2011

Topological changes in 2D simplicial meshes for the simulation of fractures

James R. Thunes
Michigan Technological University

Recommended Citation

TOPOLOGICAL CHANGES IN 2D SIMPLICIAL MESHES FOR THE SIMULATION

OF FRACTURE

By

James R. Thunes

A THESIS

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

(Mechanical Engineering)

MICHIGAN TECHNOLOGICAL UNIVERSITY

2011

This thesis, "Topological Changes in 2D Simplicial Meshes for the Simulation of Fracture,"

is hereby approved in partial fulfillment of the requirements for the Degree of MASTER

OF SCIENCE IN MECHANICAL ENGINEERING.

Department of Mechanical Engineering - Engineering Mechanics

Signatures:

Thesis Advisor   _____

Dr. S. Maiti

Thesis Co-Advisor   _____

Dr. G. Odegard

Committee Member   _____

Dr. Z. You

Chair   _____

Dr. William W. Predebon

Date   _____

Contents

List of Figures

List of Tables

List of Algorithms

Acknowledgments

I would like to thank my advisor, Dr. Spandan Maiti, along with my fellow students in his group, Siladitya Pal in particular, for the help and support they have all shown during the creation of this document. Without thier suggestions and feedback this work would have been inferior to what it has become.

Once more, I would like to thank those who have helped me; those who I have mentioned as well as those who I have forgot. Thank you.

Abstract

A method for the introduction of strong discontinuities into a mesh will be developed. This method, applicable to a number of eXtended Finite Element Methods (XFEM) with intra-element strong discontinuities will be demonstrated with one specific method: the Generalized Cohesive Element (GCE) method. The algorithm utilizes a subgraph mesh representation which may insert the GCE either adaptively during the course of the analysis or *a priori*. Using this subgraphing algorithm, the insertion time is $O(n)$ to the number of insertions. Numerical examples are presented demonstrating the advantages of the subgraph insertion method.

Chapter 1

Introduction

One of the most influential additions to the engineering community in the 20th century was the finite element method. While other improvements have undoubtedly changed the engineering landscape, few could be considered as pervasive as finite elements. The reason for this is evident; computational testing allows faster design revision at lower cost. While less than a century ago, all testing was conducted with expensive and time consuming physical experimentation, most of today's products are subjected to computational testing long before a physical prototype is constructed.

Fracture is one region of design which had until the late 1980s resisted wide-scale computational testing. This changed with the availability of cost-effective and powerful computers starting at that time. As for the finite element methods, the benefits of increasing the use of computational analysis for fracture problems are apparent. However, while the behavior of stresses and strains in a specimen are well understood, there is not a comparable understanding of fracture phenomena. Theoretical and experimental testing since the seminal work by Griffith (1920) have added significantly to the knowledge of the subject, but there are still a number of areas where understanding is limited or contradictory. In addition to the cost, both in time and resources, physical testing for fracture concerns are also ill suited to mainstays of modern engineering such as design revision and optimization.

The increasing use of materials which predominantly fail in fracture highlights the need to accurately model fracture response in the future. The recent development of the Boeing 787 Dreamliner is only one of the myriad products which show industry's focus on non-traditional engineering materials. "High-Performance Composites", a trade magazine in the composites industry, report that the 787 uses 50% composite material by weight, significantly reducing the weight of the aircraft and thus fuel costs

(Griffiths (2005)). This increase in the usage of composite materials is a departure from previous offerings by Boeing such as the Boeing 777, an airframe entered into service 10 years earlier, which is composed of approximately 12% composites.

The usage of materials which require fracture testing is not constrained to the aerospace industry. For example, ceramic materials have been utilized as the main armor plating on the United States M1 Abrams tank and the British Challenger 1 tank. On a smaller scale, inserted ceramic plates in body armor allow for protection from small arms up to rifle rounds. Many plastics, ubiquitous in modern life are brittle materials. In addition to the brittle materials discussed thus far, ductile materials can, under the correct loading scenarios, fail in a brittle manner. Perhaps the most well known of these scenarios is fatigue loading; the explosive decompression of Aloha Airlines Flight 243 in 1988 being a widely reported example. High loading rates, as seen in automobile crashes or ballistic impacts can cause fracture. Ballistic impact in particular may also cause especially high loading rates. Experimental testing for these applications often reach 3km/s (Hohler et al. (1995); Sumramanian and Bless (1995)) and can reach as high as 8km/s (Kozhushko et al. (1991)).

Despite the progress made in computational fracture, there are still issues in the simulation of fracture; especially when the specimen is loaded at a high rate of speed. As a way to decrease computational overhead, many recent studies have suggested the use of fracture methods which update the topological structure of the finite element mesh during the course of the analysis. This work will show a new method to update the topology of the finite element mesh as well as demonstrate the speed and effectiveness of the method.

1.1 Literature Review

1.1.1 Cohesive elements

The cohesive element method, pioneered by Dugdale (1960); Barenblatt (1962); Rice (1968) among others propose that fracture occurs as a gradual separation process along an extended crack tip, with the increase in separation resisted by cohesive tractions. Increasing the separation decreases the magnitude of the traction across the cohesive element until a critical separation is reached resulting in zero traction across the surface and thus a fully fractured surface. By inserting cohesive elements into the finite element mesh before the analysis, the separation in the cohesive elements and thus the nucleation and propagation of cracks in the material may form as required by

the stress state in the material. In this way, the crack response may grow organically over the course of the analysis without a predetermined crack evolution or path.

The generalized cohesive element (GCE), developed by Maiti et al. (2009), expands upon the cohesive element by allowing for a cohesive surface to pass through the interior of the volumetric elements of the mesh. This addition to the cohesive element method allows for cracks to pass through the specimen without restraint to the edges of the volumetric elements as is the case in the conventional cohesive element method.

1.1.2 Dynamic finite element analysis

1.1.2.1 Pinwheel meshes and nodal perturbation

One of the possible problems inherent in the cohesive element method stem from the relationship between the crack path and the mesh geometry. As the cohesive elements are situated at the edges of the volumetric elements, the crack path is confined to pass along the edges of these elements. In cases where the crack path is well known, the mesh may be created such that the element boundaries correspond to the crack path. In other situations, such as unknown or complex crack behavior, this may be impractical. Even in cases where there is a well understood crack behavior, it may not be feasible to tailor the mesh with the expected crack path. Some recent work has been published attempting to solve the issue through changes to the mesh prior to the analysis.

Figure 1.1* describes a thought experiment devised by Papoulia et al. (2006) which highlights the potential problems inherent in computationally modeling fracture using cohesive elements. In the experiment, a crack passes through a region discretized with a structured mesh of triangular elements. The line segment between the corner nodes of the mesh is a mathematical path. The other path, a jagged line between the same two points is an approximation of the mathematical path when the path is constrained to the edges of the volumetric elements of the mesh (as with a cohesive element).

Papoulia et al. (2006) states that in a structured mesh such as shown in the figure, the error between the mathematical and actual path will be approximately 8% regardless

---

*Reprinted from International Journal for Numerical Methods in Engineering 67 (1), Papoulia, K. D., Vavasis, S. A., Ganguly, P., Spatial convergence of crack nucleation using a cohesive finite-element model on a pinwheel-based mesh, 1–16, Copyright 2006, with permission from John Wiley & Sons. See Appendix C for more information.
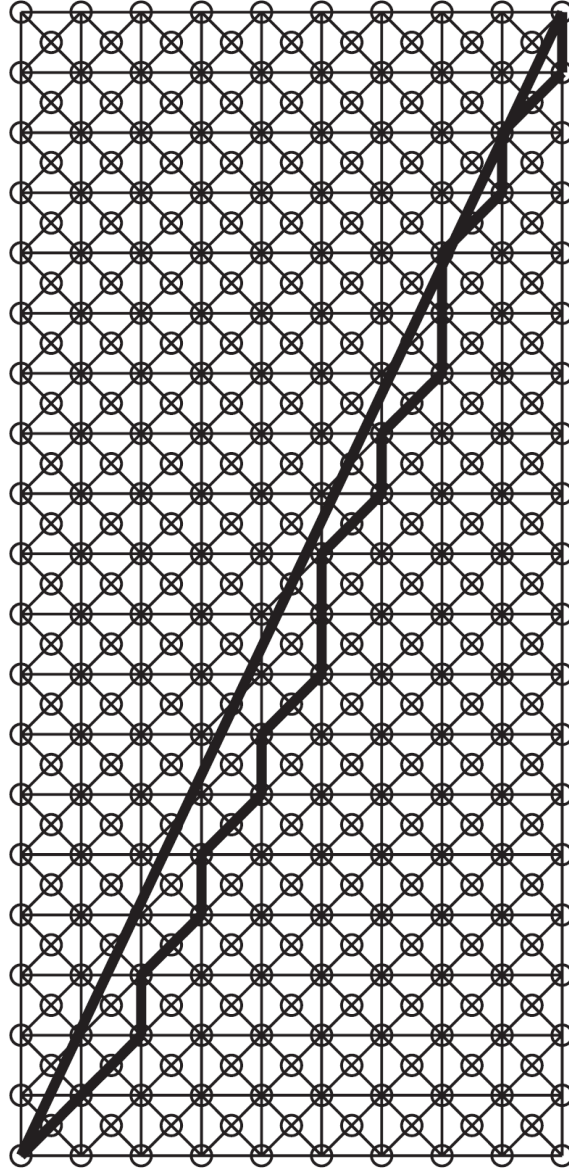
Figure 1.1: A structured mesh with two potential crack paths. The mathematical crack path is the line segment between the corner nodes of the mesh while the best approximation follows the edges of the volumetric elements. The jagged path will always be approximately 8% longer than the mathematical path. Reprinted with permission from Papoulia et al. (2006). See Appendix C for more information.

of the increase in the elements in the mesh. Park (2009) expands upon this idea by stating that for a structured mesh of this type, the maximum error in the angle between the mathematical path and the geometrical crack path is between 22.5° and 45°, depending on the number of edges connected to the node. When the Hausdorff

distance is used to minimize the difference between the length of the mathematical path and the path through the mesh, Park (2009) shows a "zig-zag" pattern in the path through the mesh. From Figure 1.1, it is seen that the only instances in which the geometrical path exactly matches the mathematical path occurs when the mathematical path is oriented at $0°, \pm45°$ or $90°$. In all other cases, there will be a non-zero error between the two paths.

While there are few studies which directly test for the behavior described above, numerical testing by Sam (2005); Park (2009) have shown a correlation between the element orientation and crack path behavior. However, testing by Zhang and Paulino (2005) shows less mesh dependence.

A potential solution to the mesh dependence on the crack path has been proposed by Papoulia et al. (2006). Based on the pinwheel tiling described in Radin (1994); Radin and Sadun (1996), Papoulia, et al. utilize a pinwheel mesh which can be subdivided indefinitely to "represent every possible crack path". The group has developed an algorithm for creating such meshes in Ganguly et al. (2006). As any portion of the mesh can be further refined, a mesh which allows for a crack to form along the preferred path can be created. Testing in Papoulia et al. (2006) suggest that the pinwheel mesh can better approximate the crack path than either a structured or unstructured mesh.

According to Papoulia et al. (2006), there are some potential limitations of pinwheel meshes which may reduce the utility of this method. As of the publication of the paper, there was no known method for creation of pinwheel meshes in more than two dimensions. Additionally, testing suggests that the convergence rate of the method may be slow compared to other meshing options.

Another potential solution to the mesh dependence issue has been developed by Park (2009); Paulino et al. (2010). For a dynamic analysis of a structured mesh, a combination of a nodal perturbation and/or edge swap operations can be utilized to reduce the discrepancy between the mathematical and geometrical crack path. A nodal perturbation randomly moves the nodes of a structured mesh such that the mesh is geometrically unstructured. As the topological structure of the mesh is retained, there are the same number of potential crack paths at every node as in the original mesh. With the addition of the perturbation, edges of the elements are no longer constrained to the small number of directions in a structured mesh. In this way, the difference between the mathematical and geometrical crack path are potentially reduced.

In a test of 100 meshes with four different perturbation factors, Paulino et al. (2010)

reports a decrease in the error between the mathematical crack length and the mean geometrical crack length when the perturbation factor is increased. The standard deviation of the error increases with an increase in the perturbation factor, but to a lesser extent than the decrease seen in the mean error.

Instead of a mathematical crack angle of approximately 67° as in the test shown in Figure 1.1, an additional test was conducted by Paulino et al. with a mathematical crack angle of 45°. This value was picked to correspond with the angle of the element edges in the structured mesh with no nodal perturbation. Testing a set of 100 meshes with various nodal perturbation factors as above, the error between the mathematical crack length and the mean geometrical crack length increased with an increased nodal perturbation factor as the geometrical crack path in the meshes with the perturbation factor were unable to exactly follow the mathematical crack path. However, the increase in the error for this test was less than the decrease in the error seen in the test with the mathematical crack path not coincident with the element edges of the unstructured mesh. This test, along with the one above, suggest that some of the benefits of an unstructured mesh can be realized by randomly perturbing the locations of the nodes of a structured mesh.

An edge-swap operator retains the geometric structure of a mesh while creating a topologically unstructured mesh. When a crack reaches a node, the preferred direction of the crack is determined. If the direction is along one of the edges from the node, no change is made. If however there are no edges which are aligned with the crack, the edge-swap operator may swap one of the surrounding edges such that it it aligned with the crack. This reduces the "zig-zag" pattern seen in the structured meshes. As the edge-swap operator does not add new nodes, but instead updates the edges between the nodes, the original geometry relative to the nodes remains constant. Figure 1.2[†] describes the edge-swap operator on a mesh with a crack growing horizontally. Between the first and second steps, one of the edges is swapped so the crack can follow a path better fitting with the stress state in the mesh. The same process is repeated between the second and third steps.

With both the nodal perturbation and edge swap operators, Paulino et al. (2010) found a significant decrease in the error between the mathematical and mean geometric crack lengths on a set of 100 meshes with varying degrees of nodal perturbation as opposed to a nodal perturbation alone. Numerical testing with the edge swap operator and nodal perturbation show a straighter crack path than with a normal structured mesh. Utilizing these two factors, the authors reported less deviation from

---

[†]Reprinted from International Journal for Numerical Methods in Engineering 84, Paulino, G., Park, K., Celes, W., Espinha, R., Adaptive dynamic cohesive fracture simulation using nodal perturbation and edge-swap operators, 1303–1343, Copyright 2010, with permission from John Wiley & Sons. See Appendix C for more information.
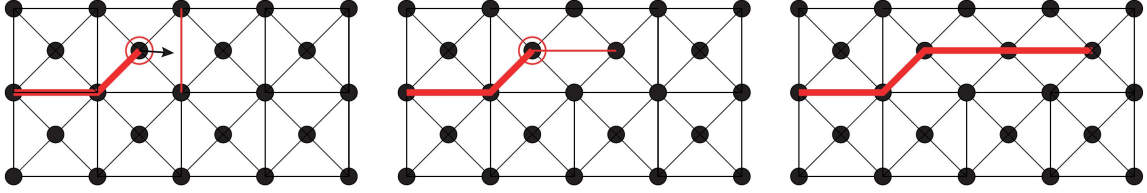
Figure 1.2: The stages of the edge-swap operator on a crack growing horizontally. Reprinted with permission from Paulino et al. (2010). See Appendix C for more information.

the mathematical crack path than using the pinwheel mesh proposed by Papoulia et al. (2006).

A recent work by Brochu and Bridson (2009) details an algorithm capable conducting mesh operations similar to Paulino et al. (2010). The algorithm is capable of an edge swap operation as well as edge splitting and edge deletion. Although not tested with cohesive elements, the algorithm has been successfully tested with meshes undergoing significant mesh deformation, including tearing, without reduction in mesh robustness or accuracy.

While these methods can reduce, and in some cases eliminate, the errors between the predicted and actual crack paths, it is important to note that there are no currently published solutions using the cohesive element method which can exactly follow the preferred crack path under arbitrary loading. While one mesh may agree closely with the predicted crack response, different loading or boundary conditions could result in a significant difference in the crack path error. The response of the system is intimately connected with the discretization used.

1.1.2.2   Adaptive cohesive element insertion

Instead of inserting the cohesive elements into the mesh prior to the analysis, cohesive elements may be added to the mesh as needed during the course of the simulation. This allows for the possibility of a more efficient analysis as only the number of cohesive elements needed are added to the mesh. In *a priori* analyses, a large number of cohesive elements must be added to the mesh indiscriminately to ensure that the fracture behavior is captured. Insertion of cohesive elements into the mesh during analysis is not quite as efficient as expected at first glance because the topological changes to the mesh during insertion add a significant amount of overhead to the program. Ill defined or ad hoc insertion routines also increase the possibility for error in the insertion process.

For this reason, various groups have recognized the need for a more robust data structure for dynamic fracture simulation. For efficient insertion of cohesive elements into a mesh during analysis, as well as any other topological change to the mesh, the entities of the mesh (nodes, edges, faces, and regions) along with the relations between the entities must be known. The traditional datastructures used in finite element analyses utilize a number of arrays which explicitly track only portions of the entities and relationships (nodes, elements, and depending on the particular implementation, adjacent elements). All entities of the mesh may be tracked explicitly (e.g. Mota et al. (2008)), but the more explicitly tracked portions of the mesh, the higher the storage and computational overhead. Conversely, implicitly representing portions of the mesh can introduce added complexity in the datastructure, increasing the likelihood of the introduction of more error prone and ad hoc algorithms. Beall and Shephard (1997) have pointed out some of the issues an implicit representation entail, most of them focusing on the difficulties in modifying portions of the mesh which are implicitly represented (e.g. removing an edge). With implicit representation of mesh data, important aspects such as the nodal ordering of the elements may become ambiguous when the topological changes are made.

Celes, et al. have developed a topological datastructure which attempts to strike a balance between explicitly representing all entities and relationships in the mesh and the traditional datastructure (Celes et al. (2005a,b); Paulino et al. (2008)). They have developed a method named TopS, which explicitly or implicitly represents all aspects of the mesh. In the method, only the nodes and elements are explicitly represented while the element edges and faces along with interfaces and corner nodes are implicitly represented. The datastructure has been developed such that any implicit entity may be determined regardless of the topological changes in the mesh. Using the datastructure, the insertion of new cohesive elements into the mesh scales linearly with the number of insertions.

Dooley et al. (2009) approaches the problem of adaptive insertion in a different manner. Unlike the other methods mentioned in this section, the cohesive elements are not added during the course of the analysis with this method. Instead, Dooley et al. (2009) inserts the cohesive elements along all element edges before the analysis but does not allow for the failure process to begin until the traction across the edge reaches a pre-determined value. When this value is reached, the cohesive element is activated and the failure process may develop at will. This method obviously negates the complexity inherent in adaptive insertion of cohesive elements as all changes to the mesh occur before the analysis begins. The method was developed for usage in parallel computing applications and testing in the paper shows that the method is useful for these problems. However, one of the main benefits of the adaptive insertion method, a reduction in the number of cohesive elements in the mesh, no longer exists.

1.1.2.3   Graph representation

The idea of a finite element mesh being represented by a graph is not new, both Paulino et al. (1994) and Hsieh et al. (1995) utilize graph representations of finite element meshes; Paulino, et al. to optimize nodal ordering and Hsieh, et al. to partition a mesh for parallel computing. These methods, as well as traditional mesh representations, are partial representations of the entire mesh. While some portions of the mesh are explicitly given (e.g. nodal connectivity), others are implicitly stated (e.g. edge numbering).

In contrast to the methods described above, Mota et al. (2008) describes a hierarchal, directed graph based on simplicial complexes which explicitly states all aspects of the mesh (nodes, edges, elements). By representing all aspects of the mesh in the graph, it is possible to modify the mesh (as when a cohesive element is inserted) with less difficulty than when the mesh is represented in the traditional manner.

As opposed to the complex algorithms used to update a mesh when inserting additional elements, the graph representation method accomplishes the same task with relatively little overhead. While storing more information than other methods (e.g. Paulino et al. (2008)), the hierarchal nature of the graph as well as the explicit representation of all mesh objects allow for significantly less complex insertion algorithms. Testing by this group compared the graph representation method with a reference method utilizing a traditional mesh representation as described in Pandolfi et al. (1999); Pandolfi and Ortiz (2002). For a number of 2D meshes with triangular elements, the graph method was slightly slower with a smaller number of elements but faster when the number of elements increased. Fracture of all edges in the meshes were completed in linear time with the graph representation method and in quadratic time with the reference method.

Additionally, the graph representation method was able to accurately handle meshes which are difficult to manipulate with traditional approaches. Two non-manifold meshes, the first of two cubes attached along a common edge and the second at a common point, were split along all element edges. The reference method was unable to do the splitting process correctly while the graph representation method completed the task with no error.

1.1.2.4   eXtended Finite Element Methods (XFEM)

While the discussion thus far has centered on cohesive elements for the simulation of fracture, it is by no means the only possible method. While the cohesive element method has been developed such that the cohesive surface falls on the edges of the volumetric elements of the mesh, methods such as the partition of unity method (Melenk and Babuska (1996); Wells and Sluys (2001)) allow for cracks which pass through the middle of the volumetric element. In this method and its successors, the crack is modeled as an enrichment field in the element, adding additional degrees of freedom at the nodes of the element. The extended finite element method (XFEM) is one of the most popular of these methods (Belytschko and Black (1999); Moës et al. (1999)). The XFEM method allows for cracks to pass through the test specimen without being constrained to the edges of the volumetric elements as in the conventional cohesive element method. The method has yielded results similar to experimental testing in a variety of well known dynamic fracture cases (Song et al. (2006b,a) among others).

While early attempts were not able to model branching, recent work with methods based on the partition of unity method have reduced the advantage cohesive element methods have had in this area (Song et al. (2006b); Linder and Armero (2009); Armero and Linder (2009)).

A number of recent works in XFEM have focused on the introduction of strong discontinuities into the volumetric elements of the finite element mesh. Based on the work of Hansbo and Hansbo (2004) and expanded by Mergheim et al. (2005); Linder and Armero (2007); Rabczuk et al. (2008); Armero and Linder (2009); Linder and Armero (2009); van der Meer and Sluys (2009); Linder et al. (2011) among others, these methods allow for an discontinuity to pass within the volumetric element negating many of the detriments of the conventional cohesive element methods. The methods by Mergheim et al. (2005) and van der Meer and Sluys (2009) in particular are quite close to the method proposed by Maiti et al. (2009) in that the three methods introduce a cohesive surface on the interior of the volumetric element. The original volumetric element is split into two child elements superimposed with the original area split between the child elements. In recent literature, these methods are referred to as phantom node methods.

1.2   Motivation

The discussion in the literature review section suggests possible limitations in the currently published work. First, the cohesive element method, as currently developed, is unable to accurately follow the predicted crack path under some mesh discretizations. Further Papoulia et al. (2006) shows that in some situations, a structured mesh is unable to exactly follow the predicted crack regardless of the discretization of the mesh. With the inclusion of the recent work by Park (2009) suggests that this issue is not constrained to structured meshes but is applicable to unstructured meshes as well.

Work by Maiti et al. (2009), based upon that of Hansbo and Hansbo (2004), proposes a solution to the crack path problem. By embedding the crack surfaces in the volumetric elements of the mesh, the crack can grow in an arbitrary manner independent of the underlying mesh. This method has the advantages of the XFEM methods (e.g. cracks on the interior of volumetric elements) while retaining the simplicity and ease of use of the cohesive element methods.

Additionally, a number of groups have implemented methods for the insertion of cohesive elements into a mesh during the course of a finite element analysis, among them Paulino et al. (2008); Mota et al. (2008). These groups have shown efficient insertion of cohesive elements, with both groups able to insert a cohesive element in linear time, but they do not solve the central problem of the cohesive element methods: the crack is constrained to the boundaries of the volumetric elements. However, these groups have shown that insertion of cohesive elements into a finite element mesh can be done in a robust and efficient manner. In the XFEM realm, there are no currently published algorithms which show the same computationally efficient insertions.

1.3   Research objectives

From the Literature and Motivation sections, two main goals have been identified for this work:

1. Develop a robust method for the insertion of strong discontinuities into a finite element mesh during the course of an analysis. While applicable to a number of XFEM implementations based on the work of Hansbo and Hansbo (2004), the method will be developed using the GCE method described by Maiti et al.

(2009). This system shall allow for,

(a) Arbitrary GCE insertion. Allow for an arbitrary number of GCEs to be inserted into the finite element mesh. These elements may be oriented at will inside of the volumetric elements.

(b) Data management of large meshes. Expanding on the work of Mota et al. (2008) and modifying for the usage of GCEs instead of cohesive elements, develop a graph based mesh representation method.

(c) Parallel insertion of GCEs in a finite element mesh. Develop a sub-graph representation, such that the insertion process is independent of the other volumetric elements.

2. Show the utility of GCE for mesh independent crack propagation. While it has been shown for quasi-static analysis, the GCE method is untested for dynamic problems. Its utility, both in *a priori* and adaptive cases shall be shown.

1.4   Thesis organization

Chapter 2 shall develop the numerical framework to study dynamic finite element analyses using adaptively inserted GCEs. The main areas of the framework shall be discussed in detail. First the volumetric element stress is found followed by the cohesive element tractions. The GCE method is compared with the conventional cohesive element method. The majority of the chapter is devoted to development of the insertion algorithm. A short description of the simplicial graph mesh representation used for this work is given. Finally, it is shown that the insertion algorithm, in conjunction with the GCE method, can accurately deal with the edge cases of the GCE method. These include GCE insertion along a volumetric element edge or node.

Numerical results are discussed in Chapter 3. A number of experiments have been devised to show the utility of the GCE insertion scheme developed in Chapter 2. First, the ability to arbitrarily insert GCEs *a priori* into a mesh is shown. Next is a discussion on two competing methods for the insertion process. These are what we have called the full graph and sub graph representations. It is shown that the sub graph method is significantly faster than the full graph method. In the third test, the adaptive insertion routine is tested by inserting a cohesive line into a mesh during an analysis. The GCE and conventional cohesive element methods are compared in the fourth test, showing a strong agreement between the two methods. The importance of a scaling factor in the insertion criteria is also discussed in this test. Finally, another

test of the GCE method is conducted utilizing a ballistic test widely reported in computational fracture literature.

In Chapter 4, the thesis concludes with a review of the work developed herein as well as the salient findings. Limitations of the current work are discussed and future research opportunities in this area are suggested.

Chapter 2

Numerical Methodology

This chapter shall detail the pertinent sections of the finite element code used in this work. The chapter shall start with a brief discussion of the finite element formulation. Following this is an overview of the volumetric element routine. A detailed look at the GCE formulation will be given in Section 2.2. The chapter will conclude with a discussion of the insertion routine.

Some notes on the mathematical notation used in this chapter have been listed below.

1. Unless otherwise stated, non-bolded variables are scalar. Bolded terms (e.g. $\mathbf{N}$, $\mathbf{v}$) are vectors, matrices, or tensors.

2. A matrix or vector transpose is written as $(\cdot)^T$ and a matrix inverse by $(\cdot)^{-1}$. The trace of a matrix is denoted by $\mathrm{tr}(\cdot)$.

3. Regions shall be denoted by Ω and their boundary by either $\partial\Omega$ or Γ.

4. While general explanations shall be given in the main text, simplices and simplicial complexes, used to develop the GCE insertion routine, are discussed in greater detail in Appendix A.

5. Following convention, the nodes of the finite elements are ordered in a counter clockwise manner. Additionally, the ordering of the edges of the elements (i.e. the line segments) is defined as opposite the nodes. Thus edge 1 of an element is opposite node 1.

6. Voigt notation is used throughout to reduce the order of symmetric second order tensors to first order tensors. As an example, Equation 2.1 shows the Cauchy stress tensor in matrix form and Voigt form.

$$\sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{12} & \sigma_{22} & \sigma_{23} \\ \sigma_{13} & \sigma_{23} & \sigma_{33} \end{bmatrix} \rightarrow \left\{ \sigma_{11} \quad \sigma_{22} \quad \sigma_{33} \quad \sigma_{23} \quad \sigma_{13} \quad \sigma_{12} \right\}^T \qquad (2.1)$$

2.1 Finite element formulation

The formulation used for this work derives from the following principal of virtual work,

$$\int_{\Omega_0} \mathbf{S} : \delta\mathbf{E} \, d\Omega_0 + \int_{\Omega_0} \rho_0 \ddot{\mathbf{d}} \cdot \delta\mathbf{d} \, d\Omega_0 - \int_{\Gamma_0^{ex}} \mathbf{T}_{ex} \cdot \delta\mathbf{d} \, d\Gamma_0^{ex} - \int_{\Gamma_0^{in}} \mathbf{T} \cdot \delta\boldsymbol{\Delta} \, d\Gamma_0^{in} = 0 \quad (2.2)$$

where Ω denotes the domain, Γ_0^{ex} the boundary of the body, and Γ_0^{in} the interior boundaries along the cohesive elements. $\mathbf{T}_{ex}$ is the external traction and $\mathbf{T}$ is the cohesive traction due to a virtual separation. $\mathbf{S}$ and $\mathbf{E}$ are the 2nd Piola-Kirchhoff stress tensor and Green strain tensor, respectively. $\boldsymbol{\Delta}$ denotes the displacement jump across the cohesive surface while $\mathbf{d}$ and $\ddot{\mathbf{d}}$ denote the displacement and acceleration vectors. This formulation is discretized in the initial (i.e. Lagrangian) configuration.

This method utilizes an explicit, central difference, time stepping scheme to determine the displacement ($\mathbf{d}$), velocity ($\mathbf{v} = \dot{\mathbf{d}}$), and acceleration ($\mathbf{a} = \ddot{\mathbf{d}}$) vectors at each time step. For each timestep of the analysis the displacements, velocities, and accelerations of the nodes for the next timestep are determined using Algorithm 1.

---

for $n = 0$ to n_{max} **do**
  Compute $\mathbf{R}_{in}$ and $\mathbf{R}_{co}$
  $\mathbf{d}^{(n+1)} \leftarrow \mathbf{d}^{(n)} + \Delta t \mathbf{v}^{(n)} + \frac{1}{2}\Delta t^2 \mathbf{a}^{(n)}$
  $\mathbf{a}^{(n+1)} \leftarrow -\mathbf{M}^{-1}\left(\mathbf{R}_{in}^{(n+1)} - \mathbf{R}_{co}^{(n+1)}\right)$
  $\mathbf{v}^{(n+1)} \leftarrow \mathbf{v}^{(n)} + \frac{\Delta t}{2}\left(\mathbf{a}^{(n+1)} + \mathbf{a}^{(n)}\right)$
  Impose boundary conditions
end for

Algorithm 1: Explicit time stepping scheme

n is the step number while the superscripts in parentheses refer to the step. $\mathbf{R}_{in}$ is the global internal force vector, and $\mathbf{R}_{co}$ is the global cohesive force vector. Δt is the time step size. $\mathbf{M}$ is the lumped mass matrix.

The local internal force vector for an individual element is found through,

$$\mathbf{R}_{\text{in}}^e = \int_{\Omega^e} \mathbf{B}_I^{0T} \mathbf{S} \, \mathrm{d}\Omega^e \tag{2.3}$$

where the area of the element is Ω^e, and $\mathbf{B}_I^0$ is the strain displacement matrix. $\mathbf{S}$ is the 2nd Piola-Kirchhoff stress tensor written in Voigt notation. The steps for determining $\mathbf{R}_{\text{in}}^e$ are given in Section 2.2.

The local cohesive force vector is given by,

$$\mathbf{R}_{\text{co}}^e = \int_{\Gamma^e} \mathbf{N}^T \mathbf{T} \, \mathrm{d}\Gamma^e \tag{2.4}$$

where $\mathbf{N}$ is an array of the shape functions, $\mathbf{T}$ is the cohesive traction, and Γ^e is the length of the cohesive element. Section 2.3 describes the process used to find $\mathbf{R}_{\text{co}}^e$.

The global internal force and cohesive force vectors are constructed by summing the local vectors over all the volumetric and cohesive elements (e.g. $\mathbf{R}_{\text{in}} = \sum_{i=1}^{numel} \mathbf{R}_{\text{in}}^i$).

2.2 Volumetric element formulation

The internal forces in the body are caused by the deformations of the volumetric elements. To account for large deformations and rotations which may occur during the analysis, a non-linear strain model is used.

The deformation gradient tensor $\mathbf{F}$ is defined as,

$$\mathbf{F} = \frac{\partial \mathbf{x}}{\partial \mathbf{X}} \tag{2.5}$$

where $\mathbf{X}$ is the particle in the reference configuration and $\mathbf{x}$ is the particle in the current configuration. The right Cauchy-Green deformation tensor $\mathbf{C}$ is defined as,

$$\mathbf{C} = \mathbf{F}^T \mathbf{F} \tag{2.6}$$

As stated above, the potential of large deformations and rotations necessitate the use of a non-linear strain measure. For this reason, the Lagrangian strain tensor $\mathbf{E}$ is

17

used. This tensor is defined as,

$$E = \frac{1}{2}\left(C - I\right) = \frac{1}{2}\left(F^T F - I\right) \tag{2.7}$$

with I denoting the identity tensor.

The Green strain tensor may alternatively be written in terms of the displacement gradients as opposed to the deformation gradient. The deformation is defined as,

$$u = x - X \tag{2.8}$$

Taking the partial derivative of this equation with respect to the undeformed coordinates X,

$$\frac{\partial u}{\partial X} = \frac{\partial x}{\partial X} - I \tag{2.9}$$

Thus, an alternative formulation of the deformation gradient tensor is,

$$F = \frac{\partial x}{\partial X} = \frac{\partial u}{\partial X} + I \tag{2.10}$$

Substituting this into (2.7) and simplifying, an alternative form of E is given as,

$$E = \frac{1}{2}\left(\frac{\partial u}{\partial X} + \frac{\partial u}{\partial X} + \frac{\partial u}{\partial X}\frac{\partial u}{\partial X}\right) \tag{2.11}$$

With the strain defined, the stress in the element is found. The second Piola-Kirchhoff stress tensor S may be found through,

$$S = \frac{\partial W(E)}{\partial E} \tag{2.12}$$

with $W(E)$ defined as the strain-energy density function of the material.

While many different material models may be used to determine the stress in the element, the current work utilizes an isotropic St. Venant-Kirchhoff material model. For this material, the strain energy function is given by,

$$W\left(E\right) = \frac{\lambda}{2}\left(\text{tr}E\right)^2 + \mu \text{tr}E^2 \tag{2.13}$$

The variables λ and μ in the above equation are the two Lamé constants. These constants may be related to the more commonly used material properties: Young's

modulus (E) and Poisson's ratio (ν) through,

$$\lambda = \frac{\nu E}{(1+\nu)(1-2\nu)}, \qquad \mu = \frac{E}{2(1+\nu)} \tag{2.14}$$

Substituting Equation (2.13) into Equation (2.12) and taking the partial derivative with respect to the Green strain $\mathbf{E}$, the constitutive relationship between $\mathbf{E}$ and $\mathbf{S}$ is found.

$$\mathbf{S} = \lambda \operatorname{tr}(\mathbf{E})\mathbf{I} + 2\mu\mathbf{E} \tag{2.15}$$

The remaining variable from Equation (2.3) is $\mathbf{B}_I^0$. This matrix is defined in Belytschko et al. (2000) as,

$$B_{ijkI}^0 = \operatorname{sym}\left(\frac{\partial N_I}{\partial X_i}F_{jk}\right) \tag{2.16}$$

The above matrix however is not conductive to computational calculations. Therefore, the same reference rewrites the matrix in a more convenient form. For a 2-D triangular element, the matrix becomes,

$$\mathbf{B}_I^0 = \begin{bmatrix} \frac{\partial N_I}{\partial X}\frac{\partial x}{\partial X} & \frac{\partial N_I}{\partial X}\frac{\partial y}{\partial X} \\ \frac{\partial N_I}{\partial Y}\frac{\partial x}{\partial Y} & \frac{\partial N_I}{\partial Y}\frac{\partial y}{\partial Y} \\ \frac{\partial N_I}{\partial X}\frac{\partial x}{\partial Y} + \frac{\partial N_I}{\partial Y}\frac{\partial x}{\partial X} & \frac{\partial N_I}{\partial X}\frac{\partial y}{\partial Y} + \frac{\partial N_I}{\partial Y}\frac{\partial y}{\partial X} \end{bmatrix} \tag{2.17}$$

The subscript I ranges over the nodes of the element. Thus, for a three noded element, $\mathbf{B}_I^0$ is a 6×3 matrix.

Gaussian quadrature is used to evaluate the integral from Equation (2.3). In this form, the equation becomes,

$$\mathbf{R}_{\text{in}}^e = A\sum_{I=1}^{n} w_I \left(\mathbf{B}_I^0\right)^T \mathbf{S} \tag{2.18}$$

with A denoting the area of the element and n the number of nodes. For triangular elements, the area is evaluated as,

$$A = \frac{1}{2}(x_{21}y_{31} - x_{31}y_{21}), \quad x_{ij} = x_i - x_j \tag{2.19}$$

where x_{ij} is the coordinate of the triangle node in the reference configuration.

The stress in Equation (2.18) is the 2nd Piola-Kirchhoff stress tensor written in Voigt notation (i.e. $\mathbf{S} = \{S_{11}, S_{22}, S_{12}\}$). The Gauss weights, w_I, are the same as those listed in Baylor (1997).

This procedure described in this section is shown in pseudo-code in Algorithm 2.

Input: $\mathbf{x}, \mathbf{X}$
Output: $\mathbf{R}_{\text{in}}$
  for $i = 1$ to *numel* **do**
    {Loop over all elements}
    $\mathbf{u} \leftarrow \mathbf{x}_i - \mathbf{X}_i$
    $\mathbf{E} \leftarrow \frac{1}{2}\left((\nabla_0\mathbf{u})^T + \nabla_0\mathbf{u} + \nabla_0\mathbf{u}\cdot(\nabla_0\mathbf{u})^T\right)$
    $\mathbf{S} \leftarrow \lambda\,\text{tr}\,(\mathbf{E})\,\mathbf{I} + 2\mu\mathbf{E}$
    $\mathbf{R}_{\text{in}}^i \leftarrow A\sum_{I=1}^{3} w_I(\mathbf{B}_I^0)^T\mathbf{S}$
    $\mathbf{R}_{\text{in}} \leftarrow \mathbf{R}_{\text{in}} + \mathbf{R}_{\text{in}}^i$
  end for

Algorithm 2: Internal force

The variable *numel* in Algorithm 2 is equal to the total number of elements in the finite element mesh.

2.3   Cohesive element formulation

Like the volumetric element formulation, the purpose of this section is to relate the deformation of the elements to the forces at the nodes. In Chapter 1, it was stated that the generalized cohesive element (GCE) method is derived from the conventional cohesive element (CE). When the cohesive surface of the GCE is on the boundary, the GCE and the CE are functionally identical. In practice, it is unlikely that any but a small fraction of the cohesive surfaces of GCEs will be be situated in this manner. More commonly, the cohesive surface will pass through the interior of the volumetric elements which comprise the GCE. As the GCE is based upon the CE, the procedure for determining R_{co} for a CE will be covered first. After this, the same process using a GCE will be developed.

2.3.1 CE formulation

The traction-separation law used for this work was introduced by Geubelle and Baylor (1998) and has been used in a number of studies since including Maiti and Geubelle (2004). This is a bilinear rate-dependant and damage-dependant cohesive model.

Consider two volumetric elements allowed to move independent of each other which are initially in contact with each other. The surfaces of the elements which are in contact with each other will be denoted by $\Gamma^{(+)}$ and $\Gamma^{(-)}$. As the two elements move through space, we would like to know the location of the initially contacting surfaces. This may be done by tracking the opening Δ at each node of the surfaces. If there is a force which connects the two surfaces of Γ, the elements may no longer move independently of each other. Instead, the movement of one element will affect the movement of the other. However as the elements move apart from each other, the force between the elements decreases until finally there is no connection between the surfaces. A cohesive element with an initial area of zero will pass the entire traction at the interface of the volumetric elements between the elements until the openings at the cohesive element begin to grow. At that time, the tractions passed between the volumetric elements decrease until a critical opening is reached at which point the traction becomes zero.

A representative cohesive element under loading is shown in Figure 2.1. Δ_n is the opening in the normal direction while Δ_t is the opening in the tangental direction. As the opening between the nodes of the cohesive element grow, the traction decreases.
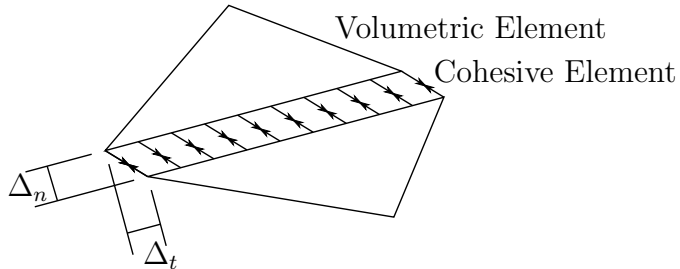
Figure 2.1: A cohesive element is situated at the interface of two volumetric elements.

The traction-separation laws as developed by Geubelle and Baylor (1998) for the normal and shear directions are given by,

$$T_n = \frac{S}{1-S}\frac{\Delta_n}{\Delta_{nc}}\frac{\sigma_{max}}{S_{init}}, \qquad T_t = \frac{S}{1-S}\frac{\Delta_t}{\Delta_{tc}}\frac{\tau_{max}}{S_{init}} \tag{2.20}$$

where S is the damage parameter, S_{init} is the initial damage and σ_{max} and τ_{max} are

the cohesive strengths in normal and shear directions, respectively. Δ_n and Δ_t are the openings in the normal and shear directions and Δ_{nc} and Δ_{tc} are the critical openings in the same directions.

The damage parameter S is in the range $S \in [0, 1]$ with $S = 1$ denoting no damage and $S = 0$ denoting a fully fractured cohesive surface. To ensure a monotonic decrease in S (no healing of the cohesive surface), S is calculated by,

$$S = \min\left(S_{min}, \left\langle 1 - ||\bar{\mathbf{\Delta}}|| \right\rangle\right) \tag{2.21}$$

S_{min} is the minimum value of S at each Gauss point and $|| \cdot ||$ is the Euclidean norm. $\langle \cdot \rangle$ is defined as,

$$\langle x \rangle = \begin{cases} x, & x > 0 \\ 0, & \text{else} \end{cases} \tag{2.22}$$

In Equation (2.21), $\bar{\mathbf{\Delta}}$ is the normalized displacement jump vector defined as,

$$\tilde{\mathbf{\Delta}} = \begin{bmatrix} \tilde{\Delta}_n \\ \tilde{\Delta}_t \end{bmatrix} = \begin{bmatrix} \Delta_n/\Delta_{nc} \\ \Delta_t/\Delta_{tc} \end{bmatrix} \tag{2.23}$$

Δ_{nc} and Δ_{tc} may be found through,

$$\Delta_{nc} = \frac{2G_{Ic}}{\sigma_{max}S_{init}}, \qquad \Delta_{tc} = \frac{2G_{IIc}}{\tau_{max}S_{init}} \tag{2.24}$$

where G_{Ic} and G_{IIc} are the mode I and II fracture toughnesses, respectively.

Figure 2.2 plots the traction-separation relationship for cohesive elements loaded purely in the normal (Figure 2.2(a)) and shear (Figure 2.2(b)) directions.

The local cohesive force vector may then be found using Equation (2.4).

2.3.2 GCE formulation

A GCE is similar to a conventional cohesive element in that the traction across the cohesive surface decreases as the opening increases. Instead of being separate from the volumetric elements of the mesh as in the conventional cohesive element formulation, the GCE is embedded within the volumetric element which it passes through. A
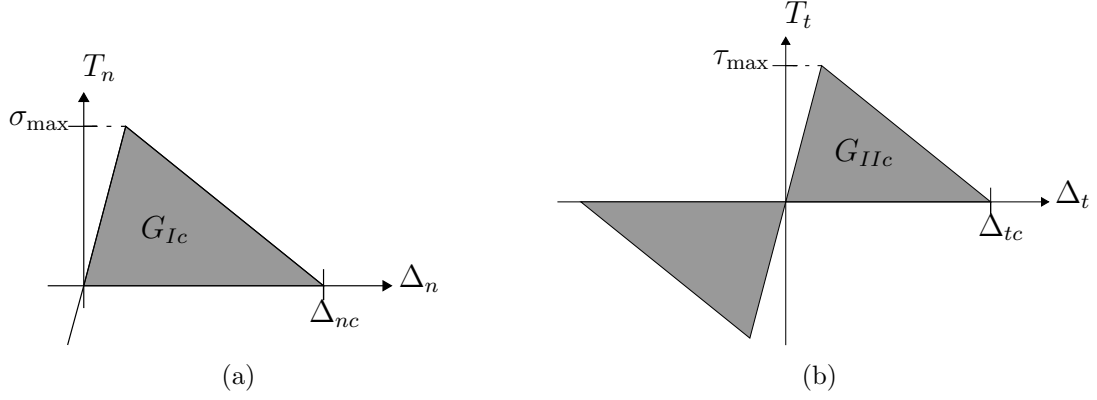
Figure 2.2: The cohesive law in pure normal and pure shear loading

volumetric element through which the cohesive surface will pass is split into two child elements as in Figure 2.3[*].
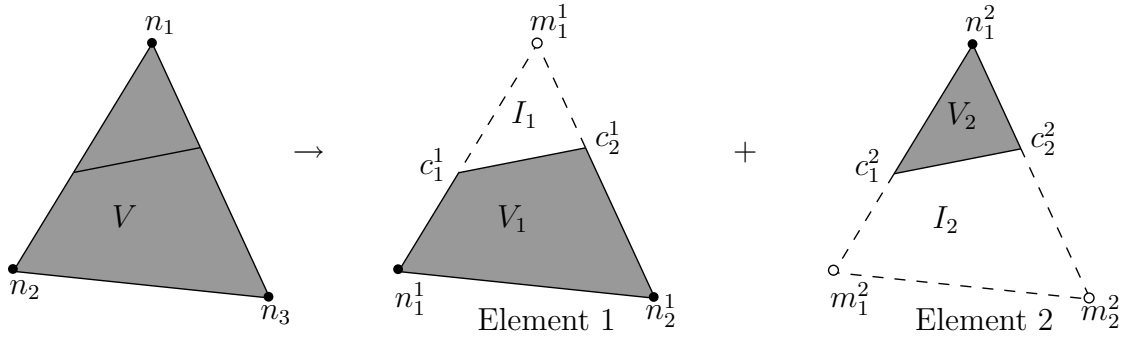
Figure 2.3: A GCE splits a volumetric element with an embedded cohesive surface into two child elements. Figure reproduced with permission from Maiti et al. (2009). See Appendix C for more information.

At insertion, the volume of the original element is split between the child elements along the embedded cohesive surface. The original area of the element is conserved in the child elements by adding a virtual volume equal to the volume removed to each child element. From the figure, the original volume is related to the volume of the child elements by $V = V_1 \cup V_2$. For each of the child elements, the volume of the element with the addition of the virtual volume is $V = V_i \cup I_i$, $i = 1, 2$. At insertion, the original nodes of the volumetric element are split between the child elements (solid circles n_j^i) while virtual nodes (open circles m_j^i), associated with the virtual areas complete the elements. The original nodes retain the original connectivity to the surrounding mesh while the virtual nodes are unattached. The endpoints of the cohesive surface, c_j^i are not explicitly tracked in the mesh topology, but are instead implicitly tracked using

[*]Reproduced from Finite Elements in Analysis and Design, 45 (89), Maiti, S., Ghosh, D., Subhash, G., A generalized cohesive element technique for arbitrary crack motion, 501–510, Copyright 2009, with permission from Elsevier. See Appendix C for more information.

barycentric coordinates associated with the undeformed element.

The shape functions used to compute the internal force vectors for the child elements are constructed such that,

$$N_i^1 = \begin{cases} N_i & \text{in } V_1 \\ 0 & \text{in } I_1 \end{cases} \quad \text{and} \quad N_i^2 = \begin{cases} 0 & \text{in } I_2 \\ N_i & \text{in } V_2 \end{cases} \tag{2.25}$$

where N_i are the same shape functions used in the non-fractured volumetric elements.

The endpoints of the cohesive surface of the GCe are not explicitly tracked but are found through interpolation from the nodes of the elements. To compute the cohesive tractions on the cohesive surface of the GCE, the nodal displacements of the volumetric element are interpolated to the cohesive surface endpoints through the barycentric coordinates computed during the insertion procedure. The tractions are then determined using the same manner as the conventional cohesive elements as described above. Finally, the tractions at the cohesive surface endpoints are extrapolated back to the nodes of the volumetric element again using the barycentric coordinates of the cohesive endpoints.

From the following discussion, the procedure to find the cohesive tractions in the GCEs of the mesh is given in Algorithm 3. The variable *numelc* in that algorithm is the number of cohesive elements (GCE in this case) which are in the mesh. As for the volumetric element calculations, the force vector is computed using a Gaussian Quadrature on the integral.

Input: $\mathbf{x}, \mathbf{X}$
Output: $\mathbf{R}_{\text{co}}$
   for $i = 1$ to *numelc* **do**
      {Loop over all GCE}
      Interpolate the nodal displacements to the cohesive endpoints
      $\tilde{\Delta}_n \leftarrow \Delta_n / \Delta_{nc}$
      $\tilde{\Delta}_t \leftarrow \Delta_t / \Delta_{tc}$
      $S \leftarrow \min(S_{\min}, < 1 - ||\tilde{\mathbf{\Delta}}|| >)$
      $T_n \leftarrow \frac{S}{1-S} \tilde{\Delta}_n \frac{\sigma_{max}}{S_{init}}$
      $T_t \leftarrow \frac{S}{1-S} \tilde{\Delta}_t \frac{\tau_{max}}{S_{init}}$
      $R_{co}^i \leftarrow \int \mathbf{N}^T \mathbf{T} \, \mathrm{d}\Gamma^i$
      Extrapolate traction to volumetric element nodes
   end for

Algorithm 3: Cohesive traction

2.4 GCE insertion formulation

The insertion of a GCE into the finite element mesh is, as described previously, dependent on the idea of simplicial complexes and graph theory. The formulation of these GCEs reduce the difficulties in the insertion process. Instead of being explicitly added into the mesh, the endpoints of the cohesive surface of the GCE are interpolated from the nodes of the volumetric elements of the GCE. These endpoints are determined at insertion on the undeformed element and stored as sets of barycentric coordinates. There are no additional nodes along the edges of the volumetric elements of the GCE and thus there is no difference how these elements are represented in the mesh as compared to the remainder of the elements.

In addition to this, the insertion of a GCE as opposed to a CE is simplified due to the inherent locality of the operation. When a CE is inserted into a mesh, the insertion process updates a number of objects in the mesh (nodes, segments and elements). The identities of the objects which require modification are given by the star of the graph vertex which describes the segment along which the CE will be oriented. The star $\mathrm{st}(\sigma)$ of a simplex $\sigma \in G$ is defined as the union of the interiors of the simplices in G which have σ as an edge. This results in an induced subgraph $G' \subset G$ which includes the simplex σ and all simplices in contact with σ, excluding the boundary of the set.

In contrast, the insertion of a GCE requires an update only to the original parent element and the segments and nodes thereof. To localize the process, instead of conducting the insertion on a graph representing the entire topology of the mesh as in Mota et al. (2008), the insertion process for this work manipulates a sub-graph consisting of only the simplex describing the element to be modified. As all elements in the mesh will be the same (i.e. n-noded simplices), the subgraph on which the insertion procedure works is the same for all insertions. This point is of particular importance in the insertion process. With this, a generic graph, representing any element in the mesh may be used instead of a number of different graphs representing the multitude of possible mesh topologies about the point of insertion. The advantage of using the sub-graphing method as opposed to a full mesh representation is shown in Section 3.3.

Between the volumetric and cohesive routines, the stress state in each volumetric element is checked to determine whether the insertion criterion is met. If so, a GCE is dynamically inserted into the mesh. This process is done in two steps. First, the orientation of the cohesive surface and the endpoints of the surface on the boundary of the volumetric element are found. After this, the new GCE is inserted. The original

volumetric element is split into two child elements and a cohesive surface is inserted. This step also updates all relevant arrays including the mesh connectivity.

There are a wide array of criteria which could be used to determine both the optimal time of insertion and the orientation of the surface of the GCE. For this work, the well known maximum principal stress criterion shall be used.

In general, a new GCE should be inserted into the finite element mesh when the stress state in the parent element approaches the cohesive strength of the element material. The insertion criterion will therefore be fulfilled when,

$$S^e \geq \alpha S^e_{coh} \tag{2.26}$$

The scalar multiplier $\alpha \in [0, 1]$ controls what the ratio between the stress in the element and the cohesive strength of the material will be before the GCE is inserted. Proper choice of this multiplier appears to be an important parameter if the insertion is to be numerically stable. This problem as well as some potential choice for α will be described in Chapter 3.

The value of S^e is determined using the maximum principal stress formulae. For 2-D plane stress, the principal stresses, S_p, are equal to,

$$S_p = \frac{S_{11} + S_{22}}{2} \pm \sqrt{\left(\frac{S_{11} - S_{22}}{2}\right)^2 + S_{12}^2} \tag{2.27}$$

S_{11}, S_{22}, S_{12} are the elements of the second Piola-Kirchhoff stress tensor found in Equation (2.15). This equation yields two values, the maximum of which is set equal to S^e.

$$S^e = \max\left(S_p(1), S_p(2)\right) \tag{2.28}$$

As stated above, insertion occurs if the insertion criterion is satisfied. If this occurs, two sub-routines are called. The first determines the orientation and location of the cohesive surface and the second inserts the GCE into the mesh, updating the variables as necessary. The pseudo-code for this procedure is given in Algorithm (4). **lm** is the connectivity array of the mesh while **S** is the 2nd Piola-Kirchhoff stress described in Section 2.2.

```
Input: S, lm
Output: lm
    for i = 1 to numel do
        Sp ← (S₁₁ + S₂₂) /2 ± √((S₁₁ − S₂₂)/2)² + S₁₂²
        Sᵉ ← max(Sₚ)
        if Sᵉ ≥ αS_coh then
            GCE_ORIENTATION
            GCE_INSERT
        end if
    end for
```

Algorithm 4: Insertion criterion

2.4.1 GCE orientation

If S^e satisfies the insertion criterion in Equation (2.26), the orientation of the cohesive surface is found. The principal direction for a 2-D plane stress state is given by,

$$\theta = \frac{1}{2} \tan^{-1} \left(\frac{2S_{12}}{S_{11} - S_{22}} \right) \tag{2.29}$$

This equation gives the orientation of the cohesive surface, but does not define the location. For this, there are two separate cases. If the element which is to have a GCE inserted is adjacent to another element with a preexisting GCE, it is said to propagate the cohesive surface. If there are no adjacent GCEs, then the element is said to nucleate a cohesive surface. Nucleation sites pass the cohesive surface through the centroid of the volumetric element oriented along the direction θ. Elements which propagate an existing cohesive surface place one cohesive surface endpoint on the endpoint of the previous cohesive surface and again are oriented along the direction θ.

Thus, there are two cases which must be considered when determining the location of the cohesive surface in the GCE. In both cases, two pieces of information about the cohesive surface are known. The direction of the cohesive surface for both cases is determined by the stress state in the element. In addition, one point on the cohesive surface is known. In the nucleation case, this point is the centroid of the element. In the propagation case, the point is the endpoint of the cohesive surface for the previous GCE. As described in the GCE formulation section, the location of the cohesive surface in the GCE is determined by its endpoints on the edges of the volumetric element. The problem therefore is to determine at which points the cohesive surface intersects the edges of the volumetric element. For this, a line intersection algorithm is needed.

A search of the literature reveals a number of potential methods for determining the intersection points of line segments some examples of which may be seen in Antonio (1992), Shaffer and Feustel (1992), and Maiti (2002). While the three algorithms listed above yield fast and robust results, a method based upon the algorithm from Antonio (1992) was most easily adapted to the needs of this application.

The final algorithm is computationally efficient, checking intersection and determining the intersection point with a single division. For comparison, this method uses the same number of divisions, but slightly fewer operations that the similar method used in Maiti (2002).

The two lines to be tested are represented by two points each. The equations of these lines may then be written as,

$$P_a = P_1 + \lambda_a (P_2 - P_1)$$
$$P_b = P_3 + \lambda_b (P_4 - P_3) \tag{2.30}$$

where $P_i = (x_i, y_i)$ in the reference configuration. The variables λ_a, λ_b are shape functions along lines a and b respectively, with $\lambda_a, \lambda_b \in [0, 1]$. A value of 0 corresponds to a point situated on the first endpoint of the line (P_1 on a and P_3 on b) while a value of 1 corresponds to a point situated on the second point.

Intersection of lines a and b occurs when $P_a = P_b$. Solving for this point yields two equations,

$$\lambda_a = \frac{(x_4 - x_3)(y_1 - y_3) - (y_4 - y_3)(x_1 - x_3)}{D}$$
$$\lambda_a = \frac{(x_2 - x_1)(y_1 - y_3) - (y_2 - y_1)(x_1 - x_3)}{D} \tag{2.31}$$

The denominator D in the above two equations is equal to,

$$D = (y_4 - y_3)(x_2 - x_1) - (x_4 - x_3)(y_2 - y_1) \tag{2.32}$$

Intersection occurs if both $\lambda_a \in [0, 1]$ and $\lambda_b \in [0, 1]$. If this is not the case, then the intersection of the lines lie outside of the segment boundaries. There are two special cases of these equations. If $D = 0$, line a and line b are parallel. If the numerators of zero in addition to $D = 0$, then the lines are coincident.

From λ_a or λ_b, the intersection point is given as a ratio of the length of the segment (i.e. the barycentric coordinate of the point on the line). If needed, the intersection

point in the global coordinate system may be found by substituting λ_a or λ_b into the equation of the line.

$$\left\{ \begin{array}{c} x \\ y \end{array} \right\} = \left\{ \begin{array}{c} x_1 \\ y_1 \end{array} \right\} + \lambda \left\{ \begin{array}{c} x_2 - x_1 \\ y_2 - y_1 \end{array} \right\} \tag{2.33}$$

The pseudo-code for this algorithm is given in Algorithm 5.

Input: P_1, P_2, P_3, P_4
Output: P, λ
  $D \leftarrow (y_4 - y_3)(x_2 - x_1) - (x_4 - x_3)(y_2 - y_1)$
  $N \leftarrow (x_4 - x_3)(y_1 - y_3) - (y_4 - y_3)(x_1 - x_3)$
  if $D == 0$ **then**
    if $N \neq 0$ **then**
      {Lines are parallel}
    else
      {Lines are coincident}
    end if
  else
    $\lambda \leftarrow N/D$
    if $\lambda \in [0, 1]$ **then**
      {Lines intersect}
    else
      {Lines do not intersect}
    end if
  end if

Algorithm 5: Line intersection

This algorithm is run for all edges of the volumetric element to determine which edges of the volumetric element intersect with the cohesive surface. In the line intersection algorithm, the two endpoints of the line segments to be tested are needed. The endpoints of the edge of the volumetric element are trivial to determine, but the known information of the cohesive surface is one point and a direction. Basic geometry will yields another point on the line in the vicinity of element, but neither points are guaranteed to be on the edges of the volumetric element nor are they either guaranteed to be exterior to the element. To allow for a robust line intersection check, the code does not check for intersection of two line segments, but instead of a line segment and an infinite line. The segment of the volumetric element edge is represented as a line segment while the cohesive surface is represented as a infinite line. As the intersection point on the element edge is required, not the intersection on the cohesive surface, this

does not affect the end result. Instead of checking λ for both lines, it is only checked for the line segment. This change allows for an accurate intersection check regardless of the endpoints of cohesive surface which are unknown until the intersection points on the edges of the element are found.

From the line intersection algorithm, the orientation of the cohesive surface on the volumetric element are found. The parameter λ from the algorithm is the same as the barycentric coordinate of the intersection point which is needed to track the endpoints of the cohesive surface. The orientation of the cohesive surface is tracked through the variable *ornt*. This variable stores the orientation using the local nodal ordering of the element. *ornt* is equal to the local numbering of the segment of the element which does not intersect with the cohesive surface. Alternatively, this is the node which is shared by both segments intersecting the cohesive surface.

2.4.2 GCE insertion

The graph used to represent the mesh is developed from the simplicial mesh graph representation described by Mota et al. (2008). To illustrate the graph and its relationship to the mesh, an example one element tetrahedral mesh with associated graph is shown in Figure 2.4.
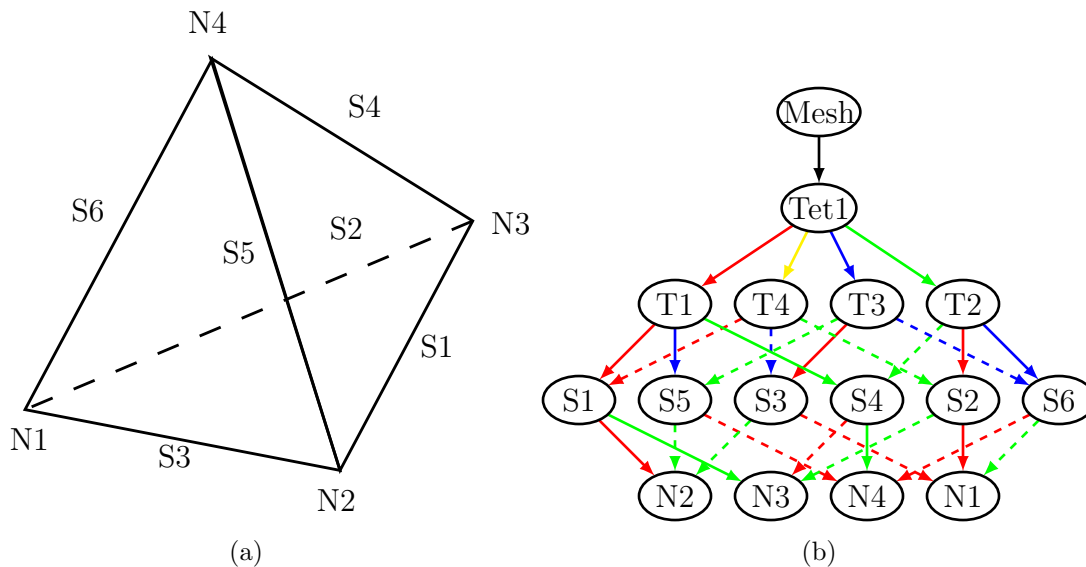

Figure 2.4: A single element tetrahedral finite element mesh (a) with associated graph (b)

In Figure 2.4, the hierarchal nature of the graph mentioned in Chapter 1 is apparent.

In a graph, the individual elements are known as vertices while the lines connecting them are referred to as edges. For this particular application, the edges of the graph are directional. That is, each edge has both an origin and a target. For each vertex, the edges of the graph are either in-edges or out-edges. As the names imply, the in-edges of the vertex are those edges which terminate at the vertex. The out-edges are those which originate at the vertex.

At the top level of the graph is the mesh vertex. This vertex does not describe any portion of the mesh itself, but instead acts as a common vertex for the elements of the mesh. Attached to this vertex are all of the elements of the finite element mesh. In the figure, there is only one element and therefore only one one vertex is attached to the mesh vertex. Attached to the element vertex are the face vertices. These represent the faces of the element. The layer below consists of the segments of the tetrahedral element. The segments of the element are shared between the element faces. The graph represent this by allowing multiple graph edges to contact the graph vertices. The last layer of the graph show the nodes of the element. Note that instead of solid lines, some of the edges in the element edge and node layers are dotted. We have chosen to represent the "ownership" of graph vertices by the line type. To ensure that the correct element connectivity is retained, the first vertex to reference a lower order vertex is deemed the vertex owner. All subsequent connections to that vertex will not be granted ownership, allowing for element ordering to be determined at a later point.

Figure 2.5 describes the way in which the nodal connectivity is tracked in the graph. In the figure, two triangular elements share a common segment. The edge coloring in the graph denotes the local ordering of the edges for a particular vertex. As an example, the ordering of the segments associated with triangle T1 are, (S1,S2,S3). In the same manner, the local ordering of the nodes associated with segment S1 are (N2,N3).

When the elements of the mesh are separate, the edge ordering is sufficient to track the nodal ordering of the elements. However, when an element shares a segment with another element, ambiguities arise in the correct nodal ordering.

Referring back to Figure 2.5, it is seen that both triangle 1 and triangle 2 share segment 1. Using the traditional counter clockwise nodal ordering, a valid nodal ordering for element 1 is (N1,N2,N3). For element 2, a valid ordering is (N4,N3,N2). In the graph, the nodal ordering of segment 1 is (N2,N3). This is consistent with the ordering required by element 1. It is opposite the ordering required by element 2.

Figure 2.6 is a subset of the mesh and graph shown in Figure 2.5 with the direction
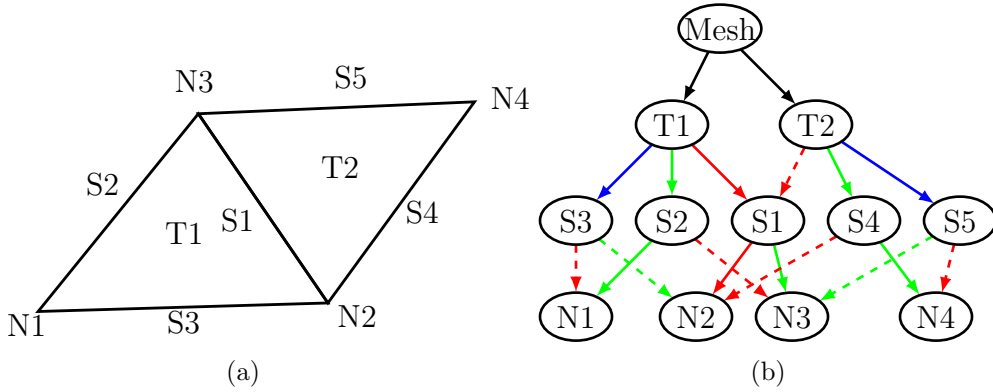
Figure 2.5: Two CST elements sharing a common edge (a) along with associated graph (b)
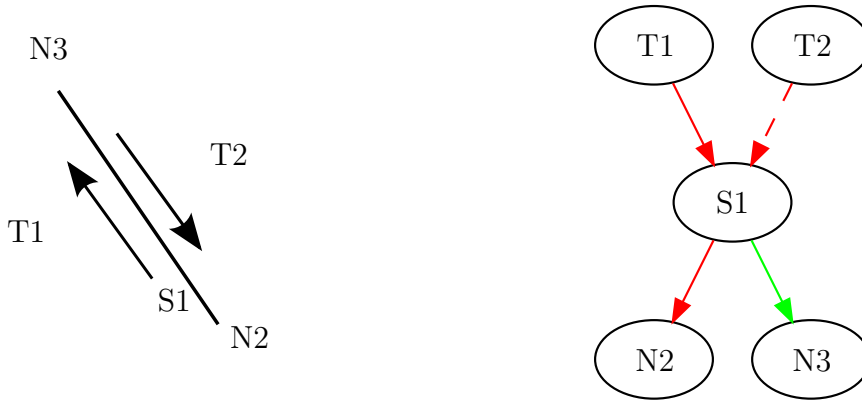
Figure 2.6: Detail view of Figure 2.5 with the common edge shown. Arrows on mesh represent the local nodal ordering for the two elements

of the ordering shown. With segment 1 with respect to element 1, the ordering must be (N2,N3). Segment 1 with respect to element 2 must be (N3,N2). This is tracked through a swap parameter stored in the edge. When the parameter is set, the local ordering out of the target vertex of the edge is swapped. This is visually in the graph by a dotted line. For T1, the edge T1→S1 is solid, meaning that the local ordering of the nodes for S1 is as shown in the graph. For T2, the edge T2→S1 is dashed, telling the program to swap the ordering of the nodes of S1.

The original nodal ordering of the element can be determined by traversing the graph. The coloring and linetype of the edges in the graph track the information necessary to reproduce the nodal ordering, an important consideration when interfacing with legacy codes which require this information.

Algorithm 6 details the process used to insert the GCE into the finite element mesh.

Given the element which will be modified and the orientation of the cohesive surface inside of the element, the algorithm outputs the updated mesh. The routine `clone` associates the properties of the first argument (edges and vertex type) to the second argument. Thus `clone`(T_1, T_2) creates new in-edges and out-edges between T_2 and the vertices connected to T_1. The edge order and ownership is conserved in this step as the two child elements after the insertion process is complete are superimposed on each other. That is, the nodal ordering of the two child elements is consistent with the parent element.

In Algorithm 6, it is important to note that the full mesh representation, $G(lm)$ is not created. Rather, a template has been identified which allows for the creation of two child elements from the parent element. Consequently, a sub-graph, G' for a generic M-noded parent element is created to represent this template. With the pseudo-code in the algorithm, the graph representation of the two child elements is created. This algorithm is able to deal with all degenerate cases, i.e. when the cohesive surface coincides with a segment or node of the element. In addition, these operations are completely localized; they do not affect the neighboring mesh topology. For this reason, the algorithm is able to insert a GCE into multiple elements in a mesh simultaneously. The locality of the algorithm is highly amenable to parallelization. It is shown in Section 3.3 that the algorithm described here is $O(n)$ while an algorithm which utilizes a full mesh representation is $O(n^2)$.

The steps used in the insertion routine are be shown graphically in figures 2.7 to 2.12 to further explain the process. For all pertinent steps, figures are shown along with a short description of the process. Some liberties have been taken in the steps to reinforce certain concepts. Chief among these is a slight reordering of the process to group update actions together. In the algorithm, update of the segments and the nodes of each element are not conducted separately, but instead occur concurrently.

For the following example, the cohesive surface across the element is oriented such that it passes through S_2 and S_3 of the element. Recall that the variable *ornt* tracks the orientation of the surface in the element. This variable is set equal to the segment which is not cut by the cohesive surface using the local segment numbering.

1. The first step in the insertion process is the creation of the sub-graph. As stated above, the insertion works at the local level and is independent of the surrounding mesh. For this reason, the graph, G', represents only the element the current element. In Figure 2.7, as wells as the algorithm, this element is referred to as T_1.

2. A new vertex is added to the graph during the second step (Figure 2.8). This

Input: $ornt, S_c, lm$

    Create sub-graph $G' = G(S_c) \subset G(lm)$

    Add element T_2 to G'

    `clone`(T_1, T_2)

    {Update the original element T_1}

    for all $out\_edges \in T_1$ **do**

      if $S_i \neq ornt$ **then**

        Add $S_{numseg+1}$ to G'

        `clone`$(S_i, S_{numseg+1})$

        Remove S_i from T_1

        $numseg \leftarrow numseg + 1$

      end if

      for all $out\_edges \in S_i$ **do**

        if $n_i = ornt$ **then**

          Add $n_{numnp+1}$ to G'

          `clone`$(n_i, n_{numnp+1})$

          Remove n_i from S_i

          $numnp \leftarrow numnp + 1$

        end if

      end for

    end for

    {Update the new element T_2}

    for all $out\_edges \in T_2$ **do**

      Add $S_{numseg+1}$ to G'

      `clone`$(S_i, S_{numseg+1})$

      Remove S_i from T_2

      $numseg \leftarrow numseg + 1$

      for all $out\_edges \in S_i$ **do**

        if $n_i \neq ornt$ **then**

          Add $n_{numnp+1}$ to g

          `clone`$(n_i, n_{numnp+1})$

          Remove n_i from S_i

          $numnp \leftarrow numnp + 1$

        end if

      end for

    end for

    Get new connectivity

Algorithm 6: GCE insertion
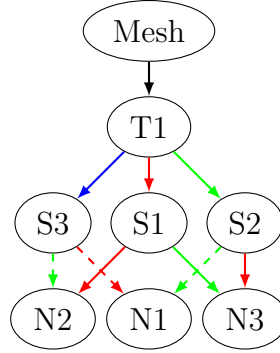
Figure 2.7: GCE insertion step 1: create graph

new vertex will represent the second element needed for the GCE. This element, T_2, will begin with the same properties as T_1 in the graph. To do this, the edges of T_1 are copied to T_2 using the routine `clone`. New edges between T_2 and the vertices connected to T_1 are created and given the same properties as the original edges.
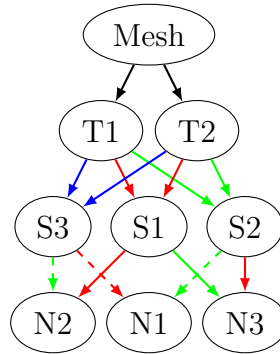
Figure 2.8: GCE insertion step 2: duplicate $T1$

3. During the third step, shown in Figure 2.9, the segments of T_1 are updated. As the insertion processes is local to the element, two segments must be replaced in T_1. These are the segments cut by the cohesive surface. In the next step, the nodes of the element will change. If these segments were not replaced, the original segments and thus the elements surrounding T_1 would change, destroying the locality of the algorithm. As in the second step, the edges of the original segment will be copied to the new segments, yielding the same connectivity as the original element.

4. The fourth step deals with the nodes of T_1. This is the last step that modifies T_1. The nodes not equal to *ornt* shall be retained in the element while the node equal to *ornt* shall be replaced. A new node, again with the same edges as the node it replaces, is inserted into the graph and associated with the element.
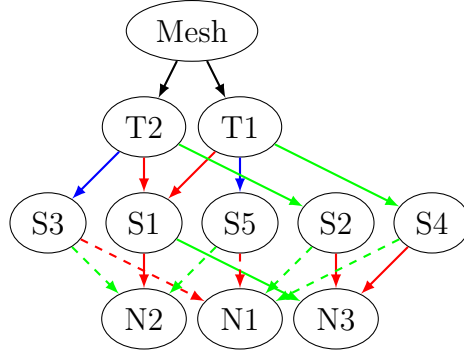
Figure 2.9: GCE insertion step 3: update segments of $T1$

This new node is the first of the virtual nodes described in the GCE section to be added to the graph. Figure 2.10 details the graph at this stage.
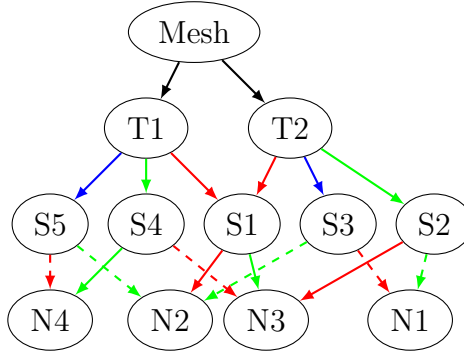
Figure 2.10: GCE insertion step 4: update nodes of $T1$

5. In the fifth step, a process similar to the third is performed. The segments of T_2 are updated to retain the locality of the insertion process. As only one of the original nodes of the parent element will remain in this element, all of the segments shall be different than the original element. Thus, all segments of T_2 must be updated. The properties of the new segments are again cloned from the segments being replaced. Note that in Figure 2.11, the two elements are now connected to each other only through the nodes.

6. The final step in the insertion process is the update of the nodes in T_2. This follows the procedure described in step four, but with all nodes except *ornt* replaced. Figure 2.12 shows the final graph for the process. While there have been insertions of new segments and nodes in both elements, the basic connectivity seen in the first step is still apparent in both elements. The three nodes in the parent element at the beginning of the process are distributed among the two child elements. Three additional nodes, called virtual nodes, have been added to the graph. While T_1 retains one of the original segments, the segments of T_2
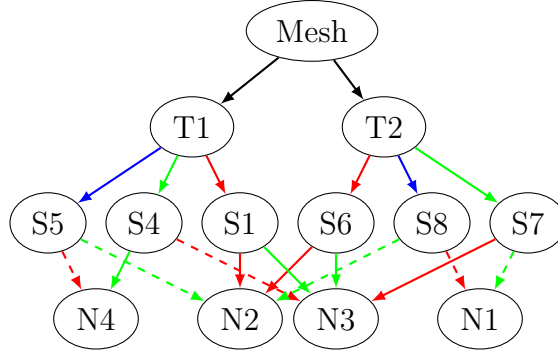
Figure 2.11: GCE insertion step 5: update segments of $T2$

are all new. To update the connectivity of lm, the connectivity array, one may iterate through the out-edges of the two elements, using the order of the edges and ownership between the element and the segment to find the correct nodal ordering. In the final graph, as well as the resulting connectivity array, the two child elements are independent of each other. The two child elements share no common segments or nodes. Connection between the child elements exists only along the cohesive surface described by the GCE. As described previously, this surface is tracked separately from the mesh by interpolating from the nodes of the child elements in the reference configuration.
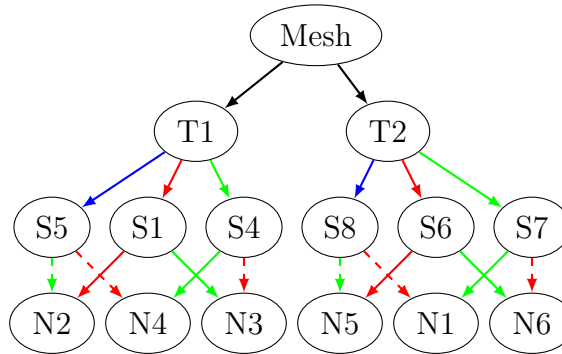
Figure 2.12: GCE insertion step 6: update nodes of $T2$

It was stated above that the Algorithm 6 is valid for certain degenerate cases which may occur during the insertion process. There are two degenerate cases which must be considered when inserting the GCE into the mesh. These will occur when the cohesive surface of the GCE does not pass through the interior of the volumetric element but instead is coincident with the surface (boundary) of the volumetric element. In the first of the cases, the cohesive surface is oriented along one of the edges of the volumetric element. The parent element before insertion and the child elements after insertion are shown in Figure 2.13 with the orientation of the cohesive surface shown

in red.
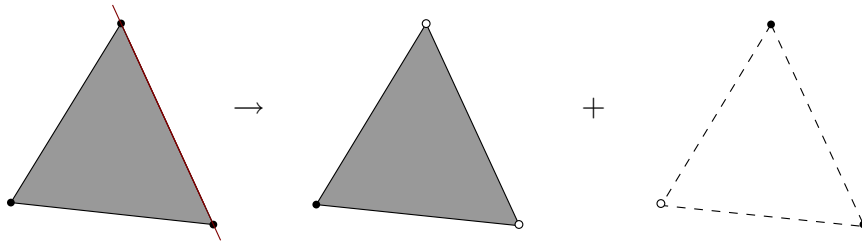
Figure 2.13: GCE degenerate case: cohesive surface oriented along element edge

After insertion, the original volume of the parent element is distributed such that one of the child elements receives the entirety of the volume while the other has none of the original volume. This case is analogous to the conventional cohesive element with the exception that although the cohesive surface is on the surface of the bulk element, it is still associated with only the bulk elements of the GCE.

The second degenerate case which may occur is shown in Figure 2.14. Again the orientation of the cohesive surface on the volumetric element is shown in red. In this situation, the cohesive surface is not passing along the element boundary, but intersecting the element only at one of the nodes. This case may occur if one of the surrounding elements has a cohesive surface which passes through a node, either due to the first degenerate case discussed above, or with a cohesive surface which passes through the interior of the adjacent element. It is necessary to consider this case as we must allow the body to fully fracture. Without it, the elements surrounding a GCE which passes through a node of the volumetric element would artificially stiffen the body as the mesh would not have a continuous discontinuity. As in the previous case, the entire volume of the parent element is within one of the child elements while the other has none.
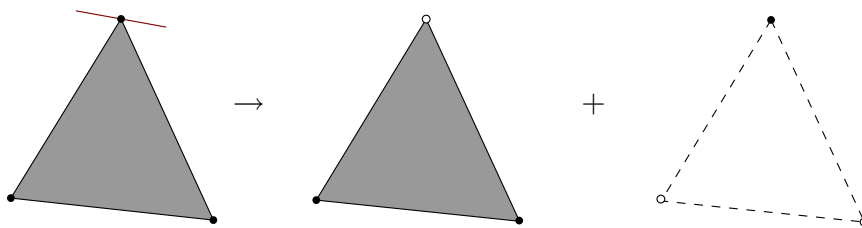
Figure 2.14: GCE degenerate case: cohesive surface passing through volumetric element node

Other possible requirements may add to the number of cases which the algorithm must consider when a GCE is inserted. One possibility is crack branching. There are many examples in the literature of experimental testing which result in a failure with crack branching. Although not developed in this work, integration of this phenomena would allow for usage in a wider variety of failure problems. A preliminary procedure for inclusion of branching in the GCE method is given in Appendix B.

Chapter 3

Numerical Results

3.1 Introduction

In this chapter, the utility and accuracy of the GCE insertion method will be shown. A series of test highlighting the applications of the method are shown first. In Section 3.2, the ability to insert a line of GCEs arbitrarily into the mesh is shown. Depending on the application, a cohesive line of this type may be inserted *a priori* into the mesh. Section 3.3 discusses the benefits gained through a sub-graph mesh representation as opposed to the alternative full graph mesh representation. Over a wide range of mesh sizes, the time to insert a GCE into every element in a mesh is approximately linear with the sub-graph mesh representation while the time taken with the full graph mesh representation is approximately quadratic for the same task. The ability to propagate a cohesive surface with GCEs is shown in Section 3.4. Section 3.5 compares the results of the conventional cohesive elements and GCE method for a notched plate. The final test in this chapter (Section 3.6) is an analysis of a ballistic impact on a double notched plate using the GCE method. This method is compared to the same specimen modeled using conventional cohesive elements.

The meshes used in the following examples were created using Triangle, a 2D unstructured mesh generator (Shewchuk (1996)).

3.2 GCE insertion along a predetermined path

In addition to adaptive insertion, a GCE may be inserted into a finite element mesh *a priori*. Figure 3.1 shows a representative test of the GCE insertion routine along three predetermined paths. In the figure, the grey portion of the mesh represents the elements which have not been modified. The red elements are those with inserted GCEs. The darker portions of these elements show the remaining material in the element while the light portions represent the portion with no remaining material. The regions on either side of the cohesive line have been separated to highlight the path.

While all three examples show the result to some degree, Figures 3.1 (b) and (c) clearly show degenerate cases of the GCE insertion. On the endpoints of the cut, the cut passes through only the node of an element. In these situations, the entire original volume of the parent element is allocated to one of the child elements while the other receives none.

Besides a single cut plane as is shown in Figure 3.1, the insertion process may also be done over multiple paths. Figure 3.2 shows the same mesh as in the previous figure cut by two predetermined paths.

From Figures 3.1 and 3.2, it is seen that the insertion may proceed at an arbitrary angle through a finite element mesh. The unstructured nature of the mesh and the resulting random nodal ordering of the elements in the mesh show that a GCE may be oriented at will within the individual finite elements, including along a element boundary or through an element node. Further, there is no inherent limit to the number of crack paths which may be inserted into the mesh. In cases where the failure response is well understood or alternatively where a preexisting discontinuity exists, a line or multiple lines of GCEs may be inserted prior to the analysis. While the above examples show straight lines across the mesh, this is not a restriction of the method. Paths of any shape may be created provided that the path within the elements of the mesh are straight. In fact, analyses where the path is not predetermined are likely to have paths which diverge from the ideal straight path due to numerical considerations.
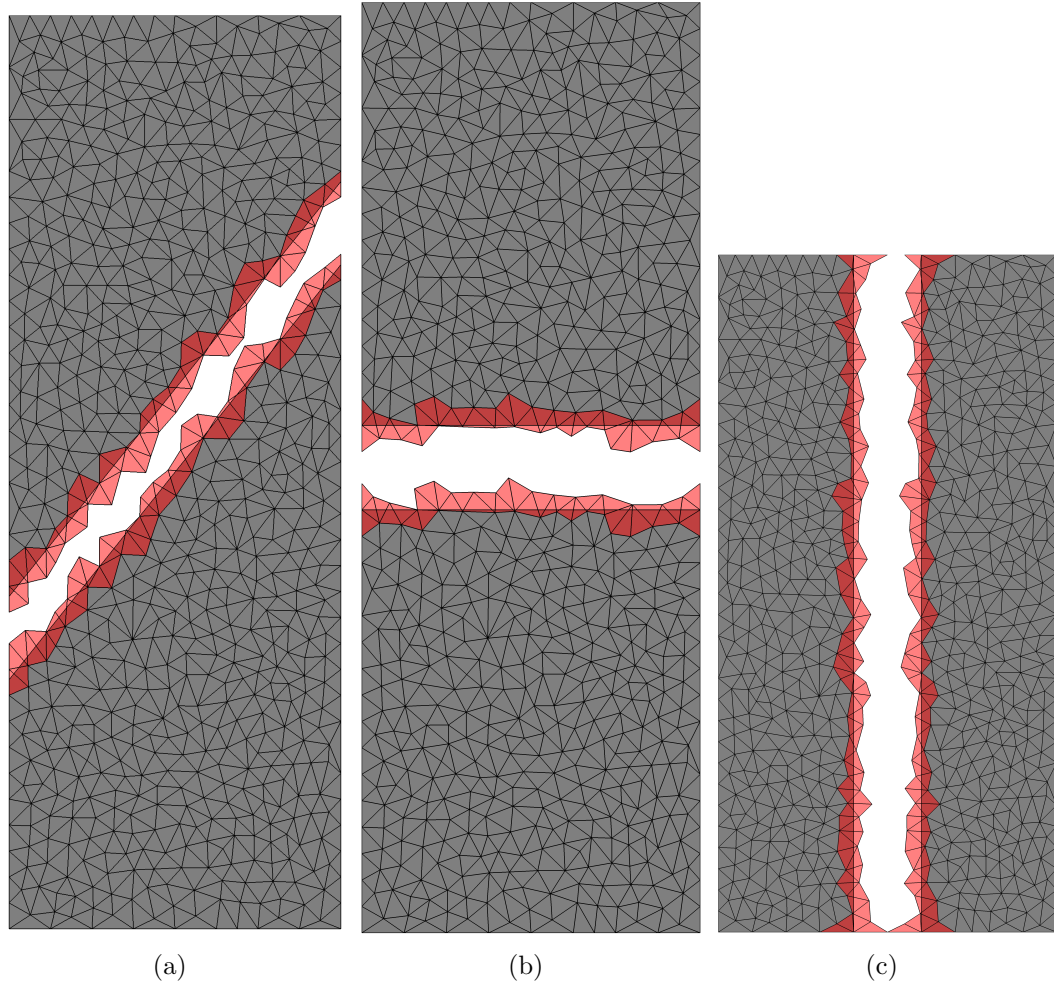
Figure 3.1: GCE insertion using the sub-graph mesh representation along predefined cut-planes at: (a) 45°, (b) 0°, (c) 90°

3.3 The benefit of sub-graphing

As stated in the formulation chapter, the traditional method of representing the finite element mesh within the computer code uses a set of arrays. Recently, Mota et al. (2008) has suggested the use of a directed, hierarchal graph as an alternative data storage device. This graph based approach simplifies the insertion process involved in the fracture analysis, but as the majority of the finite element program used is based on legacy codes, there is a need for a method which is readily applicable to existing code base. Thus two different methods of storing the mesh data were evaluated. In the first, the entire mesh was represented by the graph structure, while in the second, a sub-graph of only the element to be fractured was created as needed. In both, the
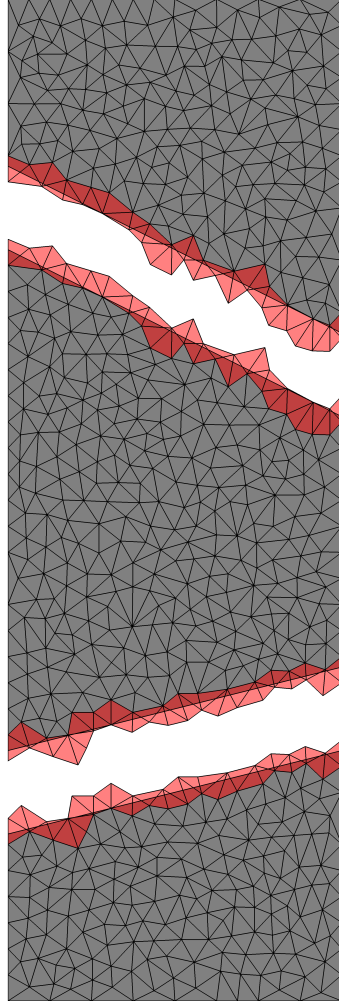
Figure 3.2: GCE insertion using the sub-graph mesh representation along multiple predetermined paths

graph (full or sub-graph) was used to update the arrays holding the mesh information for the legacy subroutines.

In the full mesh representation, the initialization process of the graph is done at the beginning of the analysis and updated as needed throughout the course of the analysis. The sub-graph method however creates a graph representing only the small subset of the entire mesh when a GCE is inserted. After insertion, this sub-graph is discarded and a new graph is created for additional insertions. The above tests were conducted using a computer with a 2.80 GHz Intel Core 2 Duo E7400 and 8 GB RAM running Redhat 5.5 Beta. The compilers used were GCC 4.1.2 for the C++ portions of the code and ifort 9.0 for the Fortran routines and final linking. The number of nodes and elements in the meshes tested ranged from 76 nodes, 126 elements to $207,724$

nodes, $413,986$ elements.

To evaluate the run time differences between the two methods, a GCE was inserted into all elements of a 40μm $\times$ 100μm rectangular mesh of three noded triangles under varying mesh densities. Insertions were conducted at random local element orientation without regard to the surrounding elements. This resulted in a final mesh with individual cohesive surfaces in each element with few if any continuous surfaces across elements. Results of these tests are given in Figure 3.3.
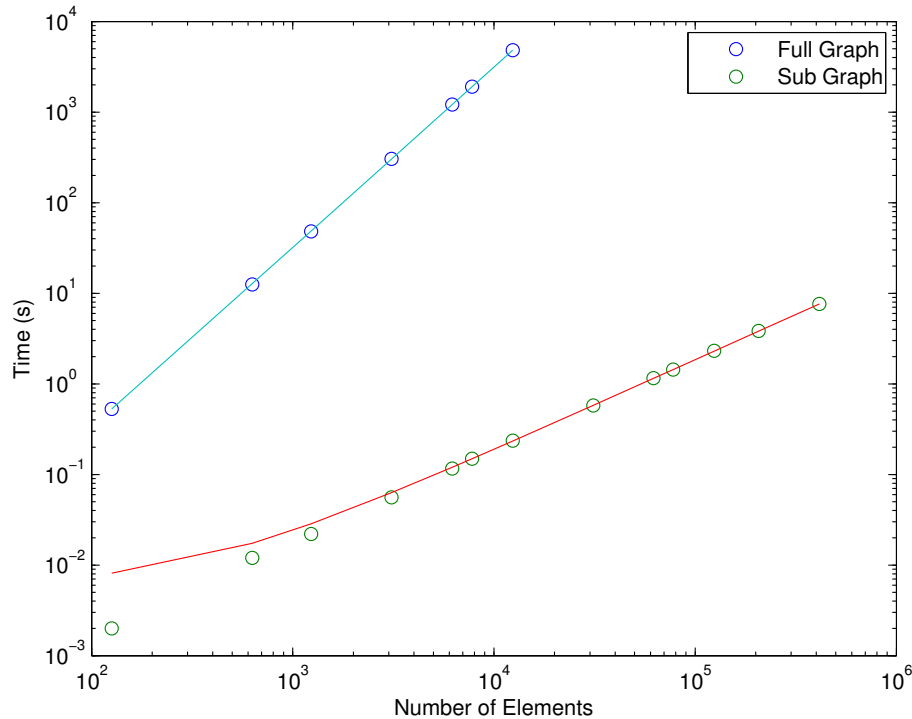
Figure 3.3: Time to insert GCE within every element of a finite element mesh

From the figure, it is apparent that there is a significant difference between the time needed to fully fracture a mesh using a sub-graph mesh representation as opposed to a full mesh representation. The sub-graph method is $O(n)$ while the full mesh representation is $O(n^2)$. For a mesh of $12,400$ elements, the full graph method takes approximately 80 minutes while the same process using the sub-graphing method is completed in approximately 0.25 seconds. A linear fit line is shown for the sub-graph data and a quadratic fit line is shown for the full graph data.

When using the sub-graphing method, no apparent problems with GCE insertion are

seen even in meshes with a relatively large number of elements (the largest shown in Figure 3.3 is 414k elements). The growth in run time is linear to the number of inserted GCEs in the mesh for the sub-graphing method while quadratic using a full mesh representation.

Despite the differences between the GCE method and the conventional cohesive element method used by other authors, alternative insertion methods such as the graph method of Mota et al. (2008) and the TopS method of Paulino et al. (2008) are comparable in speed to the sub-graph method described here, each method completing the insertion in approximately linear time with respect to the number of insertions.

3.4   Cohesive surface propagation

The previous sections detailed the ability to insert a GCE and the efficiency of doing so, but did not discuss the main purpose of the method; namely the modeling of dynamic fracture. This section, as well as the following, will focus on this aspect of the work. In Figure 3.4, a crack, beginning at the tip of a pre-existing notch is propagated through a plate. The bottom edge of the plate is fixed while a constant velocity is applied to the top edge. Original specimen dimentions are 15 μm by 70 μm. The left and right edges are free. To reduce the effect of crack nucleation on the test, 30 GCEs have been pre-inserted into the mesh starting at the notch tip reaching approximately half of the way through the remaining material in the plate. The cohesive surface is then allowed to grow as required.

Figure 3.4(a) shows the principal stress in the elements after GCEs have begun to be inserted into the mesh. The growing cohesive zone in the elements with inserted GCEs cause a stress concentration at the tip of the cohesive surface. GCEs are inserted into the mesh until the cohesive surface reaches the right side of the mesh. Figure 3.4(b) shows the principal stress in the mesh after the cohesive line has reached the far side of the specimen. At the conclusion of the test, there are no significant stress concentrations in the mesh.

3.5   Notched Plate

To compare the GCE model with a conventional cohesive element analysis, a notched plate similar to the one described in Park (2009) is used. Figure 3.5 describes the
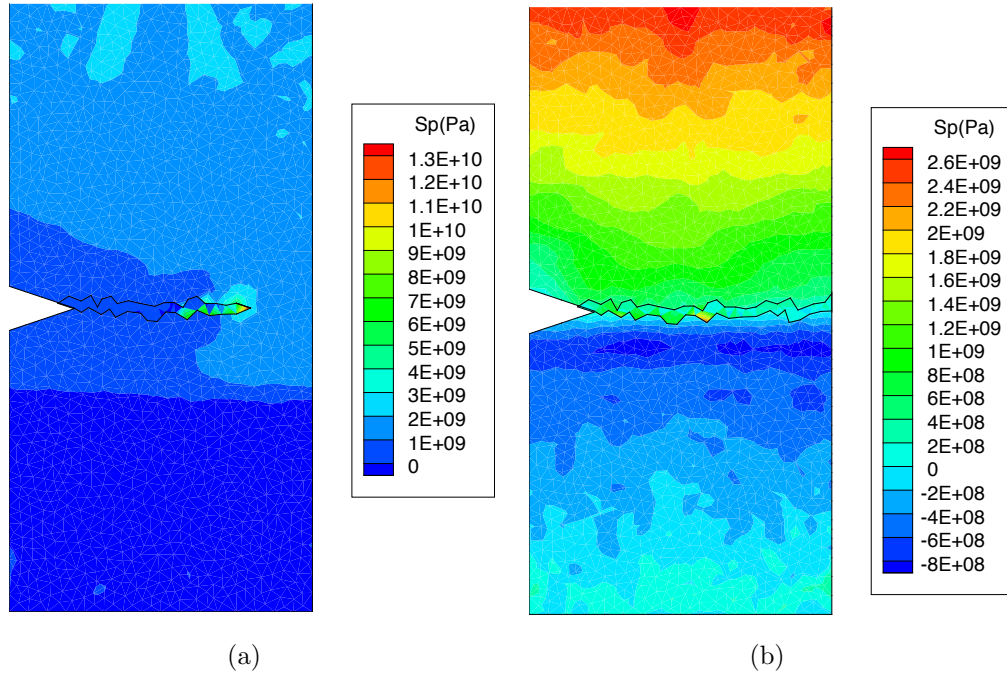
Figure 3.4: Propagation of a crack through a vertically loaded notched plate.

geometry and loading conditions. The specimen is fixed at the right end and subjected to an applied velocity $\pm v_0$ on the left end. A initial crack of length $a_0 = 0.2$ mm is situated at left side of the mesh on the mid-plane. The velocity is ramped from zero to 100 m/s over the first five time-steps of the analysis and is constant thereafter.
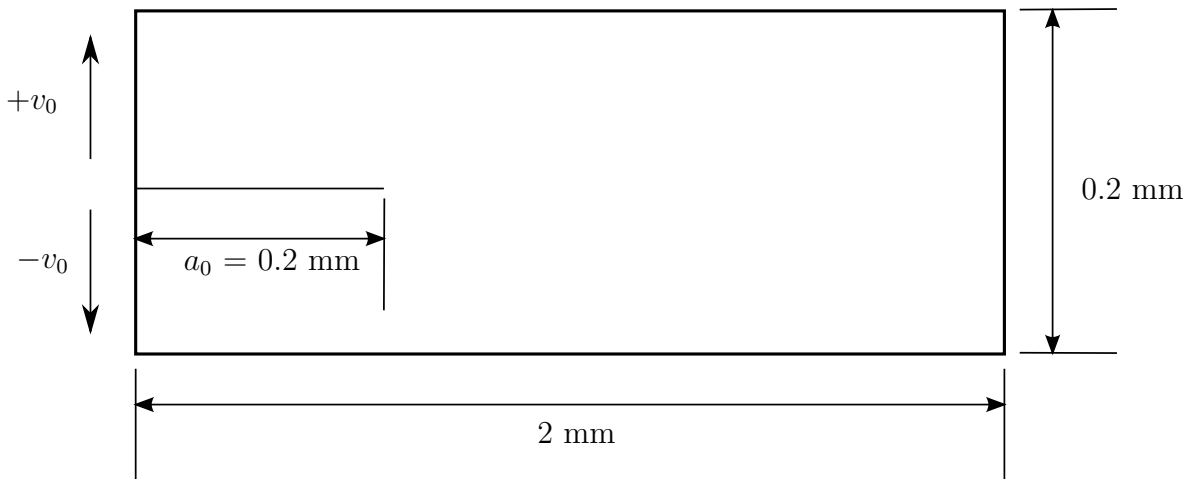
Figure 3.5: Experimental setup of a notched plate under applied velocity along top and bottom surfaces (not to scale).

Material and cohesive parameters are given in Table 3.1. The cohesive strengths σ_{max} and τ_{max} are taken as 1/10 of the Young's modulus. The mode I and mode II fracture energies (G_I and G_{II}) are both equal to 352 N/m. The critical normal and tangential opening displacements (Δ_{nc} and Δ_{tc}) are related to the cohesive strengths and the fracture energies through,

$$G_I = \tfrac{1}{2}\sigma_{max}\Delta_{nc}, \quad G_{II} = \tfrac{1}{2}\tau_{max}\Delta_{tc} \tag{3.1}$$

Table 3.1:

Material and cohesive parameters: notched plate

E (GPa)	ν	ρ (kg/m^3)	$\sigma_{max} = \tau_{max}$ (MPa)	$\Delta_{nc} = \Delta_{tc}$ (μm)
3.24	0.35	1190	324	2.173

An unstructured mesh of approximately $45,000$ elements is used for both the GCE and benchmark conventional cohesive element tests. For the conventional cohesive element test, there is a line of cohesive elements pre-inserted along the mid-plane of the plate. For the GCE test, the GCE insertion routine is constrained such that the insertions will follow a single line and the path of the cohesive surface is oriented at zero degrees. These constraints have been added such that the GCE test and the benchmark are as similar as possible.

The results of the GCE insertion tests are shown in Figure 3.6. In addition to a purely adaptive GCE insertion, the results of the GCE analysis when a small number of GCEs have been pre-inserted into the mesh are shown. After the tests had begun, the insertion routine was allowed to run unimpeded except for the aforementioned constrains on the GCE analyses for the remainder of the tests.

From the figure, it is seen that when there are no pre-inserted elements (labeled fully adaptive insertion in the figure), the cohesive zone begins to form significantly earlier than in the partially pre-inserted and reference conventional cohesive element case. Additionally, in the adaptive case, the crack extension (defined as the total length of fully failed cohesive elements) begins shortly after the cohesive zone has begun. When a small number of GCEs have been inserted into the mesh however, both the crack and the cohesive zone develop in a manner similar to the reference case up to $0.1\ \mu$sec. After this time, the results diverge. When divergence begins, the length of the cohesive zone is greater than the length of the cohesive surface pre-inserted into the mesh. That is, the divergence begins after the cohesive zone grows past the GCEs inserted *a priori*.

One possible reason for the discrepancy between the adaptive insertion case and the
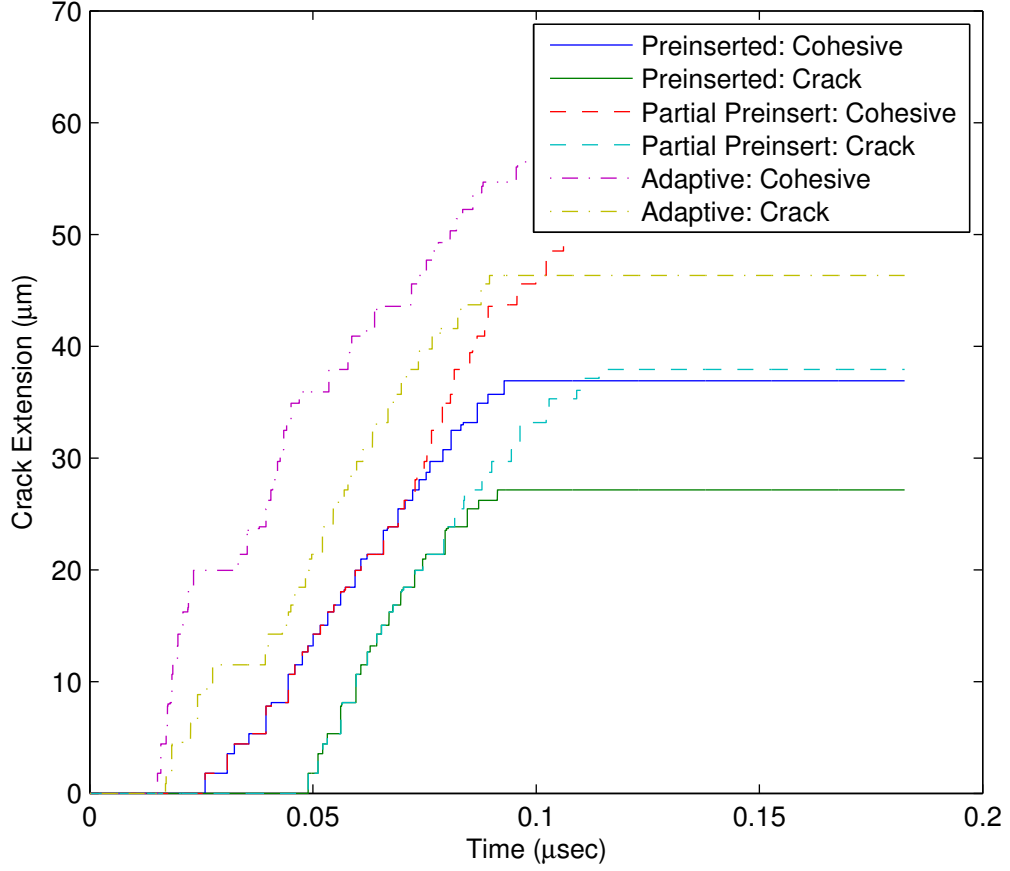
Figure 3.6: Crack extension and cohesive zone length of a notched plate with $v = 100$ m/s with fully adaptive GCE (no pre-inserted GCE), partial pre-inserted GCE, and pre-inserted conventional cohesive elements

reference case deals with the cohesive zone length. Rice (1968) states that for a converged result, the cohesive zone length must be equal to,

$$R = \frac{\pi}{8} \frac{E}{1 - \nu^2} \frac{G_{Ic}}{\sigma_{ave}^2} \tag{3.2}$$

with $\sigma_{ave} = \sigma_{max}/2$.

For the material parameters used for this analysis, this value must be equal to 19 μm for a converged result. Geubelle and Baylor (1998); Klein et al. (2001) suggest that in addition to the cohesive zone length criterion, the length of a cohesive element must not exceed 1/2 to 1/3 of the cohesive zone length.For a converged result with these material parameters, the maximum cohesive element size should be between

6.3 μm and 9.5 μm. While it is trivial to create a mesh with a mesh density along the mid-plane refined enough that this rule is met for the conventional cohesive element test, the nature of the GCE insertion poses more of a problem. As the GCE is not typically oriented along the edge of the volumetric element but instead passes through the element, any length between zero (the GCE passes through only a node of the element) and the diameter of a circle circumscribed on the element is possible. In aggregate however, analysis of this mesh shows that using a mesh density equal to that used in the conventional cohesive element test yields a majority of GCE elements which fall within the prescribed limit. It was hypothesized that the available cohesive zone ahead of the crack tip and not the cohesive element size was the source of the erroneous results. Thus, a test was constructed to determine if additional GCEs along the crack path ahead of the crack tip would result in an analysis closer to the reference case.

This was accomplished by varying the parameter α. From Equation 2.26, the insertion criterion was met if the stress in an element is equal to the cohesive strength of the material multiplied by a scalar α. Varying this value determines how many GCEs are inserted into the mesh. With $\alpha = 0$, a GCE will be inserted in all elements along the crack path in the first timestep, similar to the pre-inserted case. When $\alpha = 1$, the GCe will not be inserted until the stress in the element is equal to the cohesive strength of the material. This will result in cohesive elements which begin to fail quickly after insertion. The crack extension with α varied between zero and one is shown in Figure 3.7.

Testing shows that with $\alpha = 0$, the response of the system is equivalent to a line of pre-inserted GCEs. Additional testing with a line of pre-inserted GCEs yield the same result as a pre-inserted line of conventional cohesive elements. Increased values of α result in an increased departure from the reference case. Moderate values of α however, up to 0.5 yielded results similar to the reference. These results imply that the discrepancy seen in Figure 3.6 is indeed caused by an insufficient number of GCEs in front of the crack tip.

Two solutions to the above problem have been suggested. First, as in the α test, the multiplier α in the insertion criterion may be lowered to more appropriate value (at most 0.5). A second solution is to ensure at insertion that the crack tip is sufficiently behind the leading edge of the inserted GCEs. This may be accomplished by inserting multiple GCEs ahead of the element which has reached the insertion criterion.

The first solution has the benefit of simplicity, it may be implemented by changing a single value. The detriment however is that significantly more GCEs than necessary may be introduced into the mesh. Without constraint on the insertion location, GCEs
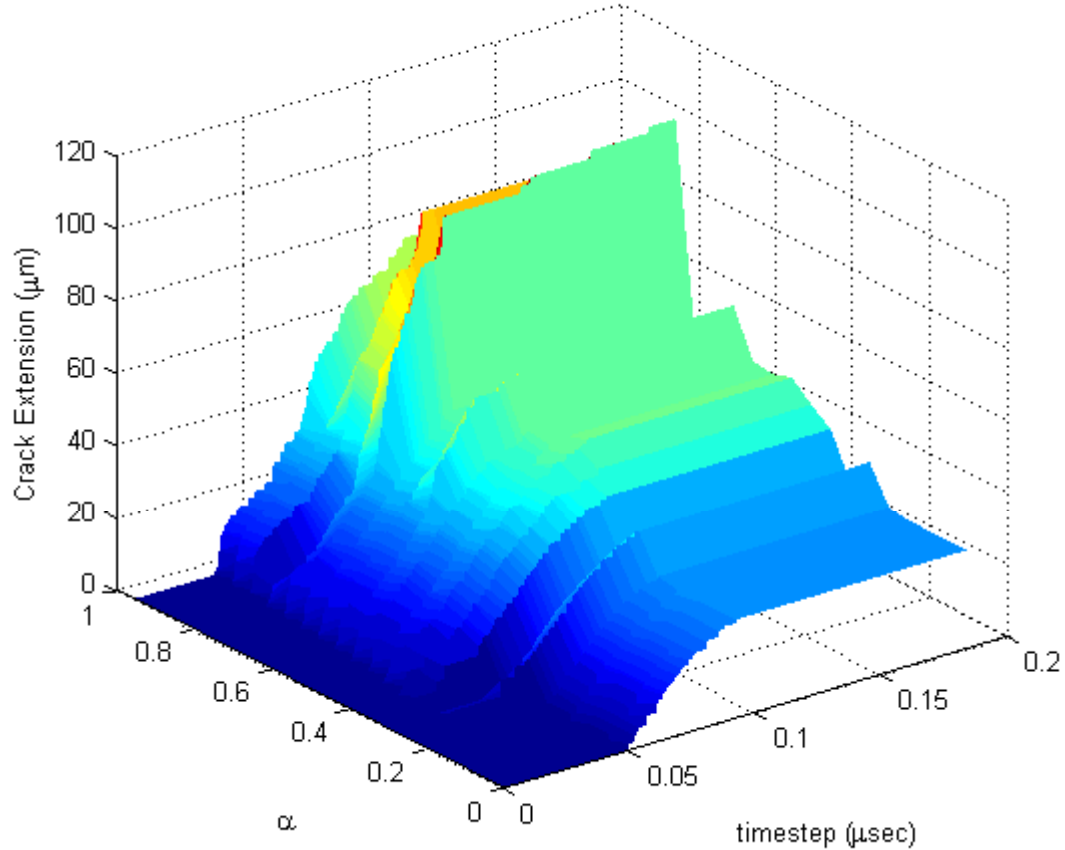
Figure 3.7: The effect of α on the adaptive GCE insertion routine of a notched plate under applied loading

will be inserted in a wide area of the mesh surrounding any stress concentration. As only a small portion of them are likely to develop into crack nucleation sites, this will add additional overhead into the analysis. Additionally, as the current implementation of the code allows for only a single GCE to be inserted into a volumetric element, spurious GCE insertion may affect the crack development as the prematurely inserted GCEs constrict the possible crack paths.

The second of the suggestions appear to be superior. As the cohesive zone length is controlled by adding additional GCEs at insertion, α may be set at a higher value, allowing less widespread insertion of possible nucleation sites and allowing for the development of a more cohesive crack path.

51

3.6   Ballistic impact

Another common test used in dynamic fracture tests will be used to compare the results from the GCE and the reference conventional cohesive element method. The simulation which will be conducted is based off of an experiment described in Kalthoff and Winkler (1987). Numerical analysis of this test has been conducted in several different papers, some examples of which are Belytschko et al. (2003); Zhang and Paulino (2005); Song et al. (2006a). In this test, a maraging steel 18Ni1900 plate with material properties given in Table 3.2 is used. The G_{Ic} and G_{IIc} fracture energies for this material are equal to 2.213×10^4 N/m.

Table 3.2:

Material and cohesive parameters: 18NI1900 maraging steel

E (GPa)	ν	ρ (kg/m^3)	$\sigma_{max} = \tau_{max}$ (MPa)	$\Delta_{nc} = \Delta_{tc}$ (μm)
190	0.30	8000	844	52.45

Physical dimensions of the plate are given in Figure 3.8. As in Song et al. (2006a), the symmetry of the system shall be utilized such that only half of the plate is modeled. The dotted line in the figure shows the extent of the numerically modeled region. Also following Song et al. (2006a), it is assumed that the elastic impedance of the projectile is the same as that of the specimen and that the applied velocity on the portion of the plate between the notches to $v_0 = 16.5$ m/s, one half of the velocity used in Kalthoff and Winkler (1987). While Kalthoff and Winkler (1987) describes failure with both a brittle and shear band behavior depending on impact velocity, the velocity used her has been chosen to fall within the brittle failure range. The notch is modeled as two parallel lines separated by 0.3 mm.

The crack tip extension for the CE and GCE tests over the course of the analyses are shown in Figure 3.9. The cohesive zone extension for the same analysis is shown in Figure 3.10. From these two figures, it is again shown that there is a high degree of similarity between the results when using the CE and GCE methods. In particular, note that both methods predict the crack propagation to begin at approximately 5μsec. The cohesive zone extension and crack extension for both of the methods are shown in Figures 3.11 and 3.12 for the CE and GCE methods, respectively.

Another view of the similarity between the methods may be seen in the velocities of the crack tip. Figure 3.13 shows the velocity of the crack tip for the CE test while Figure 3.14 shows the velocity for the GCE test.
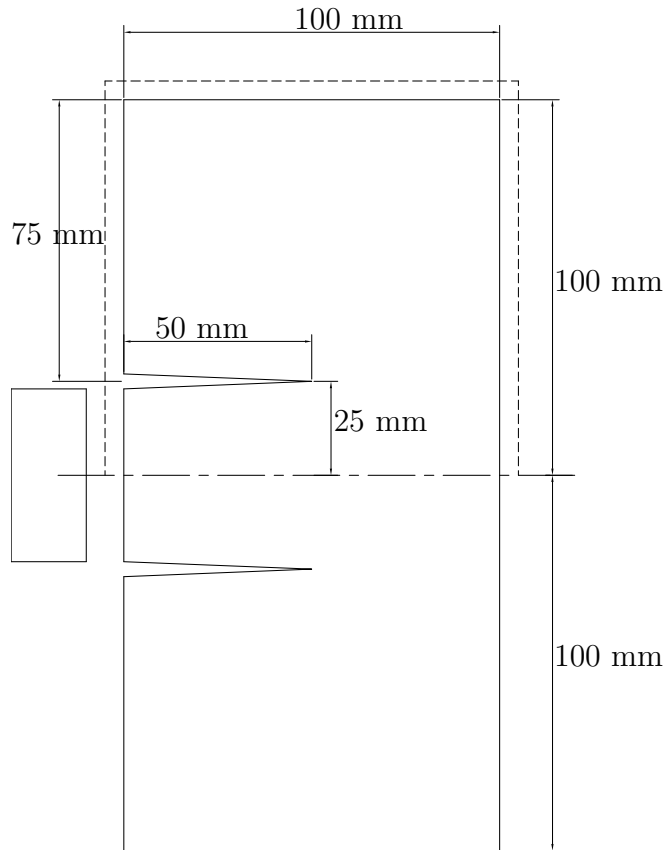
Figure 3.8: Experimental setup for an edge-cracked plate under ballistic impact. Dashed box denotes extent of numerically modeled region.

Compared with data from the papers listed at the introduction to this section, the total time for full failure of the specimen was similar, taking approximately 70μsec. The time at which the crack propagation begins however was not fully consistent with the published data. In these tests, the propagation, both for the CE and GCE methods, began around 5μsec. Published work shows crack propagation starting at 25μsec. However, it is seen in the velocity figures that there were two distinct sections in the propagation response. There was a short period of growth between 5 and 20μsec with the growth beginning again at 30μsec. This second period of growth is similar to the start of the crack propagation shown in the literature.
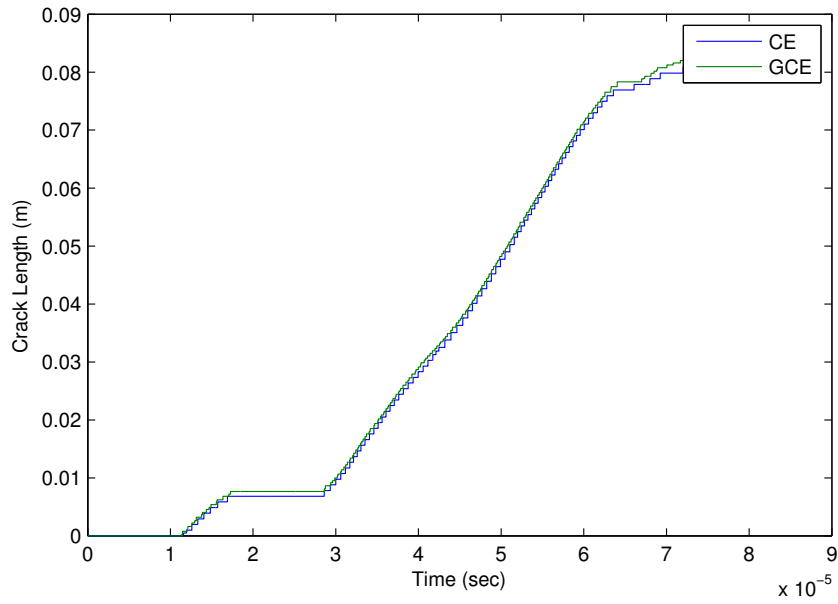
Figure 3.9: The crack tip extension for a notched plate under ballistic impact using CE and GCE elements
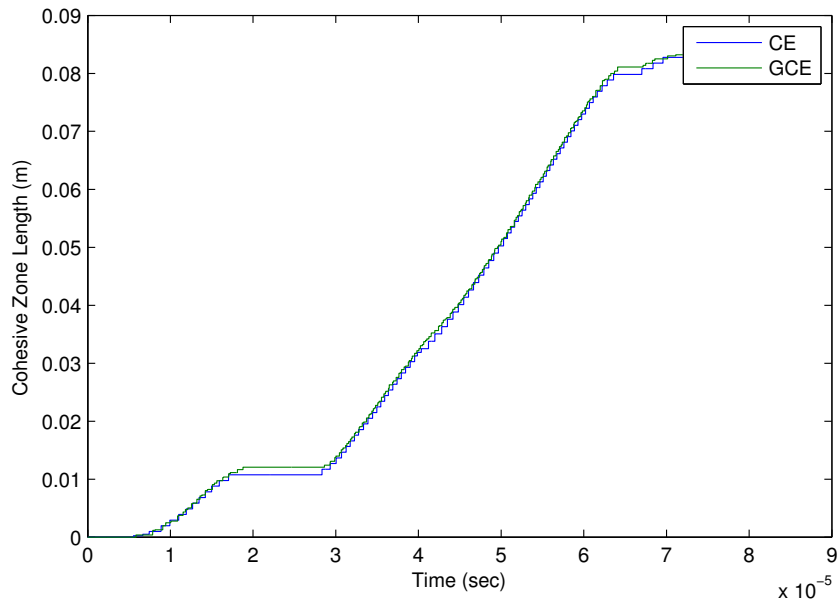
Figure 3.10: The cohesive zone extension for a notched plate under ballistic impact using CE and GCE elements

54
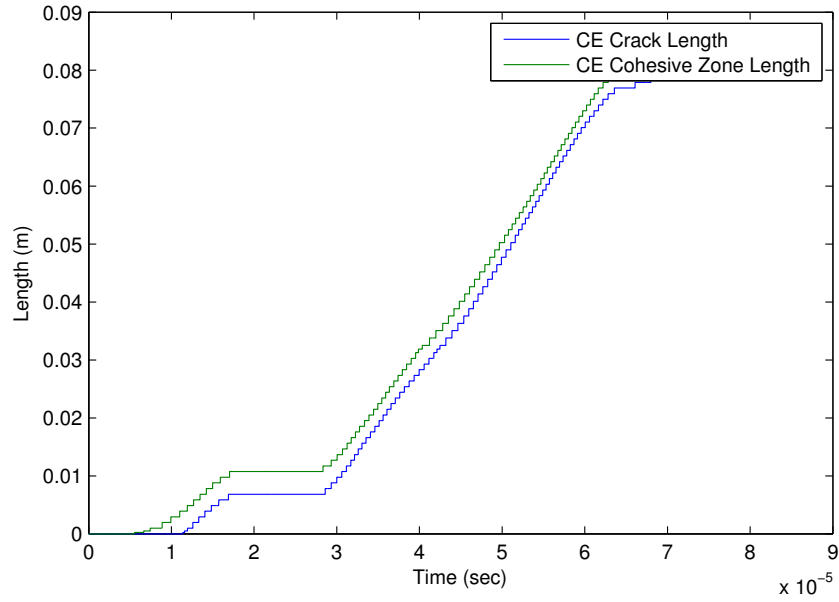
Figure 3.11: The cohesive zone and crack extension for a notched plate under ballistic impact using CE elements
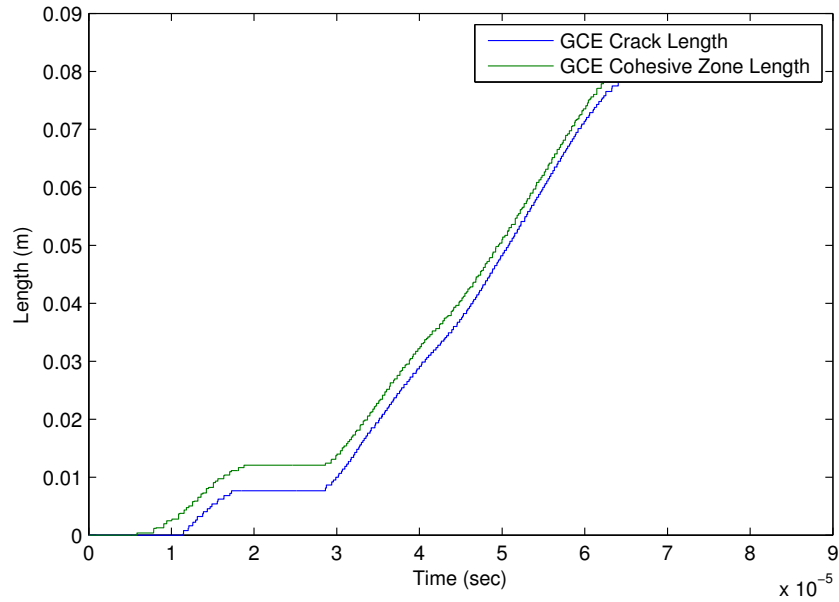
Figure 3.12: The cohesive zone and crack extension for a notched plate under ballistic impact using GCE elements
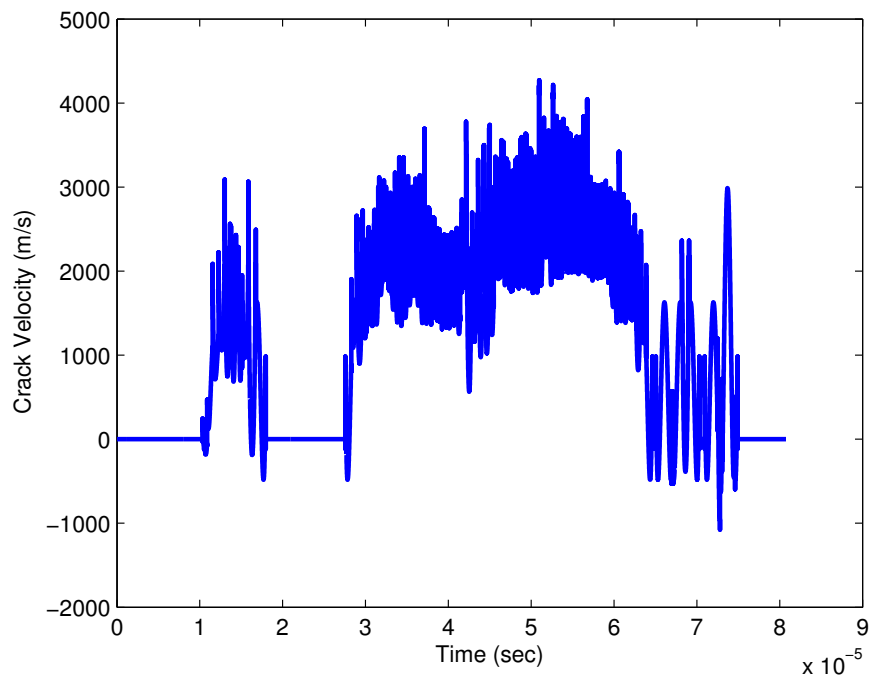
Figure 3.13: The crack tip velocity for a notched plate under impact loading using CE elements
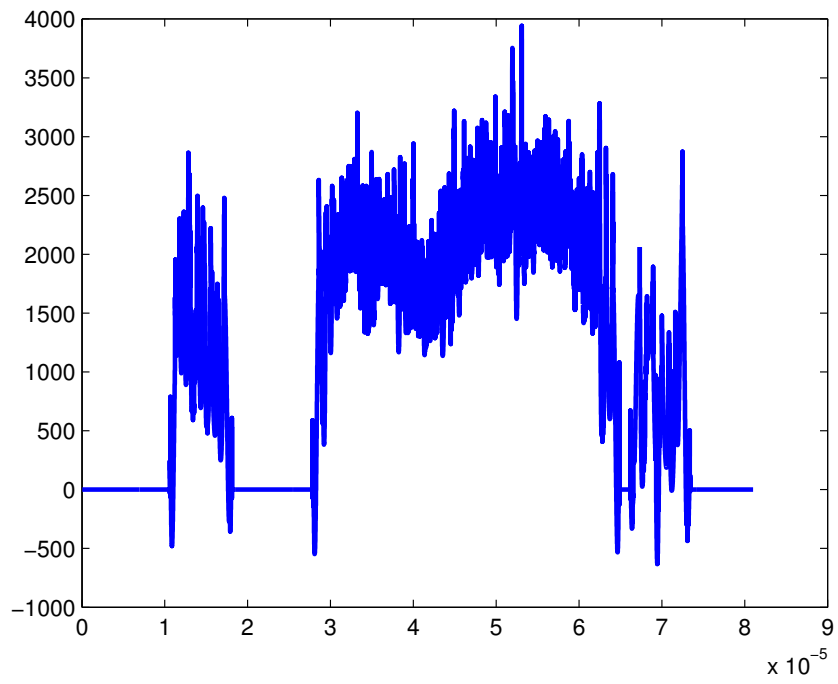
Figure 3.14: The crack tip velocity for a notched plate under impact loading using GCE elements

Chapter 4

Conclusions and Future Work

4.1 Findings of the current work

This thesis has shown that the representation of a finite element mesh as a graph based on a simplicial complex may be used to conduct dynamic fracture research. Further, the generalized cohesive element method has been proven valid for dynamic analyses. Chapter 3 has shown that the GCE method is valid for when the GCE insertion is conducted both *a priori* and adaptively during the course of the analysis. The GCE method was seen to be comparable to a reference conventional cohesive element method for a variety of dynamic fracture tests used in literature.

In the same chapter, the utility of the sub-graph method was demonstrated. Instead of storing a graph of the entire finite element mesh, it is more efficient to create and destroy sub-graphs as needed during the insertion process. The full graph representation inserted GCEs in $O(n^2)$ time while the sub-graph representation did the same task in $O(n)$ time.

It was shown that the proposed method can, in a robust and intuitive manner, insert GCE at will within a finite element mesh. This insertion is conducted in the interior of the bulk element of the mesh instead of at the boundary of the bulk element as in the conventional cohesive element. Orientation of the GCE within the bulk element may be arbitrary in both direction and location. Degenerate cases along the boundary of the element and at the node of an element are well defined and are handled automatically without additional logic in the formulation.

This work has shown the potential for the inclusion of a strong discontinuity (in the guise of a cohesive element) into a finite element mesh without dependence on the discretization of the domain. As opposed to the published literature using the cohesive element method which inserts cohesive surfaces on the boundaries of the volumetric elements, this work can insert, at arbitrary location and angle, a cohesive surface on the interior of a volumetric element.

While the method has been developed specifically for the GCE method, it is applicable for other XFEM methods which model a strong discontinuity in the volumetric element using the phantom node method. This is the first known algorithm for XFEM which is able to robustly and efficiently insert discontinuities into the mesh.

4.2   Limitations

There appears to be instability in the adaptive insertion of the GCEs depending on the insertion criterion used. While this instability can be moderated by changing the stress at which the insertion is conducted, it is unknown how much this will affect more complex geometries and loading conditions.

This insertion stability may be due to the particular insertion criterion used. While the maximum principal stress criterion has the benefit of simplicity and speed, the criterion has no knowledge of the stress state in the surrounding elements. A non-local stress criterion may be better able to anticipate both the optimal insertion time and cohesive surface orientation.

Additionally, literature has shown that there is the potential for numerical instability when modifications to the mesh topology are done during the course of an analysis. There is as of yet no widely accepted method for mitigating this instability. Further study is required to determine which of the several methods available yield the best behavior.

4.3   Future opportunities

The simplicial graph representation used in this work is not limited to the 2D analyses which have been used. As shown in the literature, graph representations of this type

may be used for N-dimensional simplicial meshes. Expanding the method so insertion of GCEs could be conducted in 3D is an obvious step.

Another possible path of further research is in mitigating the instability due to insertion of the generalized cohesive elements in particular and arbitrary topological changes in general. In particular, alternative insertion criteria which are based on a non-local stress measure could be investigated.

It was mentioned in Chapter 2 that the GCE method may be expanded to cracks on the interior of the volumetric element. A preliminary method for this process is described briefly in Appendix B. The inclusion of branching in the GCE method would allow for more general computational fracture problems, especially fracture in bodies with extremely high loading rates. Even for more sedate loading scenarios, branching is a potential method for energy dissipation and therefore suggests an important improvement over the current work.

Finally, this work, along with the published work available at this time formulate the graph structure with simplicial meshes. In practice however, many analyses are done using non-simplicial meshes (e.g. brick elements in $\mathbb{R}^2$ and $\mathbb{R}^3$). For this reason, a formulation which is valid for arbitrary polyhedra of any dimension would be of interest if this method is to be used widely for fracture analysis.

References

Adda-Bedia, M., 2004. Brittle fracture dynamics with arbitrary paths. II. dynamic crack branching under general antiplane loading. Journal of the Mechanics and Physics of Solids 52, 1407–1420.

Antonio, F., 1992. Faster line segment intersection. In: Kirk, D. (Ed.), Graphics Gems III. Academic Press, Cambridge, MA.

Armero, F., Linder, C., 2009. Numerical simulation of dynamic fracture using finite elements with embedded discontinuities. International Journal of Fracture 160, 119–141.

Barenblatt, G. I., 1962. The mathematical theory of equilibrium of cracks in brittle fracture. Advances in Applied Mechanics 7, 55–129.

Baylor, J. S., 1997. A numerical simulation of impact-induced damage of composite materials. Master's thesis, University of Illinois at Urbana-Champaign, Urbana, Illinois.

Beall, M. W., Shephard, M. S., 1997. A general topology-based mesh data structure. International Journal for Numerical Methods in Engineering 40, 1573–1596.

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

Belytschko, T., Chen, H., Xu, J., Zi, G., 2003. Dynamic crack propagation based on loss of hyperelasticity and a new discontinuous enrichment. International Journal for Numerical Methods in Engineering 58, 1873–1905.

Belytschko, T., Lui, W. K., Moran, B., 2000. Nonlinear Finite Elements for Continua and Structures. John Wiley and Sons.

Brochu, T., Bridson, R., 2009. Robust topological operations for dynamic explicit surfaces. SIAM Journal on Scientific Computing 31 (4), 2472–2493.

Celes, W., Paulino, G. H., Espinha, R., 2005a. A compact adjacency-based topological data structure for finite element mesh representation. International Journal for Numerical Methods in Engineering 64 (11), 1529–1556.

Celes, W., Paulino, G. H., Espinha, R., 2005b. Efficient handling of implicit entities in reduced mesh representations. Journal of Computing and Information Science in Engineering 5 (4), 348–359.

Dooley, I., Mangala, S., Kale, L., Geubelle, P., 2009. Parallel simulations of dynamic fracture using extrinsic cohesive elements. Journal of Scientific Computing 39 (1), 144–165.

Dugdale, D. S., 1960. Yeilding of steel sheets containing slits. Journal of the Mechanics and Physics of Solids 8, 100–104.

Fineberg, J., Gross, S. P., Marder, M., Swinney, H. L., Jul 1991. Instability in dynamic fracture. Physical Review Letters 67 (4), 457–460.

Fineberg, J., Gross, S. P., Marder, M., Swinney, H. L., 1992. Instability in the propagation of fast cracks. Physical Review B 45 (10), 5146–5154.

Fineberg, J., Marder, M., May 1999. Instability in dynamic fracture. Physics Reports - Review Section of Physics Letters 313 (1-2), 2–108.

Ganguly, P., Vavasis, S. A., Papoulia, K. D., 2006. An algorithm for two-dimentional mesh generation based on the pin-wheel tiling. SIAM Journal on Scientific Computing 28 (4), 1533–1562.

Geubelle, P. H., Baylor, J. S., 1998. Impact-induced delamination of composites: a 2d simulation. Composites Part B-Engineering 29 (5), 589–602.

Griffith, A. A., 1920. The phenomenon of rupture and flow in solids. Philisophical Transactions of the Royal Society of London 221.

Griffiths, B., 2005. Boeing sets pace for composite usage in large civil aircraft. High-Performance Composites.

Hansbo, A., Hansbo, P., 2004. A finite element method for the simulation of strong and weak discontinuities in solid mechanics. Computer Methods in Applied Mechanics and Engineering 193 (33–35), 3523–3540.

Hohler, V., Stilp, A. J., Weber, K., 1995. Hypervelocity penetration of tungsten sinter-alloy rods into aluminum. International Journal of Impact Engineering 17, 409–418.

Hsieh, S.-H., Paulino, G. H., Abel, J. F., 1995. Recursive spectral algorithms for automatic domain partitioning in parallel finite element analysis. Computer Methods in Applied Mechanics and Engineering 121 (1–4), 137–162.

Kalthoff, J. F., Winkler, S., 1987. Falure mode transition at high rates of shear loading. International Conference on Impact Loading and Dynamic Behavior of Materials 1, 185–195.

Katzav, E., Adda-Bedia, M., Arias, R., 2007. Theory of dynamic crack branching in brittle materials. International Journal of Fracture 143, 245–271.

Klein, P. A., Foulk, J. W., Chen, E. P., Wimmer, S. A., Gao, H. J., 2001. Physics-based modeling of brittle fracture: cohesive formulations and the application of meshfree methods. Theoretical and Applied Fracture Mechanics 37 (1–3), 99–166.

Kozhushko, A. A., Rykova, I. I., Sinani, A. B., 1991. Resistance of ceramics to penetration at impact velocities above 5 km/s. Journal De Physique IV 1, 117.

Linder, C., Armero, F., 2007. Finite elements with embedded strong discontinuities for the modeling of failure in solids. International Journal for Numerical Methods in Engineering 72, 1391–1433.

Linder, C., Armero, F., 2009. Finite elements with embedded branching. Finite Elements in Analysis and Design 45 (4), 280–293, the Twentieth Annual Robert J. Melosh Competition.

Linder, C., Rosato, D., Miehe, C., 2011. New finite elements with embedded strong discontinuities for the modeling of failure in electromechanical coupled solids. Computer Methods in Applied Mechanics and Engineering 200, 141–161.

Livne, A., Ben-David, O., Fineberg, J., Mar 2007. Oscillations in rapid fracture. Physical Review Letters 98 (12), 124301.

Maiti, S., 2002. Grain-level simulation of dynamic failure in ceramic materials. Ph.D. thesis, University of Illinois at Urbana-Champaign, Urbana, Illinois.

Maiti, S., Geubelle, P. H., 2004. Mesoscale modeling of dynamic fracture of ceramic materials. Computer Modeling in Engineering and Science 5, 91–101.

Maiti, S., Ghosh, D., Subhash, G., 2009. A generalized cohesive element technique for arbitrary crack motion. Finite Elements in Analysis and Design 45 (8–9), 501–510.

Maunder, C. R. F., 1970. Algebraic Topology. The New University Mathematics Series. Van Nostrand Reinhold Company, London.

Mayer, J., 1972. Algebraic Topology. Prentice-Hall, Englewood Cliffs, New Jersey.

Melenk, J. M., Babuska, I., 1996. The partition of unity finite element method: Basic theory and applications. Computer Methods in Applied Mechanics and Engineering 139 (1–4), 289–314.

Mergheim, J., Kuhl, E., Steinmann, P., 2005. A finite element method for the computational modelling of cohesive cracks. International Journal for Numerical Methods in Engineering 63 (2), 276–289.

Moës, N., Dolbow, J., Belytschko, T., 1999. A finite element method for crack growth without remeshing. International Journal for Numerical Methods in Engineering 46 (1), 131–150.

Mota, A., Knap, J., Ortiz, M., 2008. Fracture and fragmentation of simplicial finite element meshes using graphs. International Journal for Numerical Methods in Engineering 73, 1547–1570.

Munkres, J., 1984. Elements of Algebraic Topology. Addison-Wesley Publishing Company, Menlo Park, California.

Pandolfi, A., Krysl, P., Ortiz, M., 1999. Finite element simulation of ring expansion and fragmentation: The capturing of length and time scales through cohesive models of fracture. International Journal of Fracture 95, 279–297.

Pandolfi, A., Ortiz, M., 2002. An efficient adaptive procedure for three-dimentional fragmentation simulations. Engineering with Computers 18, 148–159.

Papoulia, K. D., Vavasis, S. A., Ganguly, P., Jul 2 2006. Spatial convergence of crack nucleation using a cohesive finite-element model on a pinwheel-based mesh. International Journal for Numerical Methods in Engineering 67 (1), 1–16.

Park, K., 2009. Potential-based fracture mechanics using cohesive zone and virtual internal bond modeling. Ph.D. thesis, University of Illinois at Urbana-Champaign, Urbana, Illinois.

Paulino, G. H., Celes, W., Espinha, R., Zhang, Z., March 2008. A general topology-based framework for adaptive insertion of cohesive elements in finite element meshes. Engineering with Computers 24 (1), 59–78.

Paulino, G. H., Menezes, I. F. M., Gattass, M., Mukherjee, S., 1994. Node and element resequencing using the laplacian of a finite element graph: Part I – general concepts and algorithm. International Journal for Numerical Methods in Engineering 37 (9), 1511–1530.

Paulino, G. H., Park, K., Celes, W., Espinha, R., 2010. Adaptive dynamic cohesive fracture simulation using nodal perturbation and edge-swap operators. International Journal for Numerical Methods in Engineering 84, 1303–1343.

Rabczuk, T., Zi, G., Gerstenberger, A., Wall, W. A., 2008. A new crack tip element for the phantom-node method with arbitrary cohesive cracks. International Journal for Numerical Methods in Engineering 75 (5), 577–599.

Radin, C., 1994. The pinwheel tiling of the plane. Annals of Mathematics 139 (3), 661–702.

Radin, C., Sadun, L., 1996. The isoperimetric problem for pinwheel meshes. Communications in Mathematical Physics 177, 255–263.

Rice, J. R., 1968. A path independant integral and the approximate analysis of strain concentration by notches and cracks. Journal of Applied Mechanics 35, 379–386.

Sam, C.-H., 2005. A robust formulation and solution of initally rigid cohesive interface models. Ph.D. thesis, Cornell Unversity.

Shaffer, C. A., Feustel, Charles, D., 1992. Exact computation of 2-d intersections. In: Kirk, D. (Ed.), Graphics Gems III. Academic Press, Cambridge, MA.

Sharon, E., Fineberg, J., 1996. Microbranching instability and the dynamic fracture of brittle materials. Physical Review B 54 (10), 7128–7139.

Sharon, E., Fineberg, J., 1999. Confirming the continuum theory of dynamic brittle fracture for fast cracks. Nature 397, 333–335.

Sharon, E., Gross, S. P., Fineberg, J., 1995. Local branching as a mechanism for instability in dynamic fracture. Physical Review Letters 74 (25), 5096–5099.

Shewchuk, J. R., May 1996. Triangle: engineering a 2D quality mesh generator and Delaunay triangulator. In: Lin, M. C., Manocha, D. (Eds.), Applied Computational Geometry: Towards Geometric Engineering. Vol. 1148 of Lecture Notes in Computer Science. Springer-Verlag, pp. 203–222, from the First ACM Workshop on Applied Computational Geometry.

Song, J. H., Areias, P. M., Belytschko, T., 2006a. A method for dynamic crack and shear band propagation with phantom nodes. International Journal for Numerical Methods in Engineering 67, 868–893.

Song, S. H., Paulino, G. H., Buttlar, W. G., 2006b. A bilinear cohesive zone model tailored for fracture of asphalt concrete considering viscoelastic bulk material. Engineering Fracture Mechanics 73 (18), 2829–2848.

Sumramanian, R., Bless, S. J., 1995. Penetration of semi-infinite AD995 alumina targets by tungsten long rod penetrators from 1.5 to 3.5 km/s. International Journal of Impact Engineering 17, 807–816.

van der Meer, F., Sluys, L., 2009. A phantom node formulation with mixed mode cohesive law for splitting in laminates. International Journal of Fracture 158, 107–124, 10.1007/s10704-009-9344-5.

Wallace, A. H., 1963. An Introduction to Algebraic Topology. Vol. 1 of International Series of Monographs in Pure and Applied Mathematics. Pergamon Press, London.

Wells, G. N., Sluys, L. J., 2001. A new methods for modelling cohesive cracks using finite elements. International Journal for Numerical Methods in Engineering 50, 2667–2682.

Zhang, Z. J., Paulino, G. H., 2005. Cohesive zone modeling of dynamic failure in homogeneous and functionally graded material. International Journal of Plasticity 21, 1195–1254.

Zhang, Z. J., Paulino, G. H., Celes, W., 2007. Extrinsic cohesive modelling of dynamic fracture and microbranching instability in brittle materials 72 (8), 893–923.

Appendix A

Simplicial Complexes

This appexdix has been created using information from a number of texts on algebraic topology and simplicial complexes. For a more rigorous description of simplices and simplicial complexes or for more information, refer to these texts; Mayer (1972); Maunder (1970); Munkres (1984); Wallace (1963).

A simplex is defined in terms of the set of polyhedra. Within this set are polyhedrons composed of lower order polyhedra such as line segments, triangles, tetrahedra, and their higher order analogues which are connected along their faces.

Let a be an independent set of points in $\mathbb{R}^m$. The n-simplex, $\sigma_n \in \mathbb{R}^m, n \leq m$ spanned by $a_1, a_2, \ldots, a_n$ is defined as the set of points all points x of $\mathbb{R}^m$ such that,

$$x = \sum_{i=0}^{n} \lambda_i a_i, \quad \text{where } \sum_{i=0}^{n} \lambda_i = 1 \tag{A.1}$$

From the above definition, the maximum dimension of all simplices in $\mathbb{R}^m$ is m. Thus, in $\mathbb{R}^3$ there are a possible four simplexes: the 0-simplex, 1-simplex, 2-simplex, and 3-simplex. These correspond to: vertex, line segment, triangle, and tetrahedron, respectively. The importance of the numbers λ_i, called the barycentric coordinates, will be discussed later in chapter.

The points $a_1, a_2, \ldots, a_n$ which span σ_n are called the vertices of σ_n. n is the dimension of σ.

Consider the simplex $\sigma_p = \{a_0 \ldots a_p\}$. There exists a simplex $\sigma_k = \{a_{i0} \ldots a_{ik}\}$ with $\{a_{i0} \ldots a_{ik}\} \subset \{a_0 \ldots a_p\}$. The simplex σ_k is called the face of σ_p. This shall be denoted by $\sigma_k = \text{face}(\sigma_p)$.

A simplicial complex K is a set of simplices which are subject to the following conditions.

1. For a simplex $\sigma_p \in K$, all faces $\sigma_q = \text{face}(\sigma_p) \in K$.

2. For any two simplices $\sigma_p, \sigma_q \in K$, $\sigma_p \cap \sigma_q = \sigma_r$ or $\sigma_p \cap \sigma_q = \emptyset$ where $\sigma_r = \text{face}(\sigma_p) = \text{face}(\sigma_q)$.

From the first of the conditions, if a simplex is in K, then the faces of the simplex are also in K. From the second, the intersection of two simplices in K is either the common face of both simplices or the empty set. With these two conditions and reviewing the definition of the face of a simplex, it is seen that a simplex is itself a simplicial complex. Thus a 3-simplex is a set of 2-simplices which is a set of 1-simplices, etc.

Figure A.1 show three sets of simplices. A.1(a) and A.1(b) are valid simplicial complexes while A.1(c) is not. In A.1(a) it is seen that the simplices of a simplicial complex are not required to be of the same degree. A 3-simplex is connected to a 2-simplex which is in turn connected to a 1-simplex. A.1(c) is not a simplicial complex because the intersection of the simplices is not a common face of the simplices.

(a)   (b)   (c)
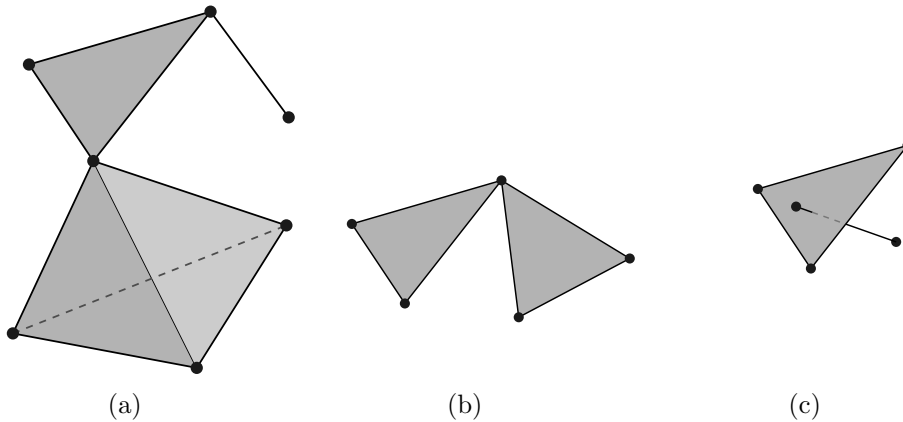
Figure A.1: The consequences of the definition of simplicial complexes

Three terms related to the region about a point in a simplicial complex are useful in the manipulation of simplices and simplicial complexes. These are the neighborhood,

star, and link. From Maunder (1970) these terms are defined as follows. For a point x in a simplicial complex K, the simplicial neighborhood, $N(x)$, of x is the set of simplices in K which contain x along with thier faces. The link of x, $Lk(x)$ is the subset of simplices in $N(x)$ which do not contain x. Finally, the star of a simplex $\sigma \in K$, $st(x)$, is the set of the interiors of the simplices of K for which σ is a face. The star may also be defined as $st(x) = N(\sigma) - Lk(\sigma)$.

Figure A.2 shows an example of the star and link of a 0-simplex. The center node is the simplex that the star and link are computed for. The black lines and grey triangles along with the node itself comprise the star of the simplex. The Green segments and nodes represent the link.
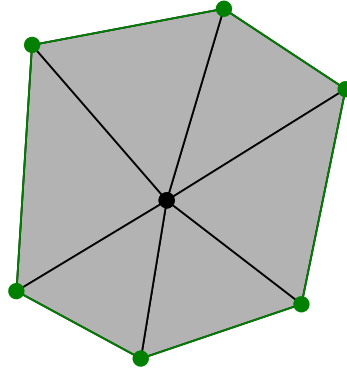
Figure A.2: Star and link of a simplex

Let us use the idea of a simplex to describe some useful engineering calculations. A barycenter is the point on a body at which it is balanced. That is, if a body of constant density is suspended from one point, the barycenter is the point at which the body is in equilibrium. For a general simplex, the barycenter, $\hat{\sigma}_n$, of a simplex $\sigma_n \in \mathbb{R}^m$ is equal to,

$$\hat{\sigma}_n = \frac{1}{n+1} \sum_{i=0}^{n} a_i, \; a_i = (x_0, x_1, \ldots, x_m) \tag{A.2}$$

More specifically, the barycenter of a line segment is given by,

$$\hat{\sigma}_1 = \frac{1}{2}(a_0 + a_1) \tag{A.3}$$

This equation yield the midpoint of the line segment.

For a 2-simplex or triangle, the equation becomes,

$$\hat{\sigma}_2 = \frac{1}{3}(a_0 + a_1 + a_2) \tag{A.4}$$

While the above equations may seem trivial, they are a good introduction to the idea of a barycentric coordinate system. Like the name suggests, barycentric coordinates are related to the idea of a barycenter. Given an arbitrary point on a body, the barycentric coordinates of that point yield the ratio of the point masses which would have to be added to the nodes of the body such that the point of interest becomes the balance point. Consider the bar shown in Figure A.3.
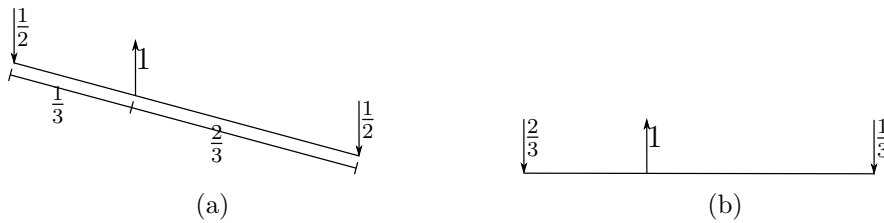
Figure A.3: The barycentric coordinates of a point "balance" the simplex

If the weight is evenly lumped at the nodes of the bar, each node will have $1/2$ of the total mass. This is equivalent to a bar of constant density. Hanging from the $1/3$ point however, the right side of the bar will be lower than the left as the moments at the left and right sides summed about the $1/3$ point are not the same (Figure A.3(a)). To be balanced, the two moments must be the same. This occurs when the forces at the left and right sides of the bar are at $2/3$ and $1/3$ of the total force, respectively (Figure A.3(b)). In all cases, the sum of the barycentric coordinates for a point will be equal to one. This is stated previously in Equation (A.1)

Revisiting the barycenter, for a 1-simplex, the barycenter will have barycentric coordinates of $\lambda_0 = 1/2, \lambda_1 = 1/2$ where λ_0, λ_1 are the barycentric coordinates of the first and second nodes of the simplex.

As an additional example, consider a point lying on the midpoint of an edge of a 2-simplex. Continuing the idea of the barycentric coordinates being the point masses which are required at the nodes of the simplex to balance the body, it is seen that the sum of the masses of the two nodes on the edge with the point will be one while the point mass at the third point will be equal to zero. Thus if the nodes on the edge on which the point is situated are n_0 and n_1, the barycentric coordinates of the point are given by $\lambda = (1/2, 1/2, 0)$. This is seen graphically in Figure A.4.
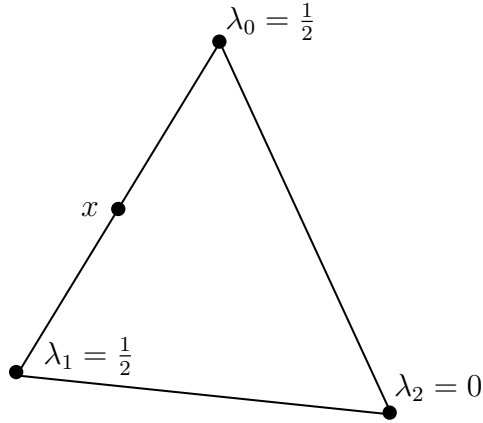
Figure A.4: Barycentric coordinate example

The p-skeleton of a simplicial complex K is defined as the set of simplices of K which are of dimension at most p. Thus $K^{(1)}$ is the set of all 1-simplices and 0-simplices in K. The special p-skeleton, $K^{(0)}$ is the set of vertices of K. This set shall be called V. The set of line segments, E, is given by $E = K^{(1)} \setminus K^{(0)} = \left\{ \sigma \in K^{(1)} | \sigma \notin K^{(0)} \right\}$.

An ordered simplex σ_n, $n \geq 1$ may be either positive or negative depending on the order of the vertices of the simplex. $+\sigma_n = +\{a_0, a_1, ..., a_n\}$ shall denote the class of positive permutations while $-\sigma_n = -\{a_0, a_1, ..., a_n\}$ shall denote the class of negative permutations.

Positive and negative permutations of the vertices will follow the convention used in finite element methods. E.g. one of the possible positive and negative permutations of a 2-simplex σ_2 shown in Figure A.5
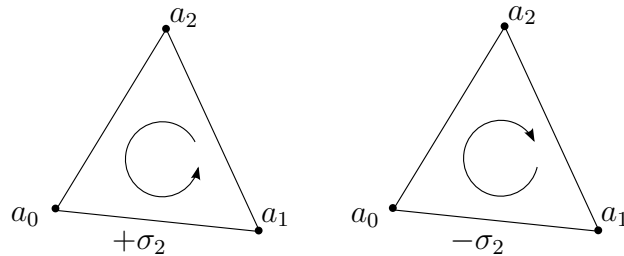
Figure A.5: Positive and negative simplex orientation

The faces of a simplex may also have a positive or negative orientation. For the current work, the highest order simplices are the 2-simplex. The faces of these simplices (the line segments or 1-simplices) shall have a positive orientation if the vertex ordering of the face is consistent with the vertex ordering of the simplex. Thus from Figure A.5, face $\{a_0, a_1\}$ is positive on $+\sigma_2$ while negative on $-\sigma_2$.

73

Appendix B

Branching

Crack branching is an important aspect of dynamic crack propagation. One of the main benefits of the cohesive element method over competing theories has been the relative ease in which cohesive elements can capture branching phenomena. As the conventional cohesive elements are situated at the boundaries of the volumetric elements, the crack may branch at will at any node of the mesh with an attached cohesive element.

Experimental studies of dynamic fracture of brittle materials have determined some factors which seem to govern the behavior of cracks in this type of analysis. For thin brittle materials, the maximum crack speed seen is between $60\% - 89\%$ of the Rayleigh wave speed v_r (Fineberg et al. (1991); Fineberg and Marder (1999); Livne et al. (2007).

Fineberg et al. (1992); Sharon et al. (1995); Sharon and Fineberg (1996, 1999) have noticed a critical crack velocity, v_c, above which crack branching occurs. Independent of specimen geometry and loading as well as the atmospheric conditions, v_c for poly(methyl methacrylate) (PMMA) is $v_c \approx .35v_R$ where v_R is the Rayleigh wave speed ($v_c = 330 \pm 30$m/sFineberg et al. (1992)). Livne et al. (2007) suggests that for some materials, the critical crack speed is determined by the shear wave speed instead of the Rayleigh wave speed. In thin brittle gels, critical crack speeds of $v_c = 0.87v_s$, with v_s the shear wave speed, were seen. Below v_c, the crack propagates along a smooth path. Crack speeds in excess of v_c result in oscillations along the crack path due to micro-branching in the material. Additionally, experimental testing by Fineberg et al. (1992) show a crack angle of approximately $30°$ in PMMA.

Theoretical work by Katzav et al. (2007) show that for brittle materials, the branching

criterion depends on the material properties and a branching angle of 27° should be seen. With a specimen under anti-plane loading, Adda-Bedia (2004) computes a somewhat larger crack angle of 40°. Also in the paper is a theoretical minimum critical crack velocity of $v_c = 0.39 v_s$. These results mirror the experimental results discussed above.

Numerical testing by Zhang et al. (2007) and Paulino et al. (2010) show that the cohesive element method can accurately model the behavior seen in testing by Sharon et al. (1995). This work, utilizing the cracking particle method notes the existence of a critical crack velocity above which branching occurs, a maximum crack speed less than the Rayleigh wave speed, and a larger number of cracks with higher crack velocities.

Current work in literature, described above, list two important factors in branching of a crack in dynamic analyses of brittle materials. First, there exists a crack velocity above which branches will form. Secondly, the angle of the branch is 27°. In analyses with conventional cohesive elements, a branch may form automatically depending on the stress state in the mesh. For both pre-inserted and adaptively inserted cohesive elements, branching along the boundaries of the volumetric elements will act to limit the crack velocity. In the current implementation of the GCE method however, the volumetric elements are artificially constrained to only one possible GCE. For this reason, a branch can not form unless the node of a GCE falls on a volumetric element node.

We would like to allow for the formation of a branch within the GCE framework. The insertion criterion used is dependent on the crack velocity. When the velocity reaches a percentage of the Rayleigh wave speed ($v = 0.35 v_R$ for PMMA Sharon et al. (1995)), a branch is created. The branch consists of two GCEs within the volumetric element which originates from the node of the previous GCE and extends through the volumetric element. Instead of a branch angle of 27° as seen in experimental tests by Sharon et al. (1995) and predicted by Katzav et al. (2007), the edges of the branch will be situated on the midpoints of the remaining volumetric element edges. Figure B.1 describes the branch insertion process. As in previous figures, grey areas denote material while white areas denote imaginary areas of the element. The insertion process is completed in two steps. Each step is completed in the same manner as the other GCEs in the mesh. First a GCE is inserted into the volumetric element in the same manner as a non-branching GCE. A second GCE is then inserted into the child element from the first step which retained an original edge.

At the end of the insertion process, there are three child elements, each of which retain one of the original volumetric element nodes (denoted as solid circles in the
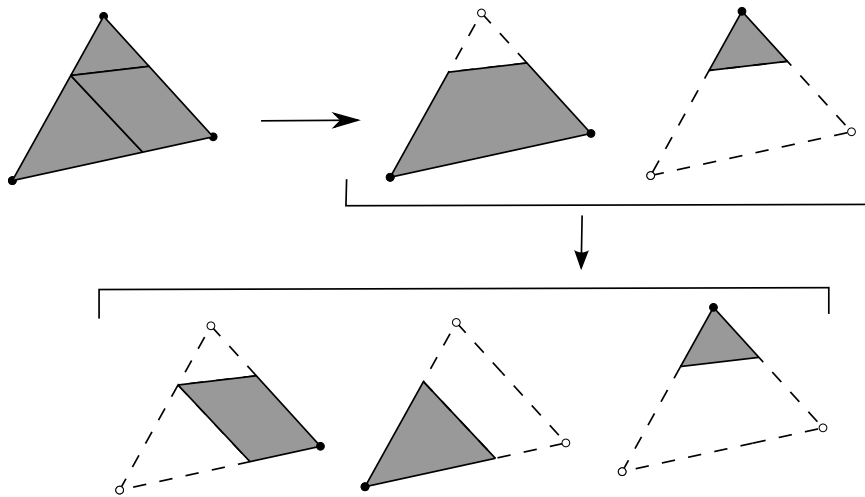
Figure B.1: A volumetric element with included branch. Each leg of the branch is a GCE. The insertion process is completed in two steps.

figure).

Appendix C

Copyright

C.1   Copyright information for Figure 1.1

This is a License Agreement between James R Thunes ("You") and John Wiley and Sons ("John Wiley and Sons") provided by Copyright Clearance Center ("CCC"). The license consists of your order details, the terms and conditions provided by John Wiley and Sons, and the payment terms and conditions.

All payments must be made in full to CCC. For payment instructions, please see information listed at the bottom of this form.

License Number	2460381149607
License date	Jul 01, 2010
Licensed content publisher	John Wiley and Sons
Licensed content publication	International Journal for Numerical Methods in Engineering
Licensed content title	Spatial convergence of crack nucleation using a cohesive finite-element model on a pinwheel-based mesh
Licensed content author	Papoulia Katerina D., Vavasis Stephen A., Ganguly Pritam
Licensed content date	Jan 18, 2006
Start page	1
End page	16
Type of use	Dissertation/Thesis
Requestor type	University/Academic

Format	Print
Portion	Figure/table
Number of figures/table	1
Original Wiley figure/table number(s)	Figure 1
Will you be translating	No
Order reference number	
Total	0.00

C.1.1   Terms and conditions

This copyrighted material is owned by or exclusively licensed to John Wiley & Sons, Inc. or one if its group companies (each a Wiley Company) or a society for whom a Wiley Company has exclusive publishing rights in relation to a particular journal (collectively WILEY). By clicking accept in connection with completing this licensing transaction, you agree that the following terms and conditions apply to this transaction (along with the billing and payment terms and conditions established by the Copyright Clearance Center Inc., (CCC's Billing and Payment terms and conditions), at the time that you opened your Rightslink account (these are available at any time at `http://myaccount.copyright.com`).

Terms and Conditions

1. The materials you have requested permission to reproduce (the "Materials") are protected by copyright.

2. You are hereby granted a personal, non-exclusive, non-sublicensable, non-transferable, worldwide, limited license to reproduce the Materials for the purpose specified in the licensing process. This license is for a one-time use only with a maximum distribution equal to the number that you identified in the licensing process. Any form of republication granted by this licence must be completed within two years of the date of the grant of this licence (although copies prepared before may be distributed thereafter). Any electronic posting of the Materials is limited to one year from the date permission is granted and is on the condition that a link is placed to the journal homepage on Wiley's online journals publication platform at `www.interscience.wiley.com`. The Materials shall not be used in any other manner or for any other purpose. Permission is granted subject to an appropriate acknowledgement given to the author, title of the material/book/journal and the publisher and on the understanding that nowhere

in the text is a previously published source acknowledged for all or part of this Material. Any third party material is expressly excluded from this permission.

3. With respect to the Materials, all rights are reserved. No part of the Materials may be copied, modified, adapted, translated, reproduced, transferred or distributed, in any form or by any means, and no derivative works may be made based on the Materials without the prior permission of the respective copyright owner. You may not alter, remove or suppress in any manner any copyright, trademark or other notices displayed by the Materials. You may not license, rent, sell, loan, lease, pledge, offer as security, transfer or assign the Materials, or any of the rights granted to you hereunder to any other person.

4. The Materials and all of the intellectual property rights therein shall at all times remain the exclusive property of John Wiley & Sons Inc or one of its related companies (WILEY) or their respective licensors, and your interest therein is only that of having possession of and the right to reproduce the Materials pursuant to Section 2 herein during the continuance of this Agreement. You agree that you own no right, title or interest in or to the Materials or any of the intellectual property rights therein. You shall have no rights hereunder other than the license as provided for above in Section 2. No right, license or interest to any trademark, trade name, service mark or other branding ("Marks") of WILEY or its licensors is granted hereunder, and you agree that you shall not assert any such right, license or interest with respect thereto.

5. WILEY DOES NOT MAKE ANY WARRANTY OR REPRESENTATION OF ANY KIND TO YOU OR ANY THIRD PARTY, EXPRESS, IMPLIED OR STATUTORY, WITH RESPECT TO THE MATERIALS OR THE ACCURACY OF ANY INFORMATION CONTAINED IN THE MATERIALS, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTY OF MERCHANTABILITY, ACCURACY, SATISFACTORY QUALITY, FITNESS FOR A PARTICULAR PURPOSE, USABILITY, INTEGRATION OR NON-INFRINGEMENT AND ALL SUCH WARRANTIES ARE HEREBY EXCLUDED BY WILEY AND WAIVED BY YOU.

6. WILEY shall have the right to terminate this Agreement immediately upon breach of this Agreement by you.

7. You shall indemnify, defend and hold harmless WILEY, its directors, officers, agents and employees, from and against any actual or threatened claims, demands, causes of action or proceedings arising from any breach of this Agreement by you.

8. IN NO EVENT SHALL WILEY BE LIABLE TO YOU OR ANY OTHER PARTY OR ANY OTHER PERSON OR ENTITY FOR ANY SPECIAL,

CONSEQUENTIAL, INCIDENTAL, INDIRECT, EXEMPLARY OR PUNITIVE DAMAGES, HOWEVER CAUSED, ARISING OUT OF OR IN CONNECTION WITH THE DOWNLOADING, PROVISIONING, VIEWING OR USE OF THE MATERIALS REGARDLESS OF THE FORM OF ACTION, WHETHER FOR BREACH OF CONTRACT, BREACH OF WARRANTY, TORT, NEGLIGENCE, INFRINGEMENT OR OTHERWISE (INCLUDING, WITHOUT LIMITATION, DAMAGES BASED ON LOSS OF PROFITS, DATA, FILES, USE, BUSINESS OPPORTUNITY OR CLAIMS OF THIRD PARTIES), AND WHETHER OR NOT THE PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION SHALL APPLY NOTWITHSTANDING ANY FAILURE OF ESSENTIAL PURPOSE OF ANY LIMITED REMEDY PROVIDED HEREIN.

9. Should any provision of this Agreement be held by a court of competent jurisdiction to be illegal, invalid, or unenforceable, that provision shall be deemed amended to achieve as nearly as possible the same economic effect as the original provision, and the legality, validity and enforceability of the remaining provisions of this Agreement shall not be affected or impaired thereby.

10. The failure of either party to enforce any term or condition of this Agreement shall not constitute a waiver of either party's right to enforce each and every term and condition of this Agreement. No breach under this agreement shall be deemed waived or excused by either party unless such waiver or consent is in writing signed by the party granting such waiver or consent. The waiver by or consent of a party to a breach of any provision of this Agreement shall not operate or be construed as a waiver of or consent to any other or subsequent breach by such other party.

11. This Agreement may not be assigned (including by operation of law or otherwise) by you without WILEY's prior written consent.

12. These terms and conditions together with CCC's Billing and Payment terms and conditions (which are incorporated herein) form the entire agreement between you and WILEY concerning this licensing transaction and (in the absence of fraud) supersedes all prior agreements and representations of the parties, oral or written. This Agreement may not be amended except in a writing signed by both parties. This Agreement shall be binding upon and inure to the benefit of the parties' successors, legal representatives, and authorized assigns.

13. In the event of any conflict between your obligations established by these terms and conditions and those established by CCC's Billing and Payment terms and conditions, these terms and conditions shall prevail.

14. WILEY expressly reserves all rights not specifically granted in the combination of (i) the license details provided by you and accepted in the course of this

licensing transaction, (ii) these terms and conditions and (iii) CCC's Billing and Payment terms and conditions.

15. This Agreement shall be governed by and construed in accordance with the laws of England and you agree to submit to the exclusive jurisdiction of the English courts.

16. Other Terms and Conditions:

   BY CLICKING ON THE "I ACCEPT" BUTTON, YOU ACKNOWLEDGE THAT YOU HAVE READ AND FULLY UNDERSTAND EACH OF THE SECTIONS OF AND PROVISIONS SET FORTH IN THIS AGREEMENT AND THAT YOU ARE IN AGREEMENT WITH AND ARE WILLING TO ACCEPT ALL OF YOUR OBLIGATIONS AS SET FORTH IN THIS AGREEMENT.

V1.2

Gratis licenses (referencing $0 in the Total field) are free. Please retain this printable license for your reference. No payment is required.

If you would like to pay for this license now, please remit this license along with your payment made payable to "COPYRIGHT CLEARANCE CENTER" otherwise you will be invoiced within 48 hours of the license date. Payment should be in the form of a check or money order referencing your account number and this invoice number RLNK0. Once you receive your invoice for this order, you may pay your invoice by credit card. Please follow instructions provided at that time.

Make Payment To: Copyright Clearance Center Dept 001 P.O. Box 843006 Boston, MA 02284-3006

For suggestions or comments regarding this order, contact Rightslink Customer Support: customercare@copyright.com or +1-877-622-5543 (toll free in the US) or +1-978-646-2777.

C.2   Copyright information for Figure 1.2

This is a License Agreement between James R Thunes ("You") and John Wiley and Sons ("John Wiley and Sons") provided by Copyright Clearance Center ("CCC").

The license consists of your order details, the terms and conditions provided by John Wiley and Sons, and the payment terms and conditions.

All payments must be made in full to CCC. For payment instructions, please see information listed at the bottom of this form.

License Number	2638341163081
License date	Mar 29, 2011
Licensed content publisher	John Wiley and Sons
Licensed content publication	International Journal for Numerical Methods in Engineering
Licensed content title	Adaptive dynamic cohesive fracture simulation using nodal perturbation and edgeswap operators
Licensed content author	Glaucio H. Paulino, Kyoungsoo Park, Waldemar Celes, Rodrigo Espinha
Licensed content date	Dec 10, 2010
Start page	1303
End page	1343
Type of use	Dissertation/Thesis
Requestor type	University/Academic
Format	Print
Portion	Figure/table
Number of figures/tables	1
Number of extracts	
Original Wiley figure/table number(s)	Figure 9
Will you be translating?	No
Order reference number	
Total	0.00 USD
Terms and Conditions	

C.2.1 Terms and Conditions

This copyrighted material is owned by or exclusively licensed to John Wiley & Sons, Inc. or one of its group companies (each a "Wiley Company") or a society for whom a Wiley Company has exclusive publishing rights in relation to a particular journal (collectively "WILEY"). By clicking "accept" in connection with completing this licensing transaction, you agree that the following terms and conditions apply to this transaction (along with the billing and payment terms and conditions established by the Copyright Clearance Center Inc., ("CCCs Billing and Payment terms and

conditions"), at the time that you opened your Rightslink account (these are available at any time at `http://myaccount.copyright.com`)

1. The materials you have requested permission to reproduce (the "Materials") are protected by copyright.

2. You are hereby granted a personal, non-exclusive, non-sublicensable, non-transferable, worldwide, limited license to reproduce the Materials for the purpose specified in the licensing process. This license is for a one-time use only with a maximum distribution equal to the number that you identified in the licensing process. Any form of republication granted by this licence must be completed within two years of the date of the grant of this licence (although copies prepared before may be distributed thereafter). The Materials shall not be used in any other manner or for any other purpose. Permission is granted subject to an appropriate acknowledgement given to the author, title of the material/book/journal and the publisher and on the understanding that nowhere in the text is a previously published source acknowledged for all or part of this Material. Any third party material is expressly excluded from this permission.

3. With respect to the Materials, all rights are reserved. Except as expressly granted by the terms of the license, no part of the Materials may be copied, modified, adapted (except for minor reformatting required by the new Publication), translated, reproduced, transferred or distributed, in any form or by any means, and no derivative works may be made based on the Materials without the prior permission of the respective copyright owner. You may not alter, remove or suppress in any manner any copyright, trademark or other notices displayed by the Materials. You may not license, rent, sell, loan, lease, pledge, offer as security, transfer or assign the Materials, or any of the rights granted to you hereunder to any other person.

4. The Materials and all of the intellectual property rights therein shall at all times remain the exclusive property of John Wiley & Sons Inc or one of its related companies (WILEY) or their respective licensors, and your interest therein is only that of having possession of and the right to reproduce the Materials pursuant to Section 2 herein during the continuance of this Agreement. You agree that you own no right, title or interest in or to the Materials or any of the intellectual property rights therein. You shall have no rights hereunder other than the license as provided for above in Section 2. No right, license or interest to any trademark, trade name, service mark or other branding ("Marks") of WILEY or its licensors is granted hereunder, and you agree that you shall not assert any such right, license or interest with respect thereto.

5. NEITHER WILEY NOR ITS LICENSORS MAKES ANY WARRANTY OR REPRESENTATION OF ANY KIND TO YOU OR ANY THIRD PARTY, EXPRESS, IMPLIED OR STATUTORY, WITH RESPECT TO THE MATERIALS OR THE ACCURACY OF ANY INFORMATION CONTAINED IN THE MATERIALS, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTY OF MERCHANTABILITY, ACCURACY, SATISFACTORY QUALITY, FITNESS FOR A PARTICULAR PURPOSE, USABILITY, INTEGRATION OR NON-INFRINGEMENT AND ALL SUCH WARRANTIES ARE HEREBY EXCLUDED BY WILEY AND ITS LICENSORS AND WAIVED BY YOU.

6. WILEY shall have the right to terminate this Agreement immediately upon breach of this Agreement by you.

7. You shall indemnify, defend and hold harmless WILEY, its Licensors and their respective directors, officers, agents and employees, from and against any actual or threatened claims, demands, causes of action or proceedings arising from any breach of this Agreement by you.

8. IN NO EVENT SHALL WILEY OR ITS LICENSORS BE LIABLE TO YOU OR ANY OTHER PARTY OR ANY OTHER PERSON OR ENTITY FOR ANY SPECIAL, CONSEQUENTIAL, INCIDENTAL, INDIRECT, EXEMPLARY OR PUNITIVE DAMAGES, HOWEVER CAUSED, ARISING OUT OF OR IN CONNECTION WITH THE DOWNLOADING, PROVISIONING, VIEWING OR USE OF THE MATERIALS REGARDLESS OF THE FORM OF ACTION, WHETHER FOR BREACH OF CONTRACT, BREACH OF WARRANTY, TORT, NEGLIGENCE, INFRINGEMENT OR OTHERWISE (INCLUDING, WITHOUT LIMITATION, DAMAGES BASED ON LOSS OF PROFITS, DATA, FILES, USE, BUSINESS OPPORTUNITY OR CLAIMS OF THIRD PARTIES), AND WHETHER OR NOT THE PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION SHALL APPLY NOTWITHSTANDING ANY FAILURE OF ESSENTIAL PURPOSE OF ANY LIMITED REMEDY PROVIDED HEREIN.

9. Should any provision of this Agreement be held by a court of competent jurisdiction to be illegal, invalid, or unenforceable, that provision shall be deemed amended to achieve as nearly as possible the same economic effect as the original provision, and the legality, validity and enforceability of the remaining provisions of this Agreement shall not be affected or impaired thereby.

10. The failure of either party to enforce any term or condition of this Agreement shall not constitute a waiver of either party's right to enforce each and every term and condition of this Agreement. No breach under this agreement shall be deemed waived or excused by either party unless such waiver or consent is

in writing signed by the party granting such waiver or consent. The waiver by or consent of a party to a breach of any provision of this Agreement shall not operate or be construed as a waiver of or consent to any other or subsequent breach by such other party.

11. This Agreement may not be assigned (including by operation of law or otherwise) by you without WILEY's prior written consent.

12. Any fee required for this permission shall be non-refundable after thirty (30) days from receipt.

13. These terms and conditions together with CCCs Billing and Payment terms and conditions (which are incorporated herein) form the entire agreement between you and WILEY concerning this licensing transaction and (in the absence of fraud) supersedes all prior agreements and representations of the parties, oral or written. This Agreement may not be amended except in writing signed by both parties. This Agreement shall be binding upon and inure to the benefit of the parties' successors, legal representatives, and authorized assigns.

14. In the event of any conflict between your obligations established by these terms and conditions and those established by CCCs Billing and Payment terms and conditions, these terms and conditions shall prevail.

15. WILEY expressly reserves all rights not specifically granted in the combination of (i) the license details provided by you and accepted in the course of this licensing transaction, (ii) these terms and conditions and (iii) CCCs Billing and Payment terms and conditions.

16. This Agreement will be void if the Type of Use, Format, Circulation, or Requestor Type was misrepresented during the licensing process.

17. This Agreement shall be governed by and construed in accordance with the laws of the State of New York, USA, without regards to such states conflict of law rules. Any legal action, suit or proceeding arising out of or relating to these Terms and Conditions or the breach thereof shall be instituted in a court of competent jurisdiction in New York County in the State of New York in the United States of America and each party hereby consents and submits to the personal jurisdiction of such court, waives any objection to venue in such court and consents to service of process by registred or certified mail, return receipt requested, at the last known address of such party. . BY CLICKING ON THE "I ACCEPT" BUTTON, YOU ACKNOWLEDGE THAT YOU HAVE READ AND FULLY UNDERSTAND EACH OF THE SECTIONS OF AND PROVISIONS SET FORTH IN THIS AGREEMENT AND THAT YOU ARE IN AGREEMENT WITH AND ARE WILLING TO ACCEPT ALL OF YOUR OBLIGATIONS AS SET FORTH IN THIS AGREEMENT.

v1.4

Gratis licenses (referencing $0 in the Total field) are free. Please retain this printable license for your reference. No payment is required.

If you would like to pay for this license now, please remit this license along with your payment made payable to "COPYRIGHT CLEARANCE CENTER" otherwise you will be invoiced within 48 hours of the license date. Payment should be in the form of a check or money order referencing your account number and this invoice number RLNK10959372. Once you receive your invoice for this order, you may pay your invoice by credit card. Please follow instructions provided at that time.

Make Payment To: Copyright Clearance Center Dept 001 P.O. Box 843006 Boston, MA 02284-3006

For suggestions or comments regarding this order, contact Rightslink Customer Support: customercare@copyright.com or +1-877-622-5543 (toll free in the US) or +1-978-646-2777.

C.3   Copyright information for Figure 2.3

This is a License Agreement between James R Thunes ("You") and Elsevier ("Elsevier") provided by Copyright Clearance Center ("CCC"). The license consists of your order details, the terms and conditions provided by Elsevier, and the payment terms and conditions.

All payments must be made in full to CCC. For payment instructions, please see information listed at the bottom of this form.

Supplier	Elsevier Limited
	The Boulevard,Langford Lane
	Kidlington,Oxford,OX5 1GB,UK
Registered Company Number	1982084
Customer name	James R Thunes
Customer address	1111 Houghton Ave
	Houghton, MI 49931
License number	2635490828833
License date	Mar 24, 2011
Licensed content publisher	Elsevier

Licensed content publication	Finite Elements in Analysis and Design
Licensed content title	A generalized cohesive element technique for arbitrary crack motion
Licensed content author	Spandan Maiti, Dipankar Ghosh, Ghatu Subhash
Licensed content date	June-July 2009
Licensed content volume number	45
Licensed content issue number	8–9
Number of pages	10
Start Page	501
End Page	510
Type of Use	reuse in a thesis/dissertation
Portion	figures/tables/illustrations
Number of figures/tables/illustrations	1
Format	print
Are you the author of this Elsevier article?	No
Will you be translating?	No
Order reference number	
Title of your thesis/dissertation	A GENERALIZED METHOD FOR THE SIMULATION OF FRACTURE IN LARGE FINITE ELEMENT MESHES
Expected completion date	Apr 2011
Estimated size (number of pages)	70
Elsevier VAT number	GB 494 6272 12
Permissions price	0.00 USD
VAT/Local Sales Tax	0.0 USD / 0.0 GBP
Total	0.00 USD
Terms and Conditions	

C.3.1 Introduction

1 The publisher for this copyrighted material is Elsevier. By clicking "accept" in connection with completing this licensing transaction, you agree that the following terms and conditions apply to this transaction (along with the Billing and Payment terms and conditions established by Copyright Clearance Center, Inc. ("CCC"), at the time that you opened your Rightslink account and that are available at any time at `http://myaccount.copyright.com`).

C.3.2   General terms

2 Elsevier hereby grants you permission to reproduce the aforementioned material subject to the terms and conditions indicated.

3 Acknowledgement: If any part of the material to be used (for example, figures) has appeared in our publication with credit or acknowledgement to another source, permission must also be sought from that source. If such permission is not obtained then that material may not be included in your publication/copies. Suitable acknowledgement to the source must be made, either as a footnote or in a reference list at the end of your publication, as follows:

"Reprinted from Publication title, Vol /edition number, Author(s), Title of article / title of chapter, Pages No., Copyright (Year), with permission from Elsevier [OR APPLICABLE SOCIETY COPYRIGHT OWNER]." Also Lancet special credit - "Reprinted from The Lancet, Vol. number, Author(s), Title of article, Pages No., Copyright (Year), with permission from Elsevier."

4 Reproduction of this material is confined to the purpose and/or media for which permission is hereby given.

5 Altering/Modifying Material: Not Permitted. However figures and illustrations may be altered/adapted minimally to serve your work. Any other abbreviations, additions, deletions and/or any other alterations shall be made only with prior written authorization of Elsevier Ltd. (Please contact Elsevier at permissions@elsevier.com)

6 If the permission fee for the requested use of our material is waived in this instance, please be advised that your future requests for Elsevier materials may attract a fee.

7 Reservation of Rights: Publisher reserves all rights not specifically granted in the combination of (i) the license details provided by you and accepted in the course of this licensing transaction, (ii) these terms and conditions and (iii) CCC's Billing and Payment terms and conditions.

8 License Contingent Upon Payment: While you may exercise the rights licensed immediately upon issuance of the license at the end of the licensing process for the transaction, provided that you have disclosed complete and accurate details of your proposed use, no license is finally effective unless and until full payment is received from you (either by publisher or by CCC) as provided in CCC's Billing and Payment terms and conditions. If full payment is not received on a timely basis, then any license preliminarily granted shall be deemed automatically revoked and shall be void as if never granted. Further, in the event that you breach any of these terms and conditions or any of CCC's Billing and Payment

terms and conditions, the license is automatically revoked and shall be void as if never granted. Use of materials as described in a revoked license, as well as any use of the materials beyond the scope of an unrevoked license, may constitute copyright infringement and publisher reserves the right to take any and all action to protect its copyright in the materials.

9 Warranties: Publisher makes no representations or warranties with respect to the licensed material.

10 Indemnity: You hereby indemnify and agree to hold harmless publisher and CCC, and their respective officers, directors, employees and agents, from and against any and all claims arising out of your use of the licensed material other than as specifically authorized pursuant to this license.

11 No Transfer of License: This license is personal to you and may not be sublicensed, assigned, or transferred by you to any other person without publisher's written permission.

12 No Amendment Except in Writing: This license may not be amended except in a writing signed by both parties (or, in the case of publisher, by CCC on publisher's behalf).

13 Objection to Contrary Terms: Publisher hereby objects to any terms contained in any purchase order, acknowledgment, check endorsement or other writing prepared by you, which terms are inconsistent with these terms and conditions or CCC's Billing and Payment terms and conditions. These terms and conditions, together with CCC's Billing and Payment terms and conditions (which are incorporated herein), comprise the entire agreement between you and publisher (and CCC) concerning this licensing transaction. In the event of any conflict between your obligations established by these terms and conditions and those established by CCC's Billing and Payment terms and conditions, these terms and conditions shall control.

14 Revocation: Elsevier or Copyright Clearance Center may deny the permissions described in this License at their sole discretion, for any reason or no reason, with a full refund payable to you. Notice of such denial will be made using the contact information provided by you. Failure to receive such notice will not alter or invalidate the denial. In no event will Elsevier or Copyright Clearance Center be responsible or liable for any costs, expenses or damage incurred by you as a result of a denial of your permission request, other than a refund of the amount(s) paid by you to Elsevier and/or Copyright Clearance Center for denied permissions.

C.3.3 Limited license

The following terms and conditions apply only to specific license types:

15 **Translation**: This permission is granted for non-exclusive world **English** rights only unless your license was granted for translation rights. If you licensed translation rights you may only translate this content into the languages you requested. A professional translator must perform all translations and reproduce the content word for word preserving the integrity of the article. If this license is to re-use 1 or 2 figures then permission is granted for non-exclusive world rights in all languages.

16 **Website**: The following terms and conditions apply to electronic reserve and author websites: **Electronic reserve**: If licensed material is to be posted to website, the web site is to be password-protected and made available only to bona fide students registered on a relevant course if: This license was made in connection with a course, This permission is granted for 1 year only. You may obtain a license for future website posting, All content posted to the web site must maintain the copyright information line on the bottom of each image, A hyper-text must be included to the Homepage of the journal from which you are licensing at `http://www.sciencedirect.com/science/journal/xxxxx` or the Elsevier homepage for books at `http://www.elsevier.com` , and Central Storage: This license does not include permission for a scanned version of the material to be stored in a central repository such as that provided by Heron/XanEdu.

17 **Author website** for journals with the following additional clauses:

All content posted to the web site must maintain the copyright information line on the bottom of each image, and he permission granted is limited to the personal version of your paper. You are not allowed to download and post the published electronic version of your article (whether PDF or HTML, proof or final version), nor may you scan the printed edition to create an electronic version, A hyper-text must be included to the Homepage of the journal from which you are licensing at `http://www.sciencedirect.com/science/journal/xxxxx` , As part of our normal production process, you will receive an e-mail notice when your article appears on Elseviers online service ScienceDirect (`www.sciencedirect.com`). That e-mail will include the articles Digital Object Identifier (DOI). This number provides the electronic link to the published article and should be included in the posting of your personal version. We ask that you wait until you receive this e-mail and have the DOI to do any posting. Central Storage: This license does not include permission for a scanned version

of the material to be stored in a central repository such as that provided by Heron/XanEdu.

18 **Author website** for books with the following additional clauses: Authors are permitted to place a brief summary of their work online only. A hyper-text must be included to the Elsevier homepage at `http://www.elsevier.com`

All content posted to the web site must maintain the copyright information line on the bottom of each image You are not allowed to download and post the published electronic version of your chapter, nor may you scan the printed edition to create an electronic version. Central Storage: This license does not include permission for a scanned version of the material to be stored in a central repository such as that provided by Heron/XanEdu.

19 **Website** (regular and for author): A hyper-text must be included to the Homepage of the journal from which you are licensing at `http://www.sciencedirect.com/science/journal/xxxxx`. or for books to the Elsevier homepage at `http://www.elsevier.com`

20 **Thesis/Dissertation**: If your license is for use in a thesis/dissertation your thesis may be submitted to your institution in either print or electronic form. Should your thesis be published commercially, please reapply for permission. These requirements include permission for the Library and Archives of Canada to supply single copies, on demand, of the complete thesis and include permission for UMI to supply single copies, on demand, of the complete thesis. Should your thesis be published commercially, please reapply for permission.

21 Other Conditions:

v1.6

Gratis licenses (referencing \$0 in the Total field) are free. Please retain this printable license for your reference. No payment is required.

If you would like to pay for this license now, please remit this license along with your payment made payable to "COPYRIGHT CLEARANCE CENTER" otherwise you will be invoiced within 48 hours of the license date. Payment should be in the form of a check or money order referencing your account number and this invoice number RLNK10956481. Once you receive your invoice for this order, you may pay your invoice by credit card. Please follow instructions provided at that time.

Make Payment To: Copyright Clearance Center Dept 001 P.O. Box 843006 Boston, MA 02284-3006

For suggestions or comments regarding this order, contact Rightslink Customer Support: customercare@copyright.com or +1-877-622-5543 (toll free in the US) or +1-978-646-2777.