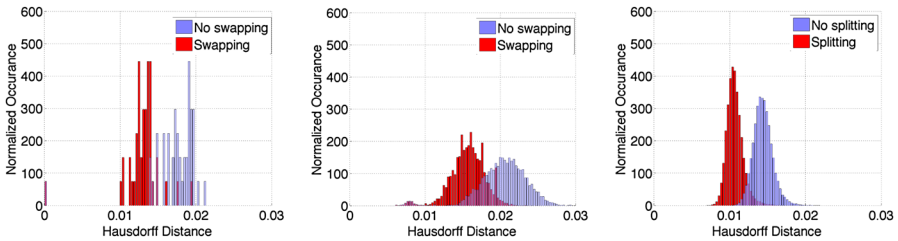

Figure 16. Histograms of Hausdorff distances for radial paths from 0° to 359° at 1° increments for (a) unperturbed 4k mesh with and without edge swapping, (b) perturbed 4k meshes with and without edge swapping, and (c) polygonal mesh with and without element splitting. Results for the 4k discretization are obtained from one mesh, while the results from 100 meshes are averaged for each the perturbed 4k and polygonal discretizations.

The paths obtained for the crack angle study are also used to investigate the Hausdorff distance. Given two curves, P and Q, in the same two-dimensional space, the Hausdorff distance, H, is the maximum of the minimum distances from all points on curve P, denoted p, to all points on curve Q, denoted q. It is defined as [48]

$$H(P, Q) = \max(h(P, Q), h(Q, P)) \tag{14}$$

where

$$h(P, Q) = \max\left[\min_{p \in P}\left[\operatorname{dist}_{q \in Q}(p, q)\right]\right] \tag{15}$$

The bias in the 4k meshes has already been established, so here, we report the occurrence of Hausdorff distances in the histograms in Figures 16(a)–16(c). Again, one mesh was used for the unperturbed 4k mesh case, while 100 were used for the perturbed 4k and polygonal mesh cases. The occurrences are normalized such that the area of each histogram is one. The distribution of the Hausdorff distances for the perturbed meshes is smoother than those of the unperturbed 4k mesh; however, there is some tradeoff when comparing the perturbed and unperturbed 4k meshes. Higher Hausdorff distances occur in the perturbed case, especially when edge swap is not employed: 95.6% of the Hausdorff distances for the unperturbed case are below 0.02, but only 42.5% of the distances for the perturbed case fall below this threshold. There is improvement when edge swap is used: 100% of the Hausdorff distances for the unperturbed case and 98.5% for the perturbed case are lower than 0.02. The striking difference comes in the comparison of the 4k meshes to the polygonal meshes. As illustrated in Figure 16(c), the Hausdorff distances for the no splitting and splitting cases are lower, and the distribution is tighter than for the 4k meshes.

The polygonal meshes with element splitting are shown to be superior to 4k meshes with nodal perturbation and adaptive edge-swap operators in geometric studies. In the next section, numerical studies are performed to demonstrate the capability of this method for dynamic fracture problems.

DOI: 10.1002/nme

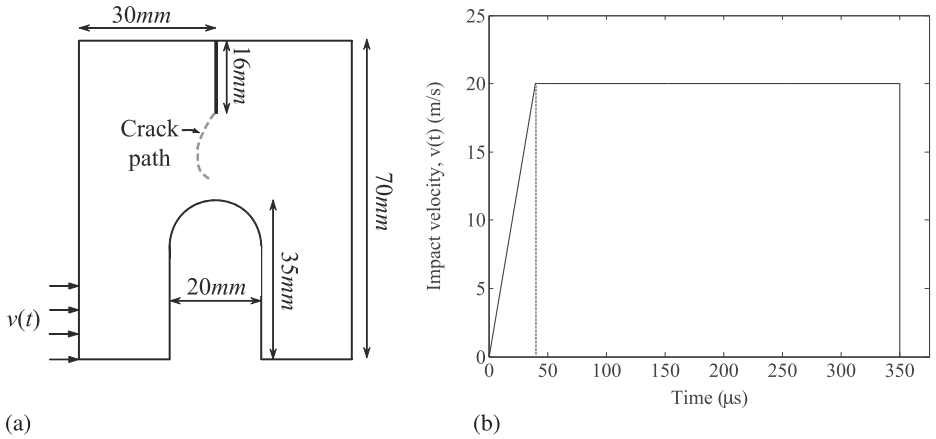

Figure 17. (a) Geometry of compression specimen test and (b) impact velocity with respect to time.

4. NUMERICAL EXAMPLES

In this section, we explore the effectiveness of the proposed scheme over the standard polygonal element approach through two numerical examples.

4.1. Compact compression specimen

Rittel and Maigre [49, 50] used the compact compression specimen (CCS) test [51, 52] to investigate mixed-mode dynamic crack propagation in PMMA. The CCS, illustrated in Figure 17(a), is fractured by applying an impact load using a Hopkinson bar [53] with a diameter of 16.5 mm. Numerically, the impact is described by an applied velocity [1] that increases linearly over a span of 40 μs, and lasts for 350 μs, as illustrated in Figure 17(b). The present approach is appropriate for this problem because (1) the curved domain is easily meshed with polygons, and (2) the expected crack is curved and could not be captured fully by a structured mesh.

Various numerical methods have been applied to model this experiment. Papoulia *et al.* [7] use the pinwheel meshes to discretize the domain and linearly softening, extrinsic cohesive elements [54] to model fracture. While the pinwheel mesh can capture increasingly smooth crack paths as the mesh is refined, the generation of the mesh is not straightforward. Paulino *et al.* combine nodal perturbation and edge-swap operators to study the crack path on a 4k mesh [1]. They use the same extrinsic cohesive elements as the current study, but display mesh-induced anisotropies as the cracks move farther from the notch tip. The 4k mesh is first generated on a rectangle, then a curvilinear coordinate transformation and conformal mapping procedure is applied to discretize the region around the notch. Menouillard and coworkers use the extended FEM to simulate the CCS experiment [55–57]. They study the effect of mass lumping techniques and explicit time integration schemes, which prove to have an influence on the resulting crack pattern.

In the present study, element generating seeds are placed throughout the domain. To achieve the necessary regions of refinement around the notch and impact location, the seed placement was controlled with user defined density functions [23]. The density function at the notch tip is exponential, with a probability of seeds being placed in this region 10^8 times greater than in the remainder of the domain. The rectangular region around the impact location has a 10^3 times greater probability of seed placement than in the remainder of the domain. A well-graded polygonal mesh is generated with this approach. The domain is discretized with 6,000 polygons and 11,403 nodes, as illustrated in Figure 18. In comparison, Paulino *et al.* [1] discretize the same domain with a 4k mesh and quadratic finite elements; resulting in a similar number of nodes (10,908).

The PMMA material is modeled with an elastic modulus of 5.76 GPa, a Poisson's ratio of 0.42 and a density of 1180 kg/m^3. The cohesive input parameters are adopted from the work of Paulino *et al.* [1]. The cohesive strength ($\sigma_{\max}$), fracture energy (ϕ_n), and shape parameter (α) are set as

DOI: 10.1002/nme

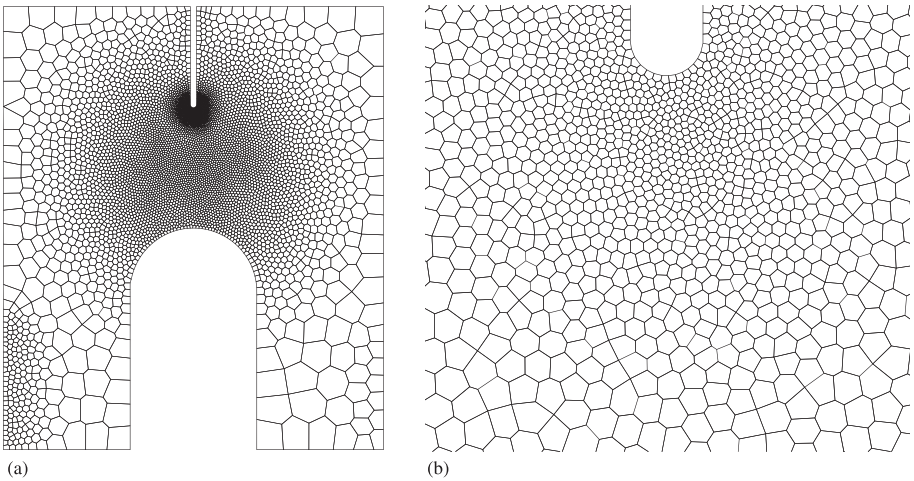

Figure 18. (a) Full mesh of compression specimen and (b) refined mesh around notch.

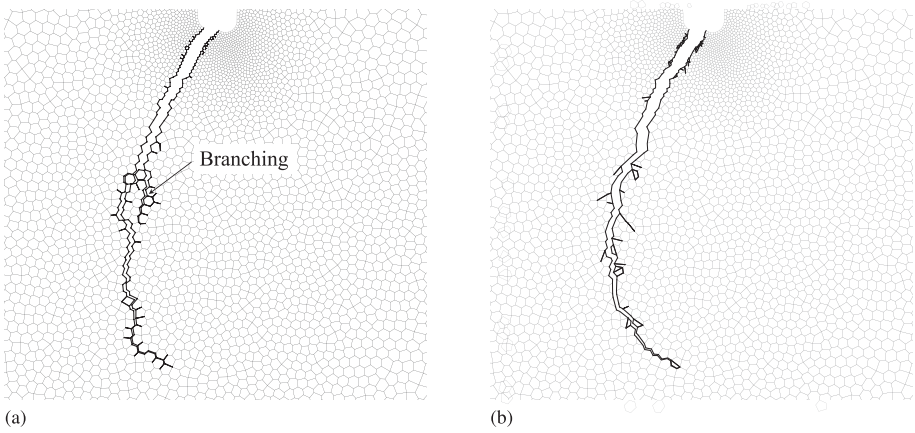

Figure 19. (a) Fracture emanating from notch without element splitting, and (b) fracture emanating from notch is smoother when element splitting is employed. Notice the branching that develops from the main crack when there is no element splitting. The dark lines represent the inserted cohesive elements.

129.6 MPa, 352.3 N/m, and 2, respectively. The normal and tangential separation parameters are set equal to one another.

The simulation is performed for a polygonal mesh without element splitting, then again on the same polygonal mesh with element splitting enabled. The crack paths for both simulations are illustrated in Figure 19. In the case without element splitting, the fracture surface is noticeably biased by the polygonal element geometry, the crack branches, and some fragmentation is observed as the crack propagates into the coarse portion of the mesh. These features are not present in numerical and experimental results in the literature [7, 49]. When element splitting is enabled, 51 elements are split, alleviating much of the mesh-induced bias and resulting in a smoother fracture surface. The fracture energy dissipated by the cohesive elements for each case is compared at a time of 100 μs. The case without element splitting dissipates 5.001 N/m; whereas the case with element splitting dissipates 4.628 N/m. The crack trajectory for the splitting case agrees well with reported experimental results [58] and numerical results published in the literature [7].

4.2. Pressure loaded cylinder

To further explore the applicability of polygonal elements with element splitting for fracture simulation, we examine the fragmentation of a cylinder with an internal impact pressure. This problem

DOI: 10.1002/nme

is particularly suitable for the use of unstructured meshes as structured meshes can induce undesirable fracture patterns; see [59] for a discussion of unsatisfactory results obtained because of mesh dependency for the pressure loaded cylinder. The thick cylinder has an internal radius of 80 mm and outer radius of 150 mm, as illustrated in Figure 20(a). The applied pressure increases linearly from 0 to 400 MPa over the course of 1 μs then decreases exponentially according to $P(t) = 400e^{(-(t-1)/100)}$, as illustrated Figure 20(b). The material has an elastic modulus of 210

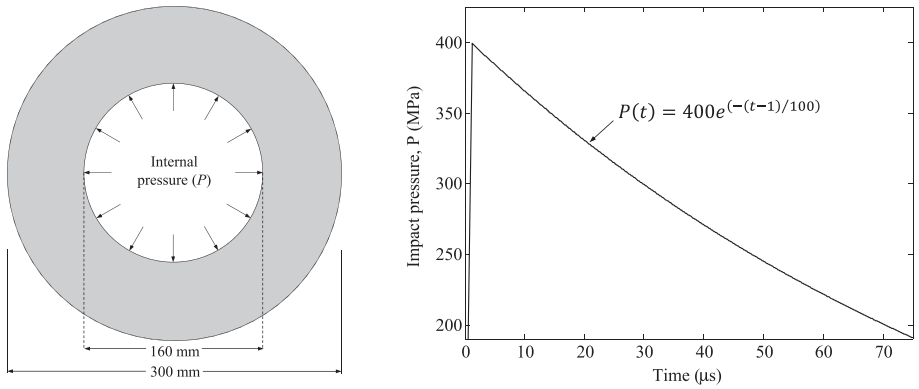

Figure 20. (a) Geometry of cylinder and (b) internal impact pressure with respect to time.

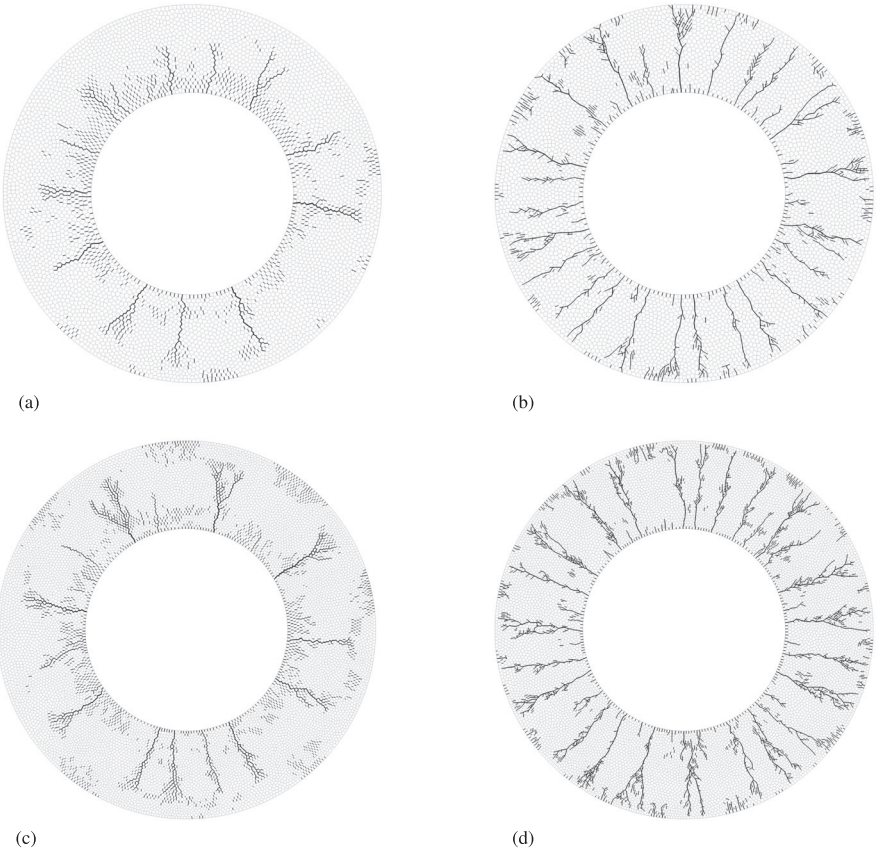

Figure 21. Comparison of results at a time of 64 μs without element splitting and with element splitting. (a) 5,000 elements without element splitting, (b) 5,000 elements with element splitting, (c) 10,000 elements without element splitting, and (d) 10,000 elements with element splitting.

DOI: 10.1002/nme

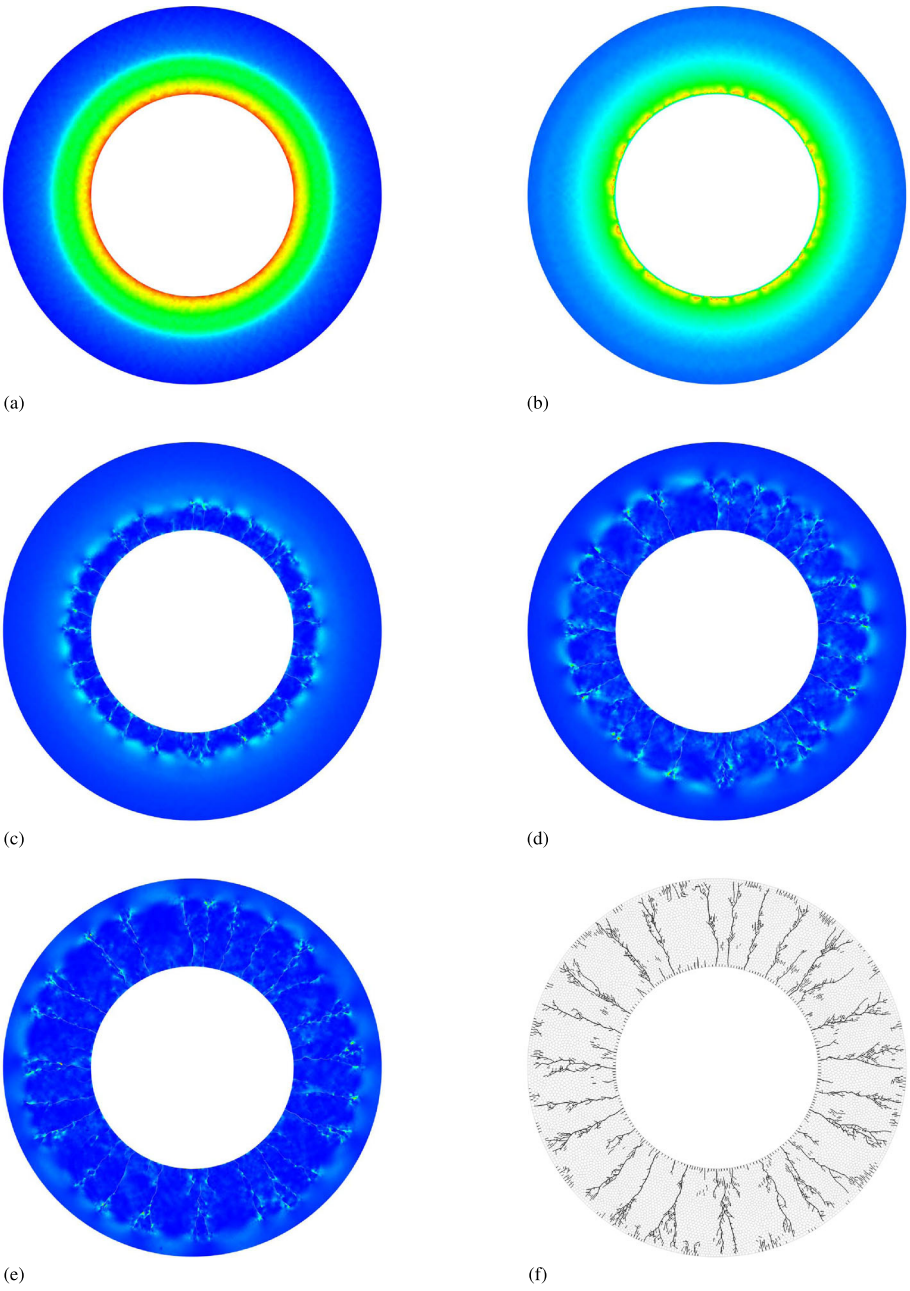

Figure 22. Progression of energy through fragmentation process, for a mesh size of 10,000 elements with element splitting enabled, at time (a) 30 μs, (b) 40 μs, (c) 47.5 μs, (d) 55 μs, and (e) 60 μs. (f) Final fragmented geometry at 64 μs.

GPa, a Poisson's ratio of 0.3 and a density of 7850 kg/m^3. The cohesive strength, fracture energy, and shape parameter in both modes are 850 MPa, 2000 N/m and 2, respectively.

In a prior numerical investigation, Rabczuk and Belytschko [60] use a mesh free cracking-nodes method and observe that the number of large and small fragments are consistent, regardless of the number of particles used to model the geometry. Song and Belytschko [61] use a cracking node method within the finite element framework. In order to break the mesh-induced symmetry in their model, they perturb the elastic modulus throughout the model by $\pm 5\%$. They also show that the number of fragments is consistent through mesh refinement and are on the order of 20 fragments. Zhou and Molinari [59] implement the same problem in 3D. They propose a method of

DOI: 10.1002/nme

overcoming mesh dependency by distributing the cohesive strength based on a modified weakest link Weibull distribution. The larger the finite element is in their mesh, the lower the cohesive strength associated with that element. Without this methodology, the cracks initiate in the interior of the cylinder. Paulino *et al.* [1] use an edge-swap operator (Section 3.2) to increase the number of fracture paths in their finite element mesh. To overcome the symmetry in their model, they implement nodal perturbation, but also impose restrictions on the location of crack initiation, so that they do not initiate in the interior of the cylinder.

In the present study, the domain is discretized at two levels of refinement, 5,000 and 10,000 polygons, and we investigate cases with and without element splitting in each. No external method (e.g., perturbation) is required to break mesh symmetry because the mesh is inherently random, and no restrictions are placed on the initiation of fracture. The final crack patterns for all cases are shown in Figure 21, and the progression of the strain energy as fracture occurs for the 10,000 polygonal element mesh with element splitting enabled is shown in Figure 22.

In both refinement levels without element splitting, we observe that many cohesive elements are inserted near the internal boundary. The insertion of cohesive elements demonstrates excessive material softening in this region, causing the cracks to arrest instead of propagating completely through the thickness of the cylinder. At a finer mesh discretization, the standard polygonal meshes are still not able to capture the full fragmentation of the cylinder. Using the same initial polygonal meshes and boundary conditions, but allowing for element splitting, the resulting behavior and fragmentation pattern agree with the expected results from the literature [1, 60, 61]. In comparison, the mesh with 5,000 polygonal elements contains 9,419 nodes, whereas that presented by Paulino *et al.* [1], for the 4k mesh results in similar fracture behavior but contains 12,800 quadratic elements and 25,920 nodes.

The stress and strain are initially concentrated around the inner boundary of the cylinder (Figure 22(a)), causing fracture to initiate almost uniformly around the inner surface (Figure 22(b)). The majority of these cracks arrest after a small amount of propagation with several cracks propagating through the entire thickness (Figure 22(c)–22(e)). At the end of the simulation (64 μs), we are interested in the number of fragments, where a fragment is defined as the volume of material between two cracks, which are continuous and span the thickness of the cylinder. The final geometry consists of 18 fragments for the 5,000 element case and 23 fragments for the 10,000 element case, which is in good agreement with the results available in the literature.

5. CONCLUDING REMARKS

In this work, we perform dynamic fracture simulations using the FEM and an inter-element cohesive zone model. To reduce mesh bias, polygonal finite elements constructed with the CVT are employed. While the polygonal mesh is shown to be isotropic, the resulting crack patterns are not ideal because each node typically only has three to four edges emanating from it; therefore, there are only two or three possible directions for a crack to propagate from a crack tip node. To circumvent this problem, we introduce polygonal element splitting, which essentially doubles the number of edges connected to a node. We find the that CVT mesh is better suited for element splitting than other types of polygonal element meshes, namely the near-MPS, because they result in elements that have less variation in edge length.

We performed geometric studies on deviation in crack length and angle to investigate the quality of crack paths for various discretizations. The polygonal element meshes were compared to the popular 4k mesh, which has been widely used in finite element and fracture applications because it is easy to construct and is well suited for mesh adaptivity. The studies indicate the strength of the polygonal element meshes with splitting over plain polygonal element meshes and over 4k meshes with and without the nodal perturbation and edge-swap operators employed.

Numerical dynamic fracture simulations further illustrate the suitability of the polygonal element meshes with splitting. To highlight the benefit of polygonal elements in meshing arbitrary domains, problems with curved boundaries were investigated. We show that the plain polygonal element meshes do not provide the expected results in either numerical simulation when compared to

DOI: 10.1002/nme

numerical and experimental results in the literature. However, crack patterns that agree with expectations are obtained with element splitting enabled. In addition, linear polygonal elements are shown to produce similar or better results to those obtained using quadratic triangular elements with a similar or fewer number of degrees of freedom.

On-the-fly mesh adaptivity (i.e., insertion of extrinsic cohesive elements and element splitting) is made computationally tenable by the TopS data structure. In previous studies, TopS has been used to perform adaptive mesh refinement on 4k meshes in crack tip regions [10]. Thus, a natural extension of the work presented here is to incorporate adaptive mesh refinement in polygonal element meshes with element splitting.

This work offers room for further extensions. While the present adaptive splitting of polygonal elements in two dimensions is not conducive to three-dimensional fracture on polyhedral elements, we indicate some directions to be pursued to achieve that goal. For instance, recently, the Wachspress shape functions, together with their gradients, have been derived in 3D [62] and either those or other related interpolants [63] could serve as a starting point. The geometrical treatment regarding three-dimensional splitting and element redefinition may incorporate some of the computational geometry ideas developed at Sandia National Laboratory [64] and in reference [65]. Alternatively, new interpolants may open avenues for the three-dimensional treatment of fracture surfaces on polytopes. In this regard, the virtual element method (VEM) may be explored. Recently, the VEM has been implemented to treat elastic problems without cracks [66], and thus, one could build upon this work to incorporate new crack surfaces where and when needed, which could take advantage of the adaptive features offered by the VEM [67, 68].

ACKNOWLEDGEMENTS

Sofie E. Leon, Daniel W. Spring, and Glaucio H. Paulino gratefully acknowledge the support of the US National Science Foundation (NSF) Graduate Research Fellowship, the Natural Sciences and Engineering Research Council of Canada, and the Donald B. and Elizabeth M. Willett endowment at the University of Illinois at Urbana-Champaign, respectively. We also acknowledge the support from the NSF through grants CMMI #1321661 and CMMI #1437535. The authors would also like to extend their appreciation to Dr. Cameron Talischi for his advice and input to this publication.

REFERENCES

1. Paulino GH, Park K, Celes W, Espinha R. Adaptive dynamic cohesive fracture simulation using nodal perturbation and edge-swap operators. *International Journal for Numerical Methods in Engineering* 2010; **84**(11):1303–1343.
2. Camacho G, Ortiz M. Computational modelling of impact damage in brittle materials. *International Journal of Solids and Structures* 1996; **33**(20–22):2899–2938.
3. Bolander J, Sukumar N. Irregular lattice model for quasistatic crack propagation. *Physical Review B* 2005; **71**(9):094106-1–094106-12.
4. Ebeida MS, Mitchell SA. Uniform random Voronoi meshes. In *Proceedings of the 20th International Meshing Roundtable*. Springer: Paris, France, 2012; 273–290.
5. Potyondy DO, Wawrzynek PA, Ingraffea AR. An algorithm to generate quadrilateral or triangular element surface meshes in arbitrary domains with applications to crack propagation. *International Journal for Numerical Methods in Engineering* 1995; **38**(16):2677–2701.
6. Freitas MO, Wawrzynek PA, Cavalcante-Neto JB, Vidal CA, Martha LF, Ingraffea AR. A distributed-memory parallel technique for two-dimensional mesh generation for arbitrary domains. *Advances in Engineering Software* 2013; **59**:38–52.
7. Papoulia KD, Vavasis SA, Ganguly P. Spatial convergence of crack nucleation using a cohesive finite-element model on a pinwheel-based mesh. *International Journal for Numerical Methods in Engineering* 2006; **67**(1):1–16.
8. Radin C, Sadun L. The isoperimetric problem for pinwheel tilings. *Communications in Mathematical Physics* 1996; **177**(1):255–263.
9. Velho L, Gomes J. Variable resolution 4-k meshes: concepts and applications. *Computer Graphics Forum* 2000; **19**(4):195–212.
10. Park K, Paulino GH, Celes W, Espinha R. Adaptive mesh refinement and coarsening for cohesive zone modeling of dynamic fracture. *International Journal for Numerical Methods in Engineering* 2012; **92**(1):1–35.
11. Rimoli JJ, Rojas JJ, Khemani F. On the mesh dependency of cohesive zone models for crack propagation analysis. *53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, American Institute of Aeronautics and Astronautics, Reston, Virigina, 2012.

DOI: 10.1002/nme

12. Rimoli JJ, Rojas JJ. *Meshing strategies for the alleviation of mesh-induced effects in cohesive element models*, 2013. arXiv preprint arXiv:13021161.
13. Ghosh S. *Micromechanical Analysis and Multi-Scale Modeling Using the Voronoi Cell Finite Element Method*. CRC Press/Taylor & Francis: Boca Raton, Florida, 2011.
14. Li S, Ghosh S. Multiple cohesive crack growth in brittle materials by the extended Voronoi cell finite element model. *International Journal of Fracture* 2006; **141**(3–4):373–393.
15. Li S, Ghosh S. Extended Voronoi cell finite element model for multiple cohesive crack propagation in brittle materials. *International Journal for Numerical Methods in Engineering* 2006; **65**(7):1028–1067.
16. Ooi ET, Yang ZJ, Guo ZY. Dynamic cohesive crack propagation modelling using the scaled boundary finite element method. *Fatigue & Fracture of Engineering Materials & Structures* 2012; **35**(8):786–800.
17. Ooi ET, Song C, Tin-Loi F, Yang ZJ. Automatic modelling of cohesive crack propagation in concrete using polygon scaled boundary finite elements. *Engineering Fracture Mechanics* 2012; **93**:13–33.
18. Ooi ET, Song C, Tin-Loi F, Yang Z. Polygon scaled boundary finite elements for crack propagation modelling. *International Journal for Numerical Methods in Engineering* 2012; **91**(3):319–342.
19. Sukumar N, Bolander JE. Voronoi-based interpolants for fracture modelling. In *Tessellations in the sciences: Virtues, Techniques and Applications of Geometric Tilings*, van de Weygaert R, Vegter G, Ritzerveld J, Icke V (eds). Springer-Verlag, 2009.
20. Bishop JE. Simulating the pervasive fracture of materials and structures using randomly close packed Voronoi tessellations. *Computational Mechanics* 2009; **44**(4):455–471.
21. Bishop JE, Strack OE. A statistical method for verifying mesh convergence in Monte Carlo simulations with application to fragmentation. *International Journal for Numerical Methods in Engineering* 2011; **88**(3):279–306.
22. Belytschko T, Liu WK, Moran B. *Nonlinear Finite Elements for Continua and Structures*. John Wiley & Sons, Inc.: West Sussex, England, 2000.
23. Talischi C, Paulino GH, Pereira A, Menezes IFM. PolyMesher: a general-purpose mesh generator for polygonal elements written in Matlab. *Structural and Multidisciplinary Optimization* 2012; **45**(3):309–328.
24. Talischi C, Paulino GH, Pereira A, Menezes IFM. Polygonal finite elements for topology optimization: a unifying paradigm. *International Journal for Numerical Methods in Engineering* 2009; **82**(6):671–698.
25. Wachspress EL. *A Rational Finite Element Basis*. Academic Press: New York, 1975.
26. Sukumar N, Tabarraei A. Conforming polygonal finite elements. *International Journal for Numerical Methods in Engineering* 2004; **6**(12):2045–2066.
27. Talischi C, Paulino GH, Pereira A, Menezes IFM. PolyTop: a Matlab implementation of a general topology optimization framework using unstructured polygonal finite element meshes. *Structural and Multidisciplinary Optimization* 2012; **45**(3):329–357.
28. Mousavi SE, Xiao H, Sukumar N. Generalized Gaussian quadrature rules on arbitrary polygons. *International Journal for Numerical Methods in Engineering* 2009; **82**(1):1–26.
29. Natarajan S, Bordas S, Roy Mahapatra D. Numerical integration over arbitrary polygonal domains based on Schwarz-Christoffel conformal mapping. *International Journal for Numerical Methods in Engineering* 2009; **80**(1):103–134.
30. Hinton E, Campbell JS. Local and global smoothing of discontinuous finite element functions using a least squares method. *International Journal for Numerical Methods in Engineering* 1974; **8**(3):461–480.
31. Newmark NM. A method of computation for structural dynamics. *Journal of the Engineering Mechanics Division* 1959; **85**(7):67–94.
32. Hinton E, Rock T, Zienkiewicz OC. A note on mass lumping and related processes in the finite element method. *Earthquake Engineering & Structural Dynamics* 1976; **4**(3):245–249.
33. Hughes TJR. *The Finite Element Method*. Prentice-Hall, Inc.: Edgewood Cliffs, NJ, 1987.
34. Klein PA, Foulk JW, Chen EP, Wimmer SA, Gao HJ. Physics-based modeling of brittle fracture: cohesive formulations and the application of meshfree methods. *Theoretical and Applied Fracture Mechanics* 2001; **37**(1):99–166.
35. Park K, Paulino GH. Computational implementation of the PPR potential-based cohesive model in ABAQUS: educational perspective. *Engineering Fracture Mechanics* 2012; **93**(C):239–262.
36. Song SH. Fracture of asphalt concrete: a cohesive zone modeling approach considering viscoelastic effects. *Ph.D. Thesis*, University of Illinois at Urbana-Champaign, 2006.
37. Falk M, Needleman A, Rice J. A critical evaluation of cohesive zone models of dynamic fracture. *Journal de Physique. IV* 2001; **11**(5):43–50.
38. Park K, Paulino GH, Roesler JR. A unified potential-based cohesive model of mixed-mode fracture. *Journal of the Mechanics and Physics of Solids* 2009; **57**(6):891–908.
39. Park K. Potential-based fracture mechanics using cohesive zone and virtual internal bond modeling. *Ph.D. Thesis*, University of Illinois at Urbana-Champaign, 2009.
40. Celes W, Paulino GH, Espinha R. A compact adjacency-based topological data structure for finite element mesh representation. *International Journal for Numerical Methods in Engineering* 2005; **64**(11):1529–1556.
41. Paulino GH, Celes W, Espinha R, Zhang ZJ. A general topology-based framework for adaptive insertion of cohesive elements in finite element meshes. *Engineering with Computers* 2008; **24**(1):59–78.
42. Erdogan F, Sih GC. On the crack extension in plates under plane loading and transverse shear. *Journal of Basic Engineering* 1963; **85**(4):519–525.
43. Sih GC. Strain-energy-density factor applied to mixed mode crack problems. *International Journal of Fracture* 1974; **10**(3):305–321.

DOI: 10.1002/nme

44. Belytschko T, Chen H, Xu J, Zi G. Dynamic crack propagation based on loss of hyperbolicity and a new discontinuous enrichment. *International Journal for Numerical Methods in Engineering* 2003; **58**(12):1873–1905.
45. Sam CH, Papoulia KD, Vavasis SA. Obtaining initially rigid cohesive finite element models that are temporally convergent. *Engineering Fracture Mechanics* September 2005; **72**(14):2247–2267.
46. Mota A, Sun W, Ostien JT, Foulk JW, Long KN. Lie-group interpolation and variational recovery for internal variables. *Computational Mechanics* 2013; **52**(6):1281–1299.
47. Dijkstra EW. A note on two problems in connexion with graphs. *Numerische Mathematlk* 1959; **1**(1):269–227.
48. Rockafellar RT, Wets RJ-B, Wets M. *Variational Analysis*, Vol. 317. Springer: Heidelberg, Germany, 1998.
49. Rittel D, Maigre H. A study of mixed-mode dynamic crack initiation in PMMA. *Mechanics Research Communications* 1996; **23**(5):475–481.
50. Rittel D, Maigre H. An investigation of dynamic crack initiation in PMMA. *Mechanics of Materials* 1996; **23**(3):229–239.
51. Bui HD, Maigre H, Rittel D. A new approach to the experimental determination of the dynamic stress intensity factor. *International Journal of Solids and Structures* 1992; **29**(23):2881–2895.
52. Rittel D, Maigre H, Bui HD. A new method for dynamic fracture toughness testing. *Scripta Metallurgica et Materialia* 1992; **26**(10):1593–1598.
53. Hopkinson B. A method of measuring the pressure produced in the detonation of high explosives or by the impact of bullets. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 1914; **89**(612):411–413.
54. Sam C-H. A robust formulation and solution of initially rigid cohesive interface models. *Ph.D. Thesis*, Cornell University, 2005.
55. Menouillard T, Rethore J, Combescure A, Bung H. Efficient explicit time stepping for the eXtended finite element method (X-FEM). *International Journal for Numerical Methods in Engineering* 2006; **68**(9):911–939.
56. Menouillard T, Moes N, Combescure A. An optimal explicit time stepping scheme for cracks modeled with X-FEM. In *IUTAM Symposium on Discretization Methods for Evolving Discontinuities*. Springer: Lyon, France, 2007; 267–281.
57. Menouillard T, Rethore J, Moes N, Combescure A, Bung H. Mass lumping strategies for X-FEM explicit dynamics: application to crack propagation. *International Journal for Numerical Methods in Engineering* 2008; **74**(3):447–474.
58. Seelig T, Gross D, Pothmann K. Numerical simulation of a mixed-mode dynamic fracture experiment. *International Journal of Fracture* 1999; **99**(4):325–338.
59. Zhou F, Molinari JF. Dynamic crack propagation with cohesive elements: a methodology to address mesh dependency. *International Journal for Numerical Methods in Engineering* 2004; **59**(1):1–24.
60. Rabczuk T, Belytschko T. Cracking particles: a simplified meshfree method for arbitrary evolving cracks. *International Journal for Numerical Methods in Engineering* 2004; **61**(13):2316–2343.
61. Song JH, Belytschko T. Cracking node method for dynamic fracture with finite elements. *International Journal for Numerical Methods in Engineering* 2009; **77**(3):360–385.
62. Floater M, Gillette A, Sukumar N. *Gradient bounds for Wachspress coordinates on polytopes*, June 2013. arXiv preprint arXiv:1306.4385v2.
63. Floater MS, Kós G, Reimers M. Mean value coordinates in 3D. *Computer Aided Geometric Design* 2005; **22**(7):623–631.
64. Ebeida MS, Mitchell SA, Patney A, Davidson AA, Owens JD. A simple algorithm for maximal poisson-disk sampling in high dimensions. *Computer Graphics Forum* 2012; **31**(2pt4):785–794.
65. Rashid MM, Selimotic M. A three-dimensional finite element method with arbitrary polyhedral elements. *International Journal for Numerical Methods in Engineering* 2006; **67**(2):226–252.
66. Gain AL, Talischi C, Paulino GH. On the virtual element method for three-dimensional elasticity problems on arbitrary polyhedral meshes. *Computer Methods in Applied Mechanics and Engineering*:In press. DOI: 10.1016/j.cma.2014.05.005.
67. Beirão da Veiga L, Brezzi F, Cangiani A, Manzini G, Marini LD, Russo A. Basic principles of virtual element methods. *Mathematical Models and Methods in Applied Sciences* 2013; **23**(01):199–214.
68. Boffi D, Brezzi F, Fortin M. Mixed finite element methods and applications. In *Springer Series in Computational Mathematics*, Vol. 44. Springer: Heidelberg, Germany, 2013.

DOI: 10.1002/nme