Next Generation Computational Tools for Extreme-Scale Simulation of Dynamic Fracture and Fragmentation in Three Dimensions

by

Andrew Nathan Seagraves

B.M.E., University of Delaware (2007)
S.M. Massachusetts Institute of Technology (2010)

Submitted to the Department of Mechanical Engineering in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Mechanical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2013

© Massachusetts Institute of Technology 2013. All rights reserved.
Next Generation Computational Tools for Extreme-Scale Simulation of Dynamic Fracture and Fragmentation in Three Dimensions

by

Andrew Nathan Seagraves

Submitted to the Department of Mechanical Engineering on May 2, 2013, in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Mechanical Engineering

Abstract

The accurate modeling of dynamic fracture and fragmentation remains one of the most difficult challenges in computational mechanics research. As part of this thesis, a scalable algorithm for modeling dynamic fracture and fragmentation of solids in three dimensions is developed. The method is based on the combination of a discontinuous Galerkin (DG) formulation of the continuum problem and Cohesive Zone Models (CZM) of fracture. Prior to fracture, the flux and stabilization terms arising from the DG formulation at interelement boundaries are enforced via interface elements, much like in conventional approaches based on CZM. Upon the onset of fracture, the traction-separation law (TSL) governing the fracture process becomes operative without the need to propagate topological changes in the mesh as cracks and fragments develop. This enables the indistinctive treatment of crack propagation across processor boundaries and, thus, scalability in parallel computations. Upon crack closure, the reinstatement of the DG terms guarantees the proper description of compressive waves across closed crack surfaces. Another advantage of the method is that it preserves consistency and stability in the uncracked interfaces, thus avoiding issues with wave propagation typical of many CZM-based approaches.

The new method is applied in this thesis to conduct the first systematic large-scale 3D simulation study of projectile impact damage in brittle plates using CZMs. Utilizing full machine access to supercomputers from the DoD Supercomputing Resource Center (DSRC), we conduct a series of impact simulations at an unprecedented scale aimed at investigating the fundamental physics governing the fracture and fragmentation of brittle plates subjected to normal impact loads. Using the full-field description provided by the simulations, we identify the driving forces for different fracture mechanisms as a function of key problem parameters such as impact speed and plate geometry. In order to validate the computational approach, we also simulate several edge-on impact experiments and compare the simulated cracking patterns directly to experimental results. Another important contribution of this thesis is the
first large-scale study of convergence of the fracture patterns predicted in 3D simulations, for which we complete full runs on up to 17,264 processors using meshes comprising up to 2.4 billion degrees of freedom.

In addition to the simulation studies, we also develop a fracture mechanics-based model for characterizing radial cracking patterns in thin plates subjected to contact loads. To this end, we derive an approximate closed-form expression for the number of radial cracks which will propagate in an elastic membrane subjected to time-invariant axisymmetric transverse loads. The model elucidates some of the basic physics governing the radial cracking process and proves useful as a tool for predicting the number of radial cracks in three-dimensional problems involving contact loads.

Thesis Supervisor: Raúl A. Radovitzky
Title: Professor of Aeronautics and Astronautics
## Contents

1 Introduction  
1.1 Plate Impact Experiments  ........................................ 20  
1.2 Computational Modeling of Impact Damage  ................. 24  
1.3 Summary of Thesis Contributions  .............................. 30

2 Review of Cohesive Zone Approaches for Modeling Brittle Fracture  33  
2.1 Continuous Galerkin Finite Element Implementation using Interface Elements  ......................... 37  
2.2 Intrinsic Approach  .............................................. 42  
2.3 Extrinsic Approach  ............................................. 51

3 DG/CZM: A Scalable 3D Approach for Extreme-Scale Dynamic Fracture Simulation  63  
3.1 DG/CZM Weak Formulation  ...................................... 64  
3.2 Cohesive Law and Post Fracture Contact  ....................... 70  
3.3 Finite Element Discretization and Explicit Time Integration  ................. 71  
3.4 Parallel Implementation using MPI  .................................. 73  
3.5 Large-Scale Scalability Testing and Performance Analysis  ............. 78

4 Validation Studies  ............................................. 89  
4.1 Edge-On Impact Simulations  .................................... 89  
4.1.1 Edge-On Impact of PMMA  ........................................ 90  
4.1.2 Edge-On Impact of Ceramics  ................................ 97
List of Figures

1-1 Post-mortem cross sectional views of crack patterns in ceramics tiles impacted by spheres. a) Al₂O₃ tile, 304 Stainless steel sphere, 586 m/s impact velocity [22], and b) SiC tile, WC sphere, 322 m/s impact velocity [20].

1-2 Ring cracks on the surface of an SiC specimen impacted by a steel sphere at 270 m/s [12].

1-3 Radial cracking patterns on the back face of an Al₂O₃ tile impacted by a steel sphere at 586 m/s from [24].

2-1 The cohesive zone interpretation of brittle fracture.

2-2 A single cohesive surface S₀ traversing a three-dimensional body B₀.

2-3 Description of a 12-node interface element introduced between two 10-node quadratic tetrahedra Ω⁺₀ and Ω⁻₀.

2-4 Schematic of the traction-separation laws utilized in a) the intrinsic approach and b) the extrinsic approach.

2-5 Mesh dependence of branching pattern from [45] for triangular elements oriented at (a) 15°, (b) 30°, (c) 45°, (d) 60°.

2-6 Mesh dependence of fracture pattern and of the force-thickness reduction curve from [55] for different aspect ratios of the crossed-triangle mesh structure - aspect 1 = 0.125x0.046 mm², aspect 2 = 0.125x0.054 mm². Note T₀,N = σc for the notation used in this work.

2-7 Plot of crack tip speed, ȧ vs time, t showing lift-off from [45]. Dashed line is the Rayleigh wave speed.
2-8 Wave propagation experiments from [91] for increasing values of the cohesive law slope $s$ ................................................................. 50
2-9 $T - \delta$ relationship for the linear decreasing extrinsic law ........... 54
2-10 Mesh dependence of the fracture pattern for symmetrically loaded pre-cracked PMMA strip for (a) 32 x 128 elements at $t = 24 \mu s$, (b) 48 x 192 elements at $t = 22 \mu s$ and (c) 48 x 192 elements at $t = 21 \mu s$ from [57] ................................................................. 55
2-11 Comparison of crack path, load history, and dissipated cohesive energy for two different mesh sizes from [49] ................................. 56
2-12 Energy convergence with respect to mesh size for crack branching in PMMA from [59] ................................................................. 58
2-13 Energy convergence for 1D ring fragmentation for uniform and random meshes from [59] ............................................................... 59
2-14 Dependence of size distribution of fragments on mesh size from [59] . 59

3-1 (a) A schematic of the element subdivision scheme utilized for mesh refinement. The original tetrahedron $(t_0,t_1,t_2,t_3)$ is subdivided into 8 subtetrahedron: $(t_0,t_{01},t_{02},t_{03})$, $(t_{01},t_{1},t_{12},t_{13})$, $(t_{03},t_{13},t_{23})$, $(t_{2},t_{02},t_{12},t_{23})$, $(t_{01},t_{02},t_{03},t_{13})$, $(t_{01},t_{02},t_{12},t_{13})$, $(t_{02},t_{03},t_{13},t_{23})$, and $(t_{02},t_{12},t_{13},t_{23})$ . 77
3-2 Initial demonstration of weak scalability of the DG/CZM method on up to 4096 processors and problems of upward of 3 billion degrees of freedom: Scaled speed-up results given as compute time per time step as a function of number of processors for all three platforms tested . 81
3-3 Recent demonstration of weak scalability of the DG/CZM method on new ARL systems Hercules and Pershing. The plot shows simulation time per time step as a function of the number of cores used. 82
3-4 Recent demonstration of strong scalability on up to 17,264 processors for two different problem sizes on new ARL system Hercules. The plot shows simulation time per time step as a function of the number of cores used. 83
3-5 Analysis of the relative costs of communication versus computation for impact damage simulations using different numbers of processors for three mesh sizes. ................................................................. 84

3-6 Measured FLOPS performance for each processor in a full machine run on new ARL system Hercules. The plot shows the measured minimum, maximum, and average FLOPS for a run of 100 timesteps. .......... 86

4-1 A schematic of the specimen and impactor geometry from the Mode I EOI tests of Umberger and Love on pre-notched PMMA plates [111]. 91

4-2 Snapshots showing crack propagation and the evolution of $\sigma_{yy}$ for simulations of the EOI tests on pre-notched PMMA plates from [111]. 93

4-3 A comparison of the steady state crack tip speeds extracted from simulations with the experimentally observed values from the EOI experiments of Umberger and Love on pre-notched PMMA plates [111]. Results are shown for simulations using both rate-independent and rate-dependent CZM. ..................................................... 94

4-4 The initial configuration for simulations of the edge-on impact tests of Strassburger and Senf [115]. The figure shows the finite element discretization used in the simulations, consisting of 266,000 volumetric elements. ................................................................. 98

4-5 Comparison of simulated (left) and experimental (right) crack propagation patterns for 85 m·s$^{-1}$ impact ................................................. 100

4-6 Comparison of simulated and experimental crack propagation lengths over time for the cone crack and the damage front for an impact speed of 85 m·s$^{-1}$ ......................................................... 101

4-7 Comparison of simulated (left) and experimental (right) crack propagation patterns for 150 m·s$^{-1}$ impact ................................. 102

4-8 Comparison of the average radial location of the damage front over time from simulations and experiments for an impact velocity of 150 m·s$^{-1}$ ................................................................. 103
4-9 Evolution of the simulated maximum principal stress and associated principal direction. For each time snapshot, the results for the lower impact velocity (85 m·s⁻¹), shown on the left of the figure, are compared to those for the higher impact velocity (150 m·s⁻¹), shown on the right of the figure.

4-10 Snapshot of simulation results at \( t = 0.52 \times 10^{-6}\) s

4-11 Snapshot of simulation results at \( t = 2.09 \times 10^{-6}\) s

4-12 Snapshot of simulation results at \( t = 3.4 \times 10^{-6}\) s

4-13 Snapshot of simulation results at \( t = 5.46 \times 10^{-6}\) s

4-14 Snapshot of simulation results at \( t = 10.9 \times 10^{-6}\) s

4-15 Snapshot of simulation results at \( t = 15.6 \times 10^{-6}\) s

4-16 Radial cracks and trace of dominant conical crack on plate back face at the end of the simulation

4-17 Deformed back face view at the end of the simulation showing radial cracks propagating across mesh partitions

5-1 Simulated damage patterns in an \( \text{Al}_2\text{O}_3 \) tile impacted at 100 m·s⁻¹ showing a slice taken near the front surface of the tile (left) and a slice taken near the back surface of the tile (right). The displacements have been magnified by a factor of 10.

5-2 Simulated damage patterns in an \( \text{Al}_2\text{O}_3 \) tile impacted at 400 m·s⁻¹ showing a slice taken near the front surface of the tile (left) and a slice taken near the back surface of the tile (right). The displacements have been magnified by a factor of 10.

5-3 Simulated damage patterns in \( \text{Al}_2\text{O}_3 \) tile impacted at 700 m·s⁻¹ showing a slice taken near the front surface of the tile (left) and a slice taken near the back surface of the tile (right). The displacements have been magnified by a factor of 5.
5-4 Post-mortem fracture patterns in an 8x100x100mm Al₂O₃ tile supported by a 2mm thick steel backing plate and impacted by a steel sphere at 1052 m·s⁻¹ [24] .............................................. 122

5-5 Snapshot of simulation results at t = 4.0µs showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right) .............................................. 126

5-6 Snapshot of simulation results at t = 6.0µs showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right) .............................................. 127

5-7 Snapshot of simulation results at t = 8.0µs showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right) .............................................. 128

5-8 Snapshot of simulation results at t = 10.0µs showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right) .............................................. 129

5-9 Snapshot of simulation results at t = 12.0µs showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right) .............................................. 130

5-10 Snapshot of simulation results at t = 14.0µs showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right) .............................................. 131
5-11 Snapshot of simulation results at \( t = 16.0 \mu s \) showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right) ........................................ 132

5-12 Simulated damage patterns in a 6mm thick Al\(_2\)O\(_3\) tile impacted at 700 m\(\cdot\)s\(^{-1}\) showing a slice taken near the front surface of the tile (left) and a slice taken near the back surface of the tile (right). The displacements have been magnified by a factor of 10. ........................................ 134

5-13 Simulated damage patterns in a 12mm thick Al\(_2\)O\(_3\) tile impacted at 700 m\(\cdot\)s\(^{-1}\) showing a slice taken near the front surface of the tile (left) and a slice taken near the back surface of the tile (right). The displacements have been magnified by a factor of 10. ........................................ 134

5-14 Fracture patterns in the cross section after a simulation time of \( t = 20.0 \mu s \) for tiles of (a) 6 and (b) 12mm thickness impacted at a velocity of 700m/s. ........................................ 135

5-15 3D renderings of the 6mm plate impacted at 700 m/s showing fragments being ejected from the back of the tile: (top) Back face view, (bottom) Side view. The displacements have been magnified by a factor of 2. ........................................ 136

5-16 3D renderings of the 6mm plate impacted at 700 m/s showing fragmentation due to significant extended cracking which has propagated through the thickness: Top face view (top), Back face view (bottom). Ejected fragments were filtered out during the rendering process in order to provide a full view of the extended cracking patterns. The displacements have been magnified by a factor of 15. ........................................ 137

5-17 Snapshot of simulation results at \( t = 4.0 \mu s \) showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right) ........................................ 139
5-18 Snapshot of simulation results at $t = 6.0\mu s$ showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right) .......................................................... 140

5-19 Snapshot of simulation results at $t = 8.0\mu s$ showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right) .......................................................... 141

5-20 Snapshot of simulation results at $t = 10.0\mu s$ showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right) .......................................................... 142

5-21 Snapshot of simulation results at $t = 12.0\mu s$ showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right) .......................................................... 143

5-22 Snapshot of simulation results at $t = 14.0\mu s$ showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right) .......................................................... 144

6-1 The initial coarse mesh used in the mesh dependency calculations. ... 147

6-2 $s = 0$ case: Full 3D rendering of final radial cracking patterns on the back face (top) and the trace of the radial crack surfaces on the back face produced from a rendering of the fractured interface elements at a time of $t = 12\mu s$. Displacements have been magnified by a factor of 150 in the 3D rendering .......................................................... 151
6-3  $s = 1$ case: Full 3D rendering of final radial cracking patterns on the back face (top) and the trace of the radial crack surfaces on the back face produced from a rendering of the fractured interface elements at a time of $t = 12\mu s$. Displacements have been magnified by a factor of 150 in the 3D rendering.

6-4  $s = 2$ case: Full 3D rendering of final radial cracking patterns on the back face (top) and the trace of the radial crack surfaces on the back face produced from a rendering of the fractured interface elements at a time of $t = 12\mu s$. Displacements have been magnified by a factor of 150 in the 3D rendering.

6-5  $s = 3$ case: Full 3D rendering of final radial cracking patterns on the back face (top) and the trace of the radial crack surfaces on the back face produced from a rendering of the fractured interface elements at a time of $t = 12\mu s$. Displacements have been magnified by a factor of 150 in the 3D rendering.

6-6  $s = 4$ case: The trace of radial crack surfaces on the back face produced from a rendering of the fractured interface elements at a time of $t = 12\mu s$. Because the file size was so large, a 3D rendering could not be produced.

6-7 Final radial cracking patterns after complete crack arrest extracted from Figures 6-2–6-6 using Datathief and overlaid on top of each other using a commercial plotting package.

6-8 The radial cracking data in Figure 6-7 is replotted here with the envelope of the average radial crack length overlayed on the plot for each value of $s$. This provides a qualitative visual depiction of convergence of the average radial crack length.
6-9 Average total radial crack length and average Euclidean (end-to-end) radial crack length after complete crack arrest as a function of the number of degrees of freedom. The diminishing slope of the plot suggests that the average radial crack lengths are starting to converge as the calculations reach the billion-size.

6-10 Temporal evolution of the dissipated cohesive energy during the radial cracking process at different levels of mesh refinement.

6-11 Total cohesive energy dissipated by the radial cracking process after complete crack arrest as a function of the number of degrees of freedom. The increased dissipation with mesh refinement indicates that we are far from convergence in this metric.

6-12 Time at complete arrest of radial cracks as a function of the number of degrees of freedom.

7-1 A schematic depiction of the star-shaped and hypocycloidal crack geometries for \( n = 6 \).

7-2 Snapshots of the final deformation and failure pattern as a function of the indentation load for diamond micro-indentation experiments on SiC-SC-1R from [133].

7-3 Comparison of the number of radial cracks predicted by the theory with experimental results [133] as a function of indentation load.

A-1 A schematic illustrating the penalty-based algorithm used to enforce the contact constraint in impact simulations in this thesis.

B-1 A schematic illustrating the search box used to estimate the crack tip location in Mode I edge-on impact simulations of section 4.1.1.

D-1 A portion of the ParaView state file constructed using low resolution data showing the hard-coded file path to the low resolution dataset (top) and the same code snippet with the hard-coded file path replaced by a simple tag (bottom).
List of Tables

4.1 Problem Parameters and Material Properties for simulations of the EOI tests of Umberger and Love [111] .................. 92

4.2 Plate and projectile dimensions and material properties used for simulations of the EOI experiments of Strassburger and Senf [115] .... 99

4.3 Problem parameters and material properties for simulations of rigid sphere impact on alumina ........................................ 107

6.1 Details on the finite element meshes used in the convergence runs at different levels of subdivision. The variables $h_{\text{min}}$ and $h_{\text{max}}$ refer to the minimum and maximum edge size of the volumetric elements, respectively. ........................................ 147

6.2 Details on the computational platforms, the number of processors, the total compute time, and the corresponding CPU-hours used for the convergence runs at different levels of subdivision ................. 147

6.3 Details on the average output file sizes produced per time step at different levels of subdivision ........................................ 148

7.1 Parameter values used in calculations ........................................ 175

C.1 Plate and projectile dimensions and material properties used in normal impact simulations ........................................ 195
Chapter 1

Introduction

The damage and failure of brittle materials subjected to intense loads is characterized by the development of intricate patterns of three-dimensional cracks. Developing accurate models to predict fracture and fragmentation has been one of the most difficult and long-standing challenges faced by the computational mechanics community. In localized impact loading, cracks are driven by complex dynamic stress fields which arise as a result of stress wave propagation. These dynamic stresses can be highly complex and heterogeneous due to the interaction of stress waves with free boundaries, material interfaces, and newly formed crack surfaces [1, 2, 3, 4]. Improving the fundamental understanding of how dynamic stress fields drive three-dimensional brittle damage patterns is critical in many applications including armor materials [5, 6], orbital debris mitigation [7] and hydraulic fracturing [8].

Given the critical importance of this problem, there have been significant attempts to develop algorithms that can predict the complex cracking patterns arising in brittle materials subjected to intense loads [9]. However, owing to fundamental challenges inherent in modeling the nucleation and propagation of multiple three-dimensional cracks under dynamically evolving stress fields, the development of a method which can accurately predict dynamic brittle fracture patterns remains one of the most difficult problems in computational mechanics research. Faced with this challenge, the overall objective of this thesis is to develop advanced theoretical and computational tools which can accurately model dynamic fracture and fragmentation processes in
3D. The main driver problem which has motivated this research and to which we apply these tools is the damage and failure of brittle plates subjected to high velocity projectile impact. In the following section, we begin the discussion of the impact problem by presenting a brief review of impact experiments.

1.1 Plate Impact Experiments

Over the last forty years, there has been a strong effort to study impact damage through plate impact experiments where the cracking patterns were analyzed in recovered specimens [1, 2, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22]. A key finding of these studies is that although highly complex, the cracking patterns resulting from impact loads can sometimes be decomposed into a small set of characteristic fracture modes whose propagation paths follow simple trajectories. For example, some of the characteristic cracking patterns have a rotational symmetry about the impact axis. The most common examples of axisymmetric cracking patterns are "conical" and "ring" cracks.

Conical cracks nucleate near the tile surface directly adjacent to the contact zone and propagate into the volume along conoidal fronts which are characterized by a cone angle (see Figure 1-1). Another axisymmetric cracking mode that is observed frequently in impact experiments are "ring" cracks, which are circular cracking patterns that nucleate some distance outside of the contact zone on the impacted tile surface (see Figure 1-2). Ring cracks are characterized by their average radius which is measured from the central impact point. Both conical and ring cracks are ubiquitous features of brittle impact damage and have been observed in impact experiments on a variety of different brittle materials including glasses [10, 13], nominally brittle polymers [19], and ceramics [1, 2, 11, 12, 14, 15, 16, 17, 18, 20, 23, 22].

In addition to providing the driving force for nucleation of axisymmetric crack fronts, the radially symmetric stress waves generated under the impactor also drive a symmetry-breaking characteristic cracking mode which is inherently three-dimensional. This failure mode, known as "radial" cracking, is characterized by multiple planar
Figure 1-1: Post-mortem cross sectional views of crack patterns in ceramics tiles impacted by spheres. a) Al\textsubscript{2}O\textsubscript{3} tile, 304 Stainless steel sphere, 586 m/s impact velocity [22], and b) SiC tile, WC sphere, 322 m/s impact velocity [20].

Figure 1-2: Ring cracks on the surface of an SiC specimen impacted by a steel sphere at 270 m/s [12]
cracks propagating simultaneously in radial directions from the impact region. Similarly to the axisymmetric fracture modes, radial cracks have been observed in impact experiments on a variety of brittle materials [2, 11, 13, 14, 15, 16, 17, 5, 19, 23]. Figure 1-3 shows a typical post-mortem radial cracking pattern on the back face of a damaged Al₂O₃ tile after impact by a steel sphere at 586 m/s from [24].

The nucleation and propagation of radial cracking patterns is generally attributed to hoop stress fields which arise at some depth directly below the impact point [2, 16, 17]. Correspondingly, radial cracks have been observed to nucleate well below the impacted surface either in the specimen volume [2], or on the opposing back surface of the plate. The plate back surface can be either a free boundary for unsupported plates [13, 15, 19], or a material interface for a plate with a backing support [14, 16, 17, 5, 23]. Radial cracks greatly complicate the analysis of damage in impact scenarios because of their tendency to nucleate in subsurface regions and because they require a fully three-dimensional analysis for a problem that could otherwise be studied in two dimensional axisymmetric mode.

Rotational symmetry of the damage response can also be violated when stress waves reflect from the lateral boundaries of the specimen. Boundary reflections can cause non-axisymmetric cracking to occur from localized tensile stresses in the reflected waves, or by the interaction of reflected waves with propagating cracks. These symmetry-breaking boundary effects are strongest when the lateral boundaries of the specimen are free surfaces. Stress wave reflections at free lateral boundaries not only promote an inherently three-dimensional damage response, but can also cause complete fragmentation of an impacted sample, as observed by Sherman in several experimental studies of high speed projectile impact on Al₂O₃ tiles [14, 15, 16, 17].

The large body of experimental evidence collected in plate impact studies shows that the progressive development of damage in an impacted plate will generally entail the nucleation and propagation of some combination of axisymmetric and non-axisymmetric cracking modes. The type of failure modes that are activated, their sequence of activation and subsequent interaction, and the final overall fracture pattern depend strongly on target and projectile material properties, target geometry and
boundary constraints, and projectile speed [1, 2, 11, 13, 14, 16, 17, 18]. This parametric dependence has been studied experimentally to some extent. For example, Evans and Wilshaw [2] conducted impact experiments of WC spheres on semi-infinite ZnS plates at different impact velocities. In their tests, the authors observed the trace of radial cracks on the impacted surface along with fragmentation and material removal from the plate surface. Their rather intuitive results indicated that both the average final radial crack lengths and the extent of material removal from the impact surface increased with impact velocity.

Similar results were reported by Normandia [18] in experiments of WC sphere impact on thin SiC-N plates encased in a steel confinement chamber. For impact velocities up to 393 m·s⁻¹, only radial cracks were nucleated. Above this threshold velocity, a fragmented crater developed under the penetrator accompanied by radial cracks, ring cracks and material removal from a shallow fragmented circumferential zone at the impacted surface. Normandia measured the radial extent of the frag-
mented region on the top surface and found that it increased linearly with impact velocity. Ball and McKenzie [13] investigated the effects of impact velocity and specimen thickness on the fracture patterns in glass plates impacted by steel spheres. The authors observed radial, ring, and conical cracks and crushing of material under the penetrator and found that the particular combination of fracture modes for a given test depended sensitively on both the specimen thickness and the impact velocity.

In a series of papers [14, 15, 16, 17], Sherman and co-workers conducted a comprehensive parametric study of high velocity impact on Al₂O₃ tiles. An important finding from these tests is that the predicted fracture response depends strongly on the tile thickness and on the confinement conditions at the lateral boundaries of the specimen. In [16, 17], Sherman used steel and aluminum supporting plates which showed traces of plastic deformation in coincidence with the location of radial cracks. This provided for the first time, a quantitative correlation between test parameters and the number of radial cracks and showed that the number of cracks depends strongly on the tile thickness. Sherman also studied the effects of lateral confinement in [15, 16, 17] and found that tiles with free lateral boundaries sustained a greater amount of damage than confined tiles under equivalent impact conditions. Indeed, while many of the confined tiles were recovered intact, the tiles with free lateral boundaries were completely fragmented in all cases. Sherman attributed the severity of damage in unconfined tiles to the activation of additional failure modes from stress waves reflected at the lateral specimen boundaries.

1.2 Computational Modeling of Impact Damage

While these experimental studies have provided some key insights into the complex dependence of cracking patterns on the problem parameters in projectile impact scenarios, some fundamental questions still remain unanswered. For example, what aspects of the stress field control the activation and evolution of different fracture modes and how are the stresses driving damage affected by changes in problem parameters such as the impact speed and the tile thickness as well as by the propagating
cracks? Also, what is the influence of non-axisymmetric boundary effects as a function of problem parameters and under what conditions are boundary effects important?

Over the last thirty years, there has been a strong effort in the solid mechanics community to develop models which describe the damage process in brittle materials subjected to impact loads. A number of approaches have been proposed including both analytical models [25, 20, 26] and computational methods [27, 28, 29, 30, 31, 22, 32, 3, 33, 34]. Some of the analytical efforts include the work of Zaera and Sánchez-Gálvez [25] who predicted the ballistic limit and residual projectile velocity in long-rod penetration of ceramic/steel sandwich structures, the work of Satapathy [26] who predicted the depth of penetration in semi-infinite ceramics using spherical cavity expansion models, and the work of Iyer [20] who predicted final cone crack lengths in sphere impact tests on SiC. Due to the complexity of impact damage processes in brittle materials, these analytical models make very strong simplifying assumptions in order to retain analytical tractability and very often important details of the damage physics are built into the model a priori, thereby reducing its overall applicability to a range of different problems. Another limitation of analytical approaches is that they typically only attempt to model axisymmetric modes of damage and as a result, important 3D effects such as radial cracks, are ignored.

Computational approaches, on the other hand, have the potential to overcome many of the limitations of analytical models since generally they require less restrictive assumptions and can be applied to a wider range of problems. Of the various computational approaches that have been developed to model brittle impact damage, most work has focused on modeling comminution and erosion failure mechanisms through the so-called ”smeared crack” approach. In this class of methods, the loss of load carrying capacity is effected through a ”continuum damage” constitutive law. Continuum damage models are usually either completely empirical (see e.g. the early ceramic failure model of Wilkins [27, 28], the popular approach of Johnson and Holmquist [29, 35], or the recent work by Brannon and co-workers [33, 34]) or they are based on mechanistic models of microcracking (see e.g. the recent work of Deshpande and Evans [30] and co-workers [31, 22, 36]). 25
In either case, the formation of new crack surfaces is not modeled explicitly in the finite element mesh. Rather, an additional field variable is introduced into the formulation, usually called "damage", which is governed by some evolution equation and attempts to quantify the extent of progressive damage which has occurred. As the damage variable increases from zero to one, the strength of the material is gradually reduced to zero through an augmented stress-strain relation which depends on the current value of the damage. Continuum damage models offer the advantage of being relatively straightforward to implement in conventional finite element codes as they reduce the problem of modeling damage to that of constitutive modeling. However, strain-softening damage models suffer from a number of limitations including the well-known result that the energy dissipated in the damage process tends to zero as the mesh size is reduced due to unphysical strain localization [37]. In order to avoid this pernicious behavior, continuum damage models need to be enhanced through the formulation of nonlocal [38] and gradient [39] damage models which incorporate an internal length scale that regularizes the strain localization. Although they mitigate mesh dependency issue to some extent, these models have not permeated production computational frameworks, primarily for two reasons: (1) new (high-order) algorithms and computer codes are required because the existing algorithmic frameworks cannot accommodate the higher-order derivatives and their field continuity requirements and (2) additional constitutive parameters are required that in many cases do not have a clear physical meaning or a discerning experiment that can be used to calibrate them [9]. It is also not clear whether these approaches can provide an accurate description of fundamental aspects of dynamic fracture physics including dynamic crack tip stress intensity factors, crack tip speeds, crack branching, and fragmentation.

An alternative, also very popular, class of computational methods which have shown promise for modeling dynamic brittle fracture is based on the so-called "discrete crack" model. In the discrete crack approach, the initiation and propagation of cracks is modeled explicitly by introducing surfaces of discontinuity within the material. The fracture processes at these surfaces of discontinuity can be described by cohesive zone models (CZM) of fracture [40, 41] via a phenomenological traction-
separation law (TSL). The most popular implementation of this concept is the so-called “cohesive element” method (see [42] for a recent review) in which crack openings are represented as displacement jumps at the inter-element boundaries using “interface” or “cohesive” finite elements [43, 44, 45, 3, 46, 47, 48, 49, 50, 51, 52]. Despite the potential of cohesive element approaches for treating a broad class of fracture mechanics problems in three-dimensions, there exist a number of numerical issues and physical limitations of the models that still must be resolved. As is well known, simulations using cohesive element methods suffer from a well-known mesh dependency as the possible crack nucleation sites and propagation paths are constrained by the finite element discretization [45, 46, 53, 54, 47, 49, 55, 50, 56, 57, 58, 59, 60, 4]. This hinders the ability to describe complex crack patterns arising in three dimensional problems and leads to a very slow rate of convergence in the dissipated fracture energy as the mesh is refined [59]. Other important issues include problems with the propagation of stress waves and associated stability of time integration algorithms and problems with the parallel implementation of topological mesh changes emerging from the propagation of cracks.

A possible avenue for mitigating mesh dependency in cohesive element methods is to employ very fine meshes. As the mesh becomes finer, the available sites for crack nucleation and propagation increases, thus reducing crack path dependence on the mesh. Mesh refinement is also critical for resolving the size of the fracture process zone which is exceedingly small in brittle materials. The need for highly refined meshes naturally demands parallel computational schemes that are scalable to large numbers of processors for problems of increasing size, especially in three dimensional problems. However, owing to fundamental issues in the formulation and implementation of conventional cohesive element methods, scalable three-dimensional algorithms for accurately predicting fracture and fragmentation do not currently exist.

Faced with this long-standing challenge, in this manuscript we develop the first scalable cohesive element approach for modeling dynamic brittle fracture in three-dimensions, an idea that originated in my Master’s research [42, 61]. The new method is based on a discontinuous Galerkin (DG) reformulation of the continuum prob-
lem that exploits the virtues of existing cohesive element methods. Discontinuous Galerkin methods are a generalization of weak formulations, allowing for discontinuities of the problem unknowns in the interior of the problem domain. Compatibility, consistency and stability are ensured by recourse to boundary integral terms on the subdomain interfaces involving jump discontinuities [62, 63, 64, 65, 66, 67, 68, 69]. One of the advantages of the DG approach for fracture mechanics based on CZMs, is that it naturally leads to a consistent consideration of the elasticity of the cohesive elements prior to fracture, thus avoiding the common issue of properly describing stress-wave propagation in the uncracked body (discussed in detail in the review of conventional CZM in Chapter 2). Another important aspect of the DG method is that cracks can nucleate and propagate in the finite element mesh without the need for modifying the mesh topology. In a parallel computational setting, this allows for the indistinctive treatment of crack propagation across processor boundaries, thus enabling the possibility of scalable fragmentation simulations in 3D.

Building on the initial ideas presented in my Master's research, in this thesis we improve and further develop the DG/CZM computational framework in our MIT research code "Summit", with the goal of developing an implementation which is highly efficient and scalable. In an extensive set of scalability tests conducted on latest generation DoD supercomputers, we prove both the strong and weak scalability of our implementation in the Summit code for runs up to full-machine size and for exceedingly large problem sizes. Then, we apply the new method to conduct large-scale simulation studies of projectile impact damage in brittle plates with three important goals. The first main goal of this simulation effort is to qualitatively and quantitatively validate the numerical approach through direction simulation of impact experiments and comparison against experimental results. Next, we focus on using simulations to improve the fundamental understanding of the physics governing the fracture and fragmentation of brittle plates subjected to localized impact with particular emphasis on studying the driving forces for different fracture mechanisms as a function of key problem parameters. The third goal of the simulation studies is to qualitatively and quantitatively assess the convergence characteristics of the DG/CZM method up to
very large mesh sizes and for an impact problem involving three-dimensional cracking patterns.

The organization of this manuscript is as follows. In chapter 2 we describe the theoretical origins of the CZM concept and its conventional implementation in FE codes formulated within the continuous Galerkin framework. We also provide a concise review of the formulation, applications, and issues associated with the various types of cohesive laws that have been proposed in the literature. As a means for addressing the issues with conventional CZM discussed in chapter 2, in chapter 3 we introduce the DG framework and its formulation and parallel implementation in the context of dynamic fracture. Also, we present the results of scalability and node performance testing on DoD supercomputers, demonstrating that the numerical framework is highly efficient and scalable for runs up to 17,264 processors using mesh sizes up to of 19.3 billion degrees of freedom.

In chapters 5 and 6, we conduct the first systematic large-scale simulation study of normal impact on brittle plates in three dimensions. Chapter 5 focuses on understanding the parametric dependence of the cracking patterns on impact velocity and tile thickness. Through a detailed analysis of the evolution of the maximum principal stress and its direction, we gain significant insight into the physics governing the damage response of brittle plates as a function of these key problem parameters. Following this exploration of the impact damage physics, in chapter 6 we present a systematic investigation of the convergence of simulated radial cracking patterns as a function of mesh size. In this study, we push the boundaries of large-scale 3D simulation and present the results of impact damage calculations on a sequence of refined meshes up to 2.4 billion degrees of freedom. In support of this work, a new parallel batch rendering process is developed to visualize the exceedingly large output files which are generated, enabling convergence to be studied using several quantitative metrics including the average radial crack length, the dissipated cohesive energy, and the crack arrest time. This study provides a critical glimpse into the future of large scale fracture simulation and reveals the extreme levels of mesh refinement that will be necessary to achieve convergence.
In parallel with the simulation studies carried out for this thesis, a significant
effort was undertaken to develop an analytical theory capable of describing radial
cracking in thin plates under contact loads. As a step towards this goal, in chapter
7 we present the derivation of a fracture-mechanics based model which provides an
approximate closed-form expression for the number of radial cracks which will propa-
gate in an elastic membrane subjected to time-invariant axisymmetric transverse loads.
The model elucidates some of the key physics governing the radial cracking process,
providing the basic functional dependence of the number of cracks in terms of the
applied load, fracture toughness, and initial flaw size. Another key feature of the
model is that it can be extended to three-dimensional problems in a straightforward
manner, enabling predictions to be made for the number of radial cracks in spherical
indentation tests on silicon carbide.

Finally, in chapter 8 we conclude the presentation by discussing the main contribu-
tions of this thesis and outlining several directions for future research.

1.3 Summary of Thesis Contributions

- Formulation of a combined DG/CZM method for the analysis of complex three-
dimensional dynamic fracture patterns in brittle materials

- Scalable implementation of the DG/CZM method in the MIT Summit research
code for large-scale fracture simulation.

- Demonstration of the scalability of the computational framework on DoD su-
percomputers up to 17,264 processors (full machine size) and mesh sizes up to
19.3 billion degrees of freedom.

- Effort to validate simulations against PMMA and ceramic edge-on impact ex-
periments assessing the accuracy of predicted crack propagation lengths and
-crack tip speeds.

- The first systematic large-scale simulation study investigating the physics of
normal impact damage in ceramic plates in three dimensions.
• Quantitative assessment of the mesh dependency and convergence characteristics of predicted fracture patterns in 3D for full-machine simulations of normal impact damage in ceramic plates with mesh sizes up to 2.4 billion degrees of freedom.

• Development of an analytical theory for characterizing radial cracking patterns in thin plates subjected to contact loads.

• Elucidation of the fundamental mechanisms leading to the formation of radial, conical, ring and edge cracks in ceramic plate impact.
Chapter 2

Review of Cohesive Zone Approaches for Modeling Brittle Fracture

Computational approaches to dynamic fracture based on cohesive zone models are rooted in theories initially put forward by Dugdale [41] and Barenblatt [40] which established a framework for modeling nonlinear separation processes at the tips of sharp geometric discontinuities within materials that otherwise behave as linearly elastic. Theories of fracture mechanics based on linear elasticity are insufficient for directly modeling separation processes at the crack tip since they only determine the near-tip fields in regions outside of the fracture process zone. Cohesive theories, on the other hand, attempt to directly model crack face separation as a displacement jump $\Delta$ across an initially coincident surface extending from the crack tip (called a “cohesive surface”). The displacement jump is defined by

$$\Delta = \varphi^+ - \varphi^- = [\varphi]. \quad (2.1)$$

where $\varphi^+$ and $\varphi^-$ are the displacement vectors for two initially coincident points on the cohesive surface. The separation process is resisted by macroscopic forces $T$ acting on the cohesive surface (called the “cohesive tractions”), which are expected to
decay to zero when new crack surfaces are formed at some critical amount of opening ahead of the crack tip, as required by the free surface condition of new crack flanks.

In [40], Barenblatt connected the cohesive zone idealization of the separation process at the crack tip to phenomenological damage mechanisms in elastic brittle fracture involving the separation and cleavage of atomic planes in the process zone. In this work, he argues that the separation process at the crack tip involves displacement jumps of the order of the molecular spacing (occurring as a result of macroscopic separation of atomic planes) which are beyond the resolution of LEFM (see Figure 2-1). Phenomenologically, the cohesive tractions are expected to depend locally on the amount of atomic separation, but not on the undamaged bulk material outside of the cohesive zone. Hence, Barenblatt assumed that the constitutive response of a cohesive surface in a brittle elastic material can be specified through a traction-separation law (TSL) of the following form

\[ T = T(\Delta). \]  

Barenblatt argued that new crack surfaces are formed when the separation in the atomic lattice is much larger than the molecular spacing, at which point the cohesive tractions should decay to zero. Based on the assumptions that the cohesive zone is small compared to the size of the whole crack, and that the cohesive tractions conform to equation (2.2) irrespectively of its specific functional form, Barenblatt demonstrated that the size of the cohesive zone can be chosen so that the stress predicted at the free edge of the process zone is zero. This important result eliminated the singularity of the stress field at the crack tip predicted by LEFM.
A rigorous proof of the independence of the TSL from the bulk material behavior in general elastic materials was provided later by Rice [70, 71] using an analysis based on the $J$-integral. For an initially coincident cohesive surface of length $R$ ahead of a two-dimensional crack growing in the $e_1$ direction, we have

$$J = \int_0^R T \cdot \Delta_1 dX_1 = \int_0^\infty T(\Delta) \cdot d\Delta$$

and hence the form of equation (2.2) holds for linear and nonlinear elastic materials.

Due to the first and second laws of thermodynamics [72, 73], the cohesive law for brittle elastic materials can be shown to have a potential structure. This implies that the cohesive tractions can be obtained from a free energy density function $\phi$ (called the “cohesive free energy density”) by differentiation

$$T = \frac{\partial \phi}{\partial \Delta}$$

Phenomenologically, we expect that the cohesive tractions will vanish at some finite critical value of separation $\Delta_c$. Introducing this assumption into equation (2.3), and recalling that the $J$-integral is equal to the Griffith critical energy-release rate $G_c$ for elastic materials, we obtain

$$J = G_c = \int_0^{\Delta_c} T(\Delta) \cdot d\Delta \equiv \phi_{sep}$$

where $\phi_{sep} = \phi(\Delta = \Delta_c)$ is called the work of separation. Hence, the $J$-integral analysis also shows that at the critical value of separation, the work done per unit crack area by the cohesive tractions on the displacement jumps is equal to the critical energy-release rate, $G_c$, in the Griffith sense [74].

In the formulation of specific cohesive models, a particular form is chosen for the cohesive energy density function which commonly depends on the choice of a critical cohesive stress $\sigma_c$, and a critical opening displacement $\Delta_c$. Evidently then, the relation $\phi_{sep} = G_c$ establishes a fundamental link between the cohesive law and physically-based fracture process parameters, and in turn enables the calibration of
these two model parameters to experiments.

The relationship between the cohesive law and the critical energy release rate also has important consequences for finite element simulations as it introduces a characteristic length \( l_c \) into the calculation given by

\[
l_c = \frac{EG_c}{f_{ts}^2}
\]  

where \( E \) is the elastic modulus and \( f_{ts} \) is the static tensile strength [71]. According to Ruiz, et al., this characteristic length discriminates between specimen of different sizes in finite element simulations [75].

In dynamic calculations, the choice of a TSL with a finite critical opening parameter parameter \( \Delta_c \) introduces a characteristic time \( t_c \) which was first derived by Camacho and Ortiz in [3] and can be written as

\[
t_c = \frac{\rho c_d \Delta_c}{2f_{ts}}
\]  

where \( c_d \) is the dilatational wave speed, and \( \rho \) is the density. It has been argued that this characteristic time accounts for loading-rate effects, as suggested by the correct prediction of strain-rate effects in fragmentation simulations using rate-independent cohesive laws [76, 77, 75].

Another important length scale associated with cohesive theories is the cohesive zone length, \( R \), used in application of the \( J \)-integral, and defined for a Mode I Dugdale-Barenblatt crack under quasistatic loading as [71]

\[
R = \frac{\pi}{8(1-\nu^2)} \frac{E}{\sigma_c^2} \frac{4G_c}{(4-\nu)}
\]  

where \( \nu \) is the Poisson’s ratio. Evidently, the cohesive zone length has important implications on the choice and resolution of interpolation scheme and mesh size around the crack tip in numerical calculations, as the process zone must be sufficiently resolved.
2.1 Continuous Galerkin Finite Element Implementation using Interface Elements

Perhaps the success and popularity of the CZM of fracture is due to the ease with which it can be incorporated within conventional finite element formulations for deforming solids. In this section, we review and summarize the computational framework and the implementation of cohesive zone models via interface elements. The notation and approach from [48] are followed.

Our starting point is the continuum formulation of finite deformation elastodynamics. Given the material points $X$ describing the reference configuration of a body occupying a region of space $B_0 \subset \mathbb{R}^3$ at time $t = t_0$, we describe the current configuration of the body at some time $t$ in the interval $T = [t_0, t_f]$ through the deformation mapping

\[ x = \varphi(X, t) \quad \forall X \in B_0, \quad \forall t \in T \] (2.9)

The deformation of infinitesimal material neighborhoods is described by the deformation gradient

\[ F = \nabla_0 \varphi(X, t) \quad \forall X \in B_0, \quad \forall t \in T \] (2.10)

where $\nabla_0$ is the material gradient operator. We must require that the Jacobian of the deformation be positive, i.e.,

\[ J = \det(F) > 0 \quad \forall X \in B_0, \quad \forall t \in T \] (2.11)

The material is loaded by body forces $\rho_0 B$ per unit reference volume and surface tractions $T$ on the boundary $\partial B_0$. Finite element formulations allowing for a cohesive response of interelement boundaries can be rigorously derived by supposing that $B_0$ is partitioned into two subbodies $B_0^\pm$ lying on the plus or minus sides of a cohesive surface $S_0$, denoted $S_0^\pm$ [48] (depicted in Figure 2-2). In this case, balance of linear
momentum over the discontinuous body requires

\[ \nabla_0 \cdot \mathbf{P} + \rho_0 \mathbf{B} = \rho_0 \dot{\mathbf{\varphi}} \quad \forall \mathbf{X} \in B_0^{\pm}, \; \forall t \in T \quad (2.12) \]

\[ \mathbf{P} \cdot \mathbf{N} = \mathbf{T} \quad \forall \mathbf{X} \in \partial B_0^{\pm}, \; \forall t \in T \quad (2.13) \]

\[ [\mathbf{P} \cdot \mathbf{N}] = [\mathbf{T}] = 0 \quad \forall \mathbf{X} \in S_0^{\pm}, \; \forall t \in T \quad (2.14) \]

where \( \mathbf{P} = \frac{\partial W(F)}{\partial F} \) is the first Piola-Kirchoff stress tensor, and \( \mathbf{N} \) is the outward reference unit surface normal. The deformation power, representing the part of the power expended on \( B_0 \) which is not expended in raising its kinetic energy can be written as

\[ P^D = \dot{W} - \dot{K} = \sum_{\pm} \int_{B_0^{\pm}} \rho_0 (\mathbf{B} - \dot{\mathbf{\varphi}}) \cdot \dot{\mathbf{\varphi}}dV_0 + \sum_{\pm} \int_{\partial B_0^{\pm}} \mathbf{T} \cdot \dot{\mathbf{\varphi}}dS_0 \quad (2.15) \]

Inserting the equations for linear momentum balance into equation (2.15) leads to a generalization of the deformation power identity to a body containing a cohesive
This identity establishes the work conjugacy relation between the cohesive tractions $T$ and the displacement jumps $[\phi]$ at the discontinuous surface. These work-conjugacy relations form the basis for a general theory of cohesive surfaces in solids where the opening displacements play the role of a deformation measure and the tractions, of a conjugate stress measure.

In an arbitrary brittle elastic body, quite complex dynamic three-dimensional crack patterns can arise. With the CZM approach, arbitrary crack growth is allowed for by introducing cohesive surfaces at some set of interior boundaries $\partial_I B_{oh}$, in a finite element discretization $B_{oh} = \bigcup_{e=1}^E \Omega^e_0$, where $\Omega^e_0$ represents the reference element with boundary $\partial \Omega^e_0$, and $E$ is the number of volumetric finite elements. The work-conjugacy relation (2.16) results in an additional term in a formulation using the principal of virtual work, representing the total virtual work done over all the cohesive surfaces in the discretized body:

$$
\int_{B_{oh}} (\rho B_0 \Phi_h \cdot \delta \Phi_h + P_h : \nabla \delta \Phi_h) dV + \int_{\partial_I B_{oh}} T(\Delta) \cdot \delta \Delta dS = 
\int_{B_{oh}} \rho B_0 \delta \Phi_h dV + \int_{\partial N B_{oh}} T \cdot \delta \Phi_h dS \quad (2.17)
$$

In general, the response of a cohesive surface will be different for opening and sliding, making it necessary to keep track of the deformed configuration of the cohesive surface. An adequate deformation measure is furnished by the mean deformation mapping defined as

$$
\Phi = \frac{1}{2} (\Phi^+ + \Phi^-) \quad (2.18)
$$

from which the full deformation mapping is recovered from

$$
\Phi^\pm = \Phi \pm \frac{1}{2} \Delta \quad (2.19)
$$

In the simplest and most popular implementation of the cohesive zone concept,
possible crack initiation sites and propagation paths are constrained to the interelement boundaries, and so $\partial B_{0h} = \left[ \bigcup_{e=1}^{E} \partial \Omega_0^e \right] \setminus \partial B_{0h}$. This requires the computation of the mean deformation mapping at all element boundaries containing cohesive surfaces throughout the calculation. First developed by Ortiz and Suresh [44], the most common FE implementation for computing the mean deformation mapping utilizes so-called “cohesive” or “interface” elements, which for the case of quadratic 10-node tetrahedral bulk elements consist of a pair of triangular 6-noded surface finite elements whose nodes coincide with those of adjacent element facets undergoing separation (see Figure 2-3).

![Figure 2-3: Description of a 12-node interface element introduced between two 10-node quadratic tetrahedra $\Omega_0^+$ and $\Omega_0^-$.](image)

Denoting the standard shape functions for each part of the cohesive element by $N_a(s_1, s_2), a = 1, ..., 6$ where $(s_1, s_2)$ are the natural coordinates of each surface element in a convenient standard configuration, the middle surface of the element is defined parametrically as

$$x(s) = \sum_{a=1}^{6} \bar{x}_a N_a(s)$$  \hspace{1cm} (2.20)
for
\[ \bar{x}_a = \frac{1}{2} (x^+_a + x^-_a) \] (2.21)

where \( x^+_a, a = 1, \ldots, n \) are the nodal coordinates of the surface elements. The tangent basis vectors to the middle surface, \( a_\alpha (s) \) are computed from
\[ a_\alpha (s) = x_\alpha (s) = \sum_{a=1}^{6} \bar{x}_a N_{a,\alpha} (s) \quad \text{for} \quad \alpha = (1, 2) \] (2.22)

and the unit normal \( n \) (which points from \( S^- \) to \( S^+ \)), from
\[ n = \frac{a_1 \times a_2}{|a_1 \times a_2|} \] (2.23)

Finally the opening displacement in the deformed configuration is computed from
\[ \Delta (s) = \sum_{a=1}^{6} [x_a] N_a (s) \] (2.24)

with
\[ [x_a] = x^+_a - x^-_a \] (2.25)

Given the opening displacement vector, the cohesive tractions are then calculated from an assumed form of the traction-separation law (equation (2.2)) and the nodal forces follow as
\[ f_{ia}^\pm = - \int_{\partial R_{ik}} T_i (\Delta (s)) N_a dS \] (2.26)

Two fundamentally different classes of TSLs have been proposed to date, differing in the assumed pre-fracture response of the cohesive surfaces. In the "intrinsic approach," Figure 2-4a, cohesive surfaces within the material are assumed to have a reversible (i.e. elastic) response prior to the onset of fracture. Conversely, in the "extrinsic approach," Figure 2-4b, cohesive surfaces are assumed to have a rigid response prior to the onset of fracture. These two basic classes of TSLs have important implications in terms of the numerical implementation, as well as in the resulting numerical properties of the overall computational framework for dynamic fracture.
Figure 2-4: Schematic of the traction-separation laws utilized in a) the intrinsic approach and b) the extrinsic approach.

In the following chapter, we describe the formulation and implementation of each type of law in detail, provide specific examples of phenomenological TSLs that have been used in practice, and discuss the numerical issues associated with each type of cohesive law.

2.2 Intrinsic Approach

Intrinsic cohesive laws for computational fracture were initially developed for modeling delamination processes at material interfaces [78, 79, 80, 81, 82, 83]. Motivated by the physics of the separation process in interface delamination, intrinsic cohesive laws assume that the cohesive traction has an initially elastic response prior to reaching a critical value, after which the traction falls to zero when new free surfaces are formed. For interface problems, the crack path is typically well-known which allows for implementation of the TSL simply as a mixed boundary condition in the finite element mesh. In the generalization of this approach to problems involving arbitrary crack initiation and propagation, the TSL is implemented at all interelement boundaries in the FE discretization using the interface element approach described in section 2.1. Since the intrinsic form of the TSL includes an initially elastic response, the cohesive elements must be present throughout the entire calculation. This is
usually done by splitting a continuous FE mesh and creating the interface element data structures prior to the calculation. Evidently, the cohesive elements, which are "intrinsically" present in the calculation, are then responsible for maintaining the compatibility and momentum transfer across elements through the TSL prior to fracture. This, in turn, requires the TSL to have an elastic (i.e. reversible) response, as well as an "intrinsic" fracture criterion, beyond which the cohesive element response is dissipative, irreversible and responsible for describing the fracture process.

**Issues With the Intrinsic Approach**

In the discussion of the intrinsic approach above, the issue of mesh dependency of the crack propagation paths was seen to arise in several studies. Two other issues associated with the intrinsic approach include spurious crack tip speed effects (usually referred to as "Lift-Off") and problems with the propagation of elastic stress waves (usually referred to as "Artificial Compliance"). These issues are now described in detail.

**Mesh Dependency of Arbitrary Crack Paths**

In the intrinsic cohesive approach for modeling arbitrary crack paths, the possible cracked configurations of the body are limited by the topology of the FE discretization. For example, crack-tips are described in the mesh by the element corners. Moreover, at these nodal crack-tips, the possible crack propagation directions are severely limited as cracks are constrained to propagate only along the adjacent interelement boundaries. Evidently the inherent mesh dependence of the intrinsic approach, as implemented using interface elements, can be avoided by employing an adaptive scheme which grows the crack incrementally in the predicted crack propagation direction at the crack-tip by building the correct crack path into the finite element mesh [76, 84, 85]. For the intrinsic approach based on interface elements, one avenue for addressing mesh dependency of the crack propagation paths is to employ very highly refined meshes in large-scale parallel simulations which has only been
Mesh dependency of the crack propagation paths was first observed in the initial simulations of Xu and Needleman [45] of arbitrary crack growth in brittle elastic materials where the predicted crack paths were shown to depend sensitively on the orientation of the triangular elements in the FE discretization. For example, for triangles oriented at ±45° and ±60°, the crack advanced in straight path before branching occurred, while for triangles oriented at ±15° and ±30°, the crack advanced in a zig-zag fashion before branching [45]. These results are reproduced in Figure 2-5. Similarly, mesh dependency of the crack propagation paths was observed in the high-resolution parallel simulations of Xu et al. in [46], where several aspects of the fracture solution including the onset of branching and curves of crack advance and crack speed versus time, were shown to be mesh dependent for three different uniform mesh spacings.

The issue of mesh dependence of the crack path was further illustrated by Scheider and Brocks [55] in their previously mentioned simulation of the cup-cone fracture of a round tensile bar. In this work, they used the same crossed-triangle quadrilateral mesh structure as in [45] and found that the correct crack path is only achieved if the aspect ratio of the quadrilateral element pattern is chosen such that the inclined cohesive surfaces are in the direction of the maximum tangential stress. In general, convergence of an arbitrary crack path with the intrinsic approach remains an elusive goal.

The inability of the intrinsic cohesive approach to provide a sufficient number of possible crack propagation directions at the crack tip has also led to mesh dependency in brittle fragmentation studies. For example, in the aforementioned study of Miller, et al. [53] it was shown that reducing the spacing between cohesive surfaces in the finite element mesh greatly increased the number of small fragments. Convergence of the fragment size distribution predicted by the intrinsic approach for fragmentation of an arbitrary brittle elastic body also remains an elusive goal.
Figure 2-5: Mesh dependence of branching pattern from [45] for triangular elements oriented at (a) 15°, (b) 30°, (c) 45°, (d) 60°

**Lift-off**

Another general issue with the intrinsic approach has to do with a phenomenon called “lift-off,” which was first addressed in [80]. The basic problem is that some amount of finite opening occurs across cohesive elements which precludes subsequent failure before reaching the critical normal separation. If many element boundaries have opened sufficiently close to the critical opening along a potential crack path, then a sufficiently high loading can cause instant failure of all cohesive surfaces along the path. This problem is noticeable when the crack is confined to a straight path. The resulting behavior is a spuriously-high crack-tip speed [80, 45, 46]. In [45], lift-off was observed for a symmetric impact loading of 30 m/s which produced crack speeds...
Figure 2-6: Mesh dependence of fracture pattern and of the force-thickness reduction curve from [55] for different aspect ratios of the crossed-triangle mesh structure - aspect 1 = 0.125x0.046 mm$^2$, aspect 2 = 0.125x0.054 mm$^2$. Note $T_{0,N} = \sigma_c$ for the notation used in this work.

exceeding the Rayleigh wave speed of the material, Figure 2-7. The issue of lift-off is a major drawback for the intrinsic approach in the modeling of fixed crack paths under extreme loading conditions (e.g. interface delamination under impact loading).

**Artificial Compliance**

In the typical finite element implementation for hyperelastic solids in the absence of discontinuities, consistency of the formulation is guaranteed by the strong enforcement of interelement continuity. However, the intrinsic approach assumes that discontinuities in the displacement fields occur in the uncracked body at the interelement boundaries from the start of the calculation. These displacement jumps give rise to the an artificial elasticity at the interelement boundaries, akin to adding a spring between every element in the FE mesh, whose stiffness depends on the initial slope of the cohesive law. Hence, cohesive surfaces in the intrinsic approach necessarily alter the pre-fracture elastic response of the material at the interelement boundaries in the uncracked body. As a result, the consistency of the method is not preserved. Specif-
Figure 2-7: Plot of crack tip speed, $\dot{a}$ vs time, $t$ showing lift-off from [45]. Dashed line is the Rayleigh wave speed.

ically, the elastic response of interelement boundaries causes the partial transmission and partial reflection of stress waves incident on cohesive surfaces in the uncracked body. In the pre-fracture regime, this leads to an underprediction of the wave speed and stress in uniaxial wave propagation [86], and to an artificial anisotropy in three-dimensions due to impedance mis-match across the cohesive surfaces in the material [87].

This phenomenon, sometimes called artificial compliance, can be studied in the context of simple model problems. For instance, Klein, et al. [87] quantified the effect of artificial compliance by deriving an expression for the effective elastic modulus of a one-dimensional network of cohesive surfaces at constant spacing $h$, interspersed in a homogeneous material of elastic modulus, $E$. For an intrinsic cohesive relation with initial slope $s$, the effective modulus, $E_{eff}$, is given by

$$E_{eff} = E \left[ 1 - \frac{1}{1 + (sh/E)} \right]$$  \hspace{1cm} (2.27)
From this expression, we see that $E_{eff} \rightarrow E$ for either $s \rightarrow \infty$ or $h \rightarrow \infty$. Since the cohesive element spacing is directly related to the element size for an intrinsic approach, equation (2.27) shows that the effects of artificial compliance are also mesh-dependent. Indeed, based on the parameters used by Xu and Needleman in [45], this simple model estimates that the effective modulus for the finest mesh used in that study was only 60% of the actual value [87].

As suggested by equation (2.27), the effect of artificial compliance can be made negligible by forcing the initial slope of the cohesive law to be very large. For the potential-based law [88] this is achieved by choosing a large value for the critical cohesive strength. However, this precludes the calibration of the critical cohesive strength to experimental results. More freedom in the selection of the initial cohesive law slope is obtained by employing TSLs of the “bilinear”-type which were proposed by Geubelle and Baylor [47] and Espinosa, et al. [89, 90]. The latter formulation of [89] uses the mixed-mode non-dimensional displacement jump $\lambda = \sqrt{\left(\frac{\Delta u_n}{\sigma_n}\right)^2 + \left(\frac{\Delta u_t}{\sigma_t}\right)^2}$, of [81] to define the normal and tangential cohesive tractions. The tractions are defined such that they increase linearly up to some maximum value for $0 \leq \lambda \leq \lambda_{cr}$ and then decrease linearly to zero for $\lambda_{cr} < \lambda \leq 1$.

This formulation is particularly useful because $\lambda_{cr}$ acts as a penalty parameter to control the initial slope of the cohesive law, and can be chosen small enough so that wave speeds of the material are unaffected by the presence of cohesive elements. Espinosa and Zavattieri [91, 86] demonstrated this with a numerical experiment to study the effect of the initial cohesive law slope on wave propagation. The authors considered a 2D RVE of polycrystalline ceramic consisting of two identical finite element meshes with a layer of cohesive elements between them. The RVE was loaded perpendicular to the plane of cohesive elements at one end with a viscous boundary condition, the opposite boundary was left free, and periodic boundary conditions were imposed on the specimen sides resulting in the propagation of a uniaxial tensile stress wave perpendicular to the plane of cohesive elements.

As a simplification, the cohesive elements were restricted to open in a purely normal mode, so for the bilinear-type cohesive law proposed in [91], the TSL reduces
to

\[ T_n = \left( \Delta_n \over \delta_n \right) \frac{\sigma_c}{\lambda_{cr}} \]  

(2.28)

for \( \Delta_n \leq \lambda_{cr} \delta_n \) and

\[ T_n = \frac{(1 - \Delta_n)}{\delta_n} \frac{\sigma_c}{(1 - \lambda_{cr})} \]  

(2.29)

for \( \lambda_{cr} \delta_n < \Delta_n \leq \delta_n \). The initial slope \( s \) of the cohesive law is given by

\[ s = \frac{\sigma_{\text{max}}}{\lambda_{cr} \delta_n} \]  

(2.30)

For numerical comparisons, the time-history of the normal cohesive traction \( T_n \) is compared to the exact value of the normal stress component in an identical mesh without cohesive elements, for increasing values of the cohesive law slope. The results (reproduced in figure 2-8) show that for small values of \( s \), the normal cohesive traction is much lower than the correct value. As \( s \) increases, the agreement improves and convergence to the correct value is achieved for very large \( s \) values. However, as the slope is increased further a numerical instability occurs. The authors conclude that this simple test is an efficient way to determine the correct value of the cohesive slope to avoid problems of wave propagation, and alternatively, a good rule of thumb is to take \( s \geq 10 \, E/h \), where \( h \) is the characteristic size of the volumetric element [91].

While increasing the cohesive law slope reduces the effect of artificial compliance, it can place a severe restriction on the time step due to necessity of following the ascending branch of the TSL in a sufficient number of steps for stability. To account for this, Espinosa and Zavattieri [91] devised an adaptive scheme for setting the time step which guarantees that it is sufficiently small. First a time step \( \Delta t_{\text{cont}} \) is calculated in the standard way based on elastic properties and the FE discretization. Then a second time step, \( \Delta t_{\text{coh}} \) is calculated from

\[ \Delta t_{\text{coh}} = \frac{\Delta t_{\text{cont}}}{\max_i \zeta_i N} \]  

(2.31)
Figure 2-8: Wave propagation experiments from [91] for increasing values of the cohesive law slope $s$. 

$\text{s = 160, 220, 1200, 6400, 12000, 25000}$
with the $\zeta_i$ computed over each cohesive surface from

$$
\zeta_i = \begin{cases} \\
\frac{\Delta \lambda}{\lambda_{cr}} & \text{if } 0 \leq \lambda \leq \lambda_{cr} \\
\frac{\Delta \lambda}{1-\lambda_{cr}} & \text{if } \lambda_{cr} < \lambda \leq 1
\end{cases}
$$

(2.32)

In the above expressions, $\Delta \lambda = \lambda_{n+1} - \lambda_n$ where $\lambda_{n+1}$ is the local displacement jump predictor, and $N$ is the number of steps to be taken for $\lambda$ to progress from 0 to $\lambda_{cr}$. The overall time step is then chosen as $\Delta t = \min(\Delta t_{cont}, \Delta t_{coh})$ [91]. Since the cohesive time step is proportional to $\lambda_{cr}$, a sufficiently small value of $\lambda_{cr}$ can severely restrict the time step, leading to excessively long computation times.

2.3 Extrinsic Approach

Pioneered by Ortiz and coworkers, the extrinsic cohesive approach assumes that cohesive surfaces exhibit an initially rigid response. For modeling arbitrary crack initiation and propagation in brittle materials with the extrinsic approach, the TSL is implemented through the use of interface finite elements between the volume finite elements, but contrary to the intrinsic approach, the TSL starts operating only after some failure criterion is satisfied. Due to the initially rigid response of the cohesive surfaces, adjacent continuum element boundaries remain coincident prior to the onset of fracture, thus completely avoiding the issue of pre-fracture artificial compliance. When the fracture criterion is met, interface elements (see section 2.1) are dynamically inserted in the computational mesh. This, in turn, induces topological changes in the mesh which requires a dynamic modification of the mesh entities (e.g. insertion of new nodes, reconnection of existing elements) throughout the calculation. This process is particularly involved in 3D and has been treated at length in [92, 93, 94, 95].

We now describe the formulation and applications of the commonly implemented extrinsic TSL (sometimes referred to as the "Linear Irreversible Softening Law") and discuss the associated numerical issues.
Linear Irreversible Softening Law

The most widely-used extrinsic cohesive law is the linear irreversible softening law, modified from the one originally proposed by Camacho and Ortiz [3] in the context of simulations of fragmentation of brittle materials under impact loading. This law was formally presented and discussed in detail by Ortiz and Pandolfi in [48] under the general framework described in section 2.1. The authors assume that the cohesive behavior is isothermal, isotropic, independent of the bulk constitutive response, and that the free energy density is dependent on $\Delta$ only through an effective separation $\delta$ defined by

$$\delta = \sqrt{\beta^2 \Delta_t^2 + \Delta_n^2}$$

(2.33)

in the spirit of Tvergaard [81]. In this expression, $\beta$ is a parameter which assigns different weights to normal and tangential separation, and the dependence on $\Delta_t = |\Delta_t|$ is a result of the isotropic assumption. Under this set of assumptions, one can define an effective cohesive traction $T$ which is given by

$$T = \frac{\partial \phi (\delta, \mathbf{q})}{\partial \delta}$$

(2.34)

where $\mathbf{q}$ is a suitable set of internal variables, governed by kinetic relations, which describe the irreversible processes involved in decohesion. This law becomes operative when the fracture criterion

$$\sqrt{\sigma_h : [\mathbf{n} \otimes \mathbf{n}] + \beta^{-2} \sigma_h : [\mathbf{n} \otimes \mathbf{m}] \geq \sigma_c}$$

(2.35)

is satisfied. In this expression, $\mathbf{n}$ and $\mathbf{m}$ are local unit normal and tangent to the deformed cohesive mid-surface, $\sigma_h$ is the Cauchy stress, and $\sigma_c$ is the critical cohesive strength.

For this particular cohesive law, a specific functional form $T = T (\delta)$ is assumed for the effective traction. After the fracture criterion 2.35 is satisfied, the effective traction is determined from this functional form and the components of the local
traction vector follow from

\[
T = \frac{T}{\delta} (\beta^2 \Delta_t + \Delta_n n)
\]  

(2.36)

In the specific case of the linear irreversible softening law, the functional form of the effective cohesive traction for crack opening is given by

\[
T(\delta, \delta_{\text{max}}) = \sigma_c \left(1 - \frac{\delta}{\delta_c}\right) \quad \text{for } \delta \geq 0, \delta = \delta_{\text{max}}
\]  

(2.37)

where complete decohesion \((T = 0)\) occurs for \(\delta \geq \delta_c\). In reference to Figure 2-9, the variable \(\delta_{\text{max}}\) is the maximum effective opening displacement and constitutes the internal variable describing irreversibility. For crack closure, characterized by \(\delta < 0\) or \(\delta < \delta_{\text{max}}\), the functional form of the linear irreversible softening law is assumed to follow a straight path to and from the origin yielding

\[
T(\delta, \delta_{\text{max}}) = \frac{T_{\text{max}}}{\delta_{\text{max}}} \delta \quad \text{for } \delta < 0, \text{or } \delta < \delta_{\text{max}}
\]  

(2.38)

where \(T_{\text{max}}\) is the value of the effective traction at \(\delta = \delta_{\text{max}}\). Inserting the definition of the effective cohesive traction, equation (2.37), in equation (2.35), we find that the work of separation for the linear softening law is simply

\[
\phi_{\text{sep}} = \frac{1}{2} \sigma_c \delta_c
\]  

(2.39)

Figure 2-9 depicts the \(T - \delta\) relationship for the linear softening law.

Crack closure after full fracture of interface elements can be handled within the TSL, e.g. [78, 79, 88], or as suggested by Camacho and Ortiz [3], by a separate contact enforcement algorithm. However, it should be emphasized that the former approach reintroduces the artificial compliance issue (see section 2.2) e.g., under compressive waves propagating through closed cracks. Moreover, existing extrinsic (or intrinsic) TSLs neglect the post-fracture frictional response of cracked surfaces.
Issues with the Extrinsic Approach

While the fidelity of the extrinsic approach has been demonstrated for modeling known crack paths, various issues arise in modeling arbitrary brittle fracture including mesh dependency of the fracture path and dissipated cohesive energy, time-discontinuity, and a lack of scalability for 3D parallel implementations. These issues are now discussed in detail.

Mesh Dependency of Arbitrary Crack Paths

Evidently the extrinsic approach, as implemented using the interface element approach (section 2.1), faces the same mesh dependency issues for the intrinsic approach discussed in detail in section 2.2. In the context of the extrinsic approach, these issues have been identified in [49, 50, 57]. One example is the study of Zhang et al. [57], investigating the brittle microbranching instability through simulation of Mode I fracture in a pre-cracked PMMA strip. While crack branching was successfully demonstrated in simulations, the overall fracture pattern was shown to vary with successive refinement of the mesh (see Figure 2-10). Likewise, in the previously
mentioned studies on three-point bending, Ruiz et al. observed mesh dependence of the crack path in simulation of pre-notched specimens with two different mesh sizes. While the load-history curves for the two meshes were in good agreement, the predicted crack paths were shown to be mesh dependent. Moreover, for the fine mesh, a much broader region of microcracking was observed, along with a much higher amount of dissipated cohesive energy (see Figure 2-11).

Figure 2-10: Mesh dependence of the fracture pattern for symmetrically loaded pre-cracked PMMA strip for (a) 32 x 128 elements at \( t = 24\mu s \), (b) 48 x 192 elements at \( t = 22\mu s \) and (c) 48 x 192 elements at \( t = 21\mu s \) from [57]

In the context of brittle materials, Zhou and coworkers [50] investigated mesh dependency for the extrinsic approach through simulation of 2D axisymmetric ceramic ring fragmentation and of 3D Mode I fracture in a pre-notched PMMA block. The authors found generally that in these highly symmetric problems, the resulting cracks propagate in preferred directions depending sensitively on the orientation of the interelement boundaries. Based on the results, the authors recommend that extrinsic approach necessarily requires unstructured random meshes with relatively uniform element sizes. As a possible remedy for mesh dependency for simulation of brittle materials, Zhou et al. [50] proposed a modification to the linear irreversible softening

55
Figure 2-11: Comparison of crack path, load history, and dissipated cohesive energy for two different mesh sizes from [49]
law where the critical cohesive strength is spatially non-uniform and conforms to a modified Weakest-Link Weibull distribution. In this approach the strength of each interelement facet decreases with the facet area, motivated phenomenologically by the fact that larger material regions are likely to contain more defects. Using the modified TSL, they showed that the mesh-dependency observed in the previous two examples was significantly reduced.

Lastly, a notable result concerning mesh dependency is due to Papoulia et al., who proved convergence of the crack nucleation site for a so-called “pin-wheel” based mesh in [56]. This advanced mesh structure, which was implemented in [56] in 2D using the interface element approach, utilizes internal subdivision inside of triangular elements in a remeshing procedure which preserves the length of potential crack paths with refinement of the mesh. This allows for the exact representation of an arbitrary crack path in the limit of the vanishing element size. The authors speculate that this specific property of pin-wheel based meshes may be a necessary requirement for any proof of convergence in the crack path for cohesive element methods. Further, they found that in 2D simulations of three point-bending in an elastic material, convergence in the predicted crack nucleation site and subsequent propagation path was much faster for the pin-wheel mesh than for conventional quadratic triangular meshes.

Mesh Dependency of Dissipated Fracture Energy

Motivated by the large mesh-dependent increase in dissipated cohesive energy depicted in Figure 2-11, Molinari et al. [59] conducted the first systematic investigation into the energy convergence properties of the extrinsic cohesive element approach. To demonstrate the issue, they used the benchmark test of dynamic Mode I fracture of a pre-notched PMMA strip. They not only confirmed the dependence of dissipated cohesive energy on the mesh size, but they also showed that convergence could not be achieved even with fairly large meshes, see Figure 2-12.

Based on these results, convergence was then studied for a simpler model problem of fragmentation of a complete ring specimen of an elastic material, discretized with one-dimensional line elements. The results of the convergence study are reproduced...
in Figure 2-13. They show that for uniform and random element sizes at varying strain rates, the extrinsic cohesive approach is indeed convergent in the dissipated fracture energy when the mesh size goes to zero. In the case of a uniform mesh, the convergence is not monotonic and the errors are large unless extremely large meshes are used. However, convergence is monotonic and can be achieved for much coarser meshes when some randomness is introduced in the element sizes. However, even in the best case scenario in this one-dimensional setting, a minimum of approximately $10^4$ nodes are necessary to obtain reasonably converged results. Assuming that similar convergence behavior occurs for multiple dimensions, these results indicate that at least $10^8$ and $10^{12}$ nodes would be necessary to attain energy convergence in two and three dimensions, respectively. Furthermore, additional results reported in this study (see Figure 2-14) show that the size distribution of fragments is highly sensitive to the mesh size up to approximately $10^5$ nodes, upon which a degree of convergence is observed for a subsequent increase in the mesh size.
Figure 2-13: Energy convergence for 1D ring fragmentation for uniform and random meshes from [59]

Figure 2-14: Dependence of size distribution of fragments on mesh size from [59]
Scalability Issues for Three-Dimensional Problems

The previous issues of mesh convergence affecting CZMs of fracture underscore the need for large scale simulations, specially in three dimensions. This requires both data structures and algorithms enabling the efficient dynamic insertion of interface elements as cracks nucleate and propagate and the management of the associated arbitrary topological mesh changes. In addition, large scale simulation can only be achieved if the algorithms can be implemented for parallel computation and are scalable when exercised on large numbers of processors for problems of increasing size. The issues related to the efficient sequential implementation of fracture algorithms in 2D and 3D have been addressed in [92, 93, 95, 94]. In the initial approach developed by Pandolfi and Ortiz [92, 93], the algorithm is based on data structures which provide a comprehensive representation of the relevant topological entities in the mesh (face, edge, vertex) and their corresponding adjacency relationships. Paulino et al. [94] presented a new data structure which tried to significantly reduce the large storage space required by Pandolfi et al.'s approach and to increase the computational efficiency. The new method is based on an element representation which only stores information on the nodes belonging to the element and to neighboring elements, allowing for the implicit recovery of all facet, edge, and vertex information. The algorithm was shown to scale linearly in time in the number of interfaces to fracture (denoted $N_I$) in single processor simulations.

In [95], Mota et al. presented an alternative insertion algorithm based on a graph representation of the finite element mesh. It was shown that this algorithm also scales linearly in time with the number of interfaces to fracture, as compared to the approach in [92, 93] whose complexity they showed to increase as $O(N_I^{1.9})$. Although the authors mention the suitability of this mesh representation for parallel computation, there is no detail on how the implementation would allow for crack propagation across processor boundaries and the parallel performance analysis is deferred to a subsequent work.
**Time Discontinuity**

An additional drawback of the linear irreversible softening formulation presented in this section is that the interelement tractions are necessarily time-discontinuous for the time step immediately following cohesive element insertion [96, 97]. This is due to the fact that prior to fracture, the interelement tractions are extrapolated to the element boundaries from the quadrature points and so they depend on the stress field within the neighboring continuum elements. However, in the subsequent time step following cohesive element insertion, the interelement tractions depend only on $\Delta$ and $\beta$ through the cohesive law. Hence, there is no way of guaranteeing that the tractions will remain continuous throughout the calculation. Some unfortunate behaviors which have been identified as a result of time discontinuity are unphysical oscillations and non-convergence in time [96]. In order to resolve the time-discontinuity of the linear irreversible softening formulation, Papoulia *et al.* [96] proposed a modification to the TSL where the values of traction components resolved on the cohesive surface just prior to the onset of fracture are specially encoded into the model as initial values for the components of the cohesive tractions. Further investigations into obtaining time-continuous formulations for the linear irreversible softening law can be found in [97].
Chapter 3

DG/CZM: A Scalable 3D Approach for Extreme-Scale Dynamic Fracture Simulation

As discussed in the previous chapter, existing algorithmic approaches for modeling dynamic brittle fracture based on CZMs are mired by a number of issues, the most prominent being artificial compliance and a lack of scalability. A promising alternative method, which is developed in this thesis, can be found in the combination of cohesive theories of fracture and the discontinuous Galerkin (DG) formulation of the continuum dynamic problem. The main feature of interest in the DG framework is that it allows for discontinuous displacement jumps between elements with consistency and stability of the finite element solution guaranteed by additional terms in the weak statement of the problem. In the new method proposed herein, interface elements are inserted at interelement boundaries at the beginning of the simulation which proceeds using a DG approach in the pre-fracture regime. The consistency and stability provided by the DG terms ensures that prior to fracture, issues with artificial compliance are avoided and stress wave propagation is captured accurately. When a specified fracture criterion is met at an interelement boundary, the computation of the DG interface flux terms gives its place to an extrinsic cohesive law which describes the irreversible traction-separation response eventually leading to complete decohesion.
and the formation of new crack surfaces. The key advantage of the proposed method is that topological changes to the mesh and modification to interprocessor communication data structures are completely avoided as the mesh is fragmented initially by requirement of the DG method. This allows cracks to nucleate and propagate in the finite element mesh without any need for special treatment of cracks propagating across processor boundaries, enabling the possibility for scalable fracture simulations in 3D. In the following we describe the formulation and finite element discretization of the DG/CZM method along with our scalable parallel implementation of the method in the MIT Summit code.

3.1 DG/CZM Weak Formulation

Consider a body $B_0$ subjected to a force per unit mass $B$. Its boundary surface $\partial B_0$ is partitioned into a Dirichlet portion $\partial_D B_0$ constrained by displacements $\vec{\varphi}$ and a Neumann part $\partial_N B_0$ subjected to surface traction $\vec{T}$. One always has $\partial B_0 = \partial_N B_0 \cup \partial_D B_0$ and $\partial_D B_0 \cap \partial_N B_0 = \emptyset$. The continuum equations stated in material form are

\[
\rho_0 \ddot{\varphi} = \nabla \cdot P + \rho_0 B \quad \text{in } B_0 \quad (3.1)
\]
\[
\varphi = \vec{\varphi} \quad \text{on } \partial_D B_0 \quad (3.2)
\]
\[
P \cdot N = \vec{T} \quad \text{on } \partial_N B_0 \quad (3.3)
\]

In these relations $\rho_0$ is the initial density, $P$ is the first Piola-Kirchhoff stress tensor, $N$ is the unit surface normal in the reference configuration.

The discontinuous Galerkin (DG) weak form of Eqs. (3.1-3.3) arises by seeking an elementwise-continuous polynomial approximation $\varphi_h$ of the deformation over the discretization $B_{oh} = \bigcup_{e=1}^{E} \bar{\Omega}_e^0$ of $B_0$, where $\bar{\Omega}_e^0$ is the union of the open domain $\Omega_e^0$ with its boundary $\partial \Omega_e^0$, i.e. $\varphi_h \notin C^0 (B_{oh})$, as in continuous Galerkin approximations, but $\varphi_h \in C^0 (\Omega_e^0)$. Consequently, for a DG formulation the trial functions $\delta \varphi_h$ are also discontinuous across the element interfaces on the internal boundary of the body.
\[ \partial_1 B_{0h} = \left[ \bigcup_{e=1}^{E} \partial \Omega_0^e \right] \setminus \partial B_{0h}. \]

The new weak formulation of the problem is obtained in a similar way as for the continuous Galerkin approximation. The strong form (3.1) of the linear momentum balance is enforced in a weighted-average sense by multiplying by a suitable test function \( \delta \varphi_h \) and integrating in the domain. However, since both test and trial function are discontinuous, the integration by parts is not performed over the whole domain, but on each element instead, leading to

\[ \sum_e \int_{\Omega_0^e} (\rho_0 \ddot{\varphi}_h \cdot \delta \varphi_h + \mathbf{P}_h : \mathbf{\nabla}_0 \delta \varphi_h) \, dV - \sum_e \int_{\partial \Omega_0^e \cap \partial_1 B_{0h}} \delta \varphi_h \cdot \mathbf{P}_h \cdot \mathbf{N} \, dS = \]
\[ \sum_e \int_{\Omega_0^e} \rho_0 \mathbf{B} \cdot \delta \varphi_h \, dV + \sum_e \int_{\partial \Omega_0^e \cap \partial_1 B_{0h}} \delta \varphi_h \cdot \mathbf{T} \, dS \]  

(3.4)

In this equation the discretized stress tensor \( \mathbf{P}_h \) results from the discretized deformation state \( \mathbf{F}_h = \mathbf{\nabla}_0 \varphi_h \) through a constitutive material law. Equation (3.4) can be written as

\[ \int_{B_{0h}} (\rho_0 \ddot{\varphi}_h \cdot \delta \varphi_h + \mathbf{P}_h : \mathbf{\nabla}_0 \delta \varphi_h) \, dV + \int_{\partial_1 B_{0h}} \left[ \mathbf{\delta \varphi}_h \cdot \mathbf{P}_h \right] \cdot \mathbf{N}^- \, dS = \]
\[ \int_{B_{0h}} \rho_0 \mathbf{B} \cdot \delta \varphi_h \, dV + \int_{\partial_2 B_{0h}} \delta \varphi_h \cdot \mathbf{T} \, dS \]  

(3.5)

where we have used the jump operator defined on the interface of two finite elements by

\[ [\bullet] = [\bullet^+ - \bullet^-] \]  

(3.6)

The main idea of the DG method is to address the contribution of the interelement discontinuity terms by introducing a numerical flux \( \mathbf{h}(\mathbf{P}^+, \mathbf{P}^-, \mathbf{N}^-) \) dependent on the limit values on the surface from the neighboring elements, such that

\[ \int_{\partial_1 B_{0h}} \left[ \mathbf{\delta \varphi}_h \cdot \mathbf{P}_h \right] \cdot \mathbf{N}^- \, dS \rightarrow \int_{\partial_1 B_{0h}} \left[ \mathbf{\delta \varphi}_h \right] \cdot \mathbf{h}(\mathbf{P}^+, \mathbf{P}^-, \mathbf{N}^-) \, dS \]  

(3.7)

where \( \mathbf{N}^- \) is the outward unit surface normal for a given element. Rewriting the term
in question we find
\[
\int_{\partial \Omega} [\delta \varphi_h \cdot \mathbf{P}_h] \cdot \mathbf{N}^- dS = \int_{\partial \Omega} [\delta \varphi_h] \cdot (\mathbf{P}_h) \cdot \mathbf{N}^- dS + \int_{\partial \Omega} [\delta \varphi_h] \cdot [\mathbf{P}_h] \cdot \mathbf{N}^- dS
\] (3.8)
where the average operator defined by
\[
\langle \bullet \rangle = \frac{1}{2} [\bullet^+ + \bullet^-]
\] (3.9)
has been used. The last term in equation (3.8) can be neglected because the jump in \(\mathbf{P}_h\) does not require penalization to ensure consistency. Hence, \(h\) is chosen to be
\[
h (\mathbf{P}^+, \mathbf{P}^-, \mathbf{N}^-) = \langle \mathbf{P}_h \rangle \cdot \mathbf{N}^-
\] (3.10)

This form of the numerical flux was proposed by Bassi and Rebay [63] in the first DG formulation concerning elliptic equations. Other forms for the numerical flux are possible and can be found in the work of Arnold et al. [98] and Brezzi et al. [65].

Using the choice of numerical flux from equation (3.10), the weak formulation reduces to
\[
\int_{\Omega_B} \rho_0 \dot{\varphi}_h \cdot \delta \varphi_h + \mathbf{P}_h : \nabla_0 \delta \varphi_h \, dV + \int_{\partial \Omega_B} [\delta \varphi_h] \cdot (\mathbf{P}_h) \cdot \mathbf{N}^- dS = \int_{\Omega_B} \rho_0 \mathbf{B} \cdot \delta \varphi_h \, dV + \int_{\partial \Omega_B} \delta \varphi_h \cdot \mathbf{T} dS \] (3.11)

Since the interelement displacement continuity is not enforced strongly in a DG formulation, it must be enforced weakly which, in turn, ensures stability of the numerical solution. To this end, the compatibility equation \(\varphi_h^- - \varphi_h^+ = 0\) on \(\partial \Omega_B\) is enforced through a (sufficiently large) quadratic stabilization term in \(\{\varphi_h\}, [\delta \varphi_h]\).

In scalar problems this can be achieved by simply adding a term proportional to the scalar product \(\{\varphi_h\} \cdot [\delta \varphi_h]\). However, an appropriate term accounting for the material and mesh dimension must be proportional to \(\{\varphi_h\} \cdot [\delta \varphi_h] \cdot \frac{\mathbf{N}^-}{h_s} \cdot \mathbf{C} \cdot \mathbf{N}^-\), where \(\mathbf{C} = \frac{\partial \mathbf{P}}{\partial \mathbf{F}}\) is the Lagrangian tangent moduli, and \(h_s\) is the mesh size. With the addition of this quadratic term, general displacement jumps are stabilized in the numerical so-
olution and large-deformation material response is properly accounted for. The final formulation of the problem consists of finding $\varphi_h$ such that

$$
\int_{B_{Oh}} (\rho_0 \varphi_h \cdot \delta \varphi_h + P_h : \nabla_u \delta \varphi_h) dV + \int_{\partial_f B_{Oh}} [\delta \varphi_h] : (P_h) \cdot N^- dS +
$$

$$
\int_{\partial_f B_{Oh}} \{ [\delta \varphi_h] \otimes N^- : \left( \frac{\beta_s}{h_s} C \right) : \{ \varphi_h \otimes N^- \} \} dS =
$$

$$
\int_{B_{Oh}} \rho_0 B \cdot \delta \varphi_h dV + \int_{\partial N B_{Oh}} \delta \varphi_h \cdot \mathbf{T} dS \quad (3.12)
$$

where $\beta_s$ plays the role of a penalty parameter. This formulation, known as the Interior Penalty Method, has been shown to be stable (for $\beta_s > 1$), consistent and to possess the optimal convergence rate in the energy norm [68].

The extension of this DG framework to explicit dynamics time integration including parallel implementation was presented in [69], where it was also shown that the stable time step is reduced by a factor of $\sqrt{\beta_s}$ as compared to a CG formulation, i.e

$$
\Delta t < \Delta t_{\text{crit}} = \frac{h_s}{\sqrt{\beta_s}c} \quad (3.13)
$$

where $c$ is the sound speed of the material. More details concerning this approach, and in particular the numerical implementation based on interface elements can be found in [68, 69].

A dynamic simulation proceeds initially and prior to the nucleation of cracks according to the above DG framework. The onset of fracture is effected in the same manner as in the extrinsic CZM approach, i.e. following a fracture stress criterion. Upon the nucleation of a crack at an interface element, the DG flux terms cease to operate and give place to the TSL governing the fracture process in the material. It should be noted that this does not require any modifications of the mesh, but simply a change in the terms evaluated at the interface element integration points. Hence, if $\mathbf{T}$ is the surface traction resulting from the TSL in the reference configuration,
In this equation $\alpha$ is a binary operator defined as $\alpha = 0$ before fracture and $\alpha = 1$ after the fracture stress criterion is met. It should be noted that this approach provides for the possibility of partially fractured interface elements as $\alpha$ may adopt different values at different quadrature point of each interface element.

Equation (3.14) can also help to understand some of the numerical problems in the intrinsic CZM. Consistency is a requirement for convergence which, in the DG formulation for the uncracked body ($\alpha = 0$), is satisfied by the DG flux term: $[\delta \varphi_h] \cdot \langle P \rangle \cdot N^-$, as shown in [68]. By contrast, in the intrinsic CZM formulation the flux term is $T[[\varphi_h]] \cdot [\delta \varphi_h]$, which violates consistency. In fact, as stated, this term corresponds to the stabilization term of the DG formulation, which only enforces consistency in the limit when the tangent in the reversible range of the TSL $T(\varphi_h)$ tends to infinity [87]. This creates severe restrictions in the critical time step size in intrinsic CZM methods, a problem that is avoided in the discontinuous Galerkin method where the only penalty in the critical time step size is a factor of $\sqrt{\beta_s}$, as discussed above (3.13).

In most problems involving dynamic fracture, we have found that the DG stability term with a choice of $\beta$ in the range $1 < \beta_s < 10$ is enough to ensure numerical stability [69]. However, in problems involving fragmentation under intense compressive loads (e.g. in simulations of ballistic impact), we have observed numerical instabilities stemming from the formation of high frequency oscillations at element interfaces. A practical solution to this problem can be found in the introduction of an artificial viscosity term at the interface element level which is proportional to the time derivative
of the displacement jump, $[\dot{\varphi}_h]$. The modified DG/CZM formulation incorporating the interface viscosity term is given by

$$\int_{B_{oh}} (\rho_0 \ddot{\varphi}_h \cdot \delta \varphi_h + P_h : \nabla \delta \varphi_h) dV + \int_{\partial t B_{oh}} \alpha T ([\varphi_h]) : [\delta \varphi_h] dS$$

$$+ \int_{\partial t B_{oh}} (1 - \alpha) [\delta \varphi_h] : (P_h) \cdot N^d S +$$

$$\int_{\partial t B_{oh}} (1 - \alpha) [\delta \varphi_h] \otimes N^- : (\frac{\partial s}{\partial t}) C : [\varphi_h] \otimes N^- dS +$$

$$\int_{\partial t B_{oh}} (c_v [\varphi_h]) : [\delta \varphi_h] dS = \int_{B_{oh}} \rho_0 B \cdot \delta \varphi_h dV + \int_{\partial N B_{oh}} \delta \varphi_h \cdot T dS$$  \hspace{1cm} (3.15)

In this expression, $c_v$ is a parameter which determines the amplitude of the viscous forces at the interface. In practice, we have found that $c_v \sim O (10^{-2} - 10^{-1})$ MPa-s-m$^{-1}$ is sufficient to suppress the formation of oscillatory modes, allowing for stable simulations involving pervasive fragmentation with impact speeds in the ballistic range.

An important advantage of the DG/CZM formulation is that the cohesive law operates strictly at the quadrature point. This implies that within an interface element it is allowed to have both “cracked” and “uncracked” quadrature points. This affords the possibility of sub-element crack resolution. By contrast, in the extrinsic CZM all quadrature points of the interface element simultaneously respond according to the TSL when the fracture criterion is met at a single point in the mesh. Furthermore, in the DG/CZM formulation, the evaluation of the fracture criterion is naturally effected at the quadrature points of the original DG interface element, i.e. the new cracked surface occurs exactly at the same point where the fracture criterion is met. In the extrinsic approach, by contrast, the fracture criterion is commonly evaluated at the quadrature points of the bulk elements and the cohesive element is inserted at the closest interelement boundary. These two advantageous aspects of the DG formulation should render the method more accurate at describing crack nucleation.
3.2 Cohesive Law and Post Fracture Contact

The proposed approach is general in the sense that it does not rely on any particular assumption about the specific TSL employed in the description of fracture. It can thus be adapted to a wide class of brittle and ductile fracture behavior. For definiteness, we here adopt the irreversible, linear softening cohesive law of [48], presented in detail in section 2.3. In order to track the progression of damage at the interface element level, we define an additional internal variable at the quadrature points which is given by

\[ D = \min\left(\frac{\delta_{\text{max}}}{\delta_c}, 1\right) \]  

(3.16)

In this equation, the damage variable \(0 \leq D \leq 1\) provides a convenient measure of the extent of damage at each quadrature point, with \(D = 0\) and \(D = 1\) corresponding to an undamaged and completely damaged interface, respectively.

In high velocity impact problems, complicated mixed-mode fracture patterns often occur. In this situation, newly formed crack surfaces can be unloaded through interactions with stress waves, causing the crack faces to close and come back into contact. The numerical framework should ensure in this case that there is no inter-penetration between the crack surfaces and that momentum can be transferred across the interface between these surfaces when they have come back into contact. Fortunately, the inter-element DG flux terms provide a natural framework for enforcing a contact constraint upon complete closure of adjacent cracked surfaces in the post-fracture regime. Toward this end, we monitor the normal component of the displacement jump at every interface element quadrature point where fracture has occurred. When inter-penetration is detected at an interface element quadrature point, we assume that the normal response is governed by the continuum response, and therefore fall back to the DG form of the interface terms. This guarantees that compressive stress wave components can propagate across the closed crack surface as in the uncracked body. Upon crack closure, the tangential response is still governed by the TSL. Improvements on this model would incorporate a frictional component in the tangential response, but this has been neglected in the present work for simplicity and due to

70
lack of experimental data supporting this need.

3.3 Finite Element Discretization and Explicit Time Integration

The weak formulation of the dynamics problem presented above is taken as a basis for finite element discretization. To this end, the deformation mapping, its first variation and the material acceleration field are respectively approximated by the interpolations

\[
\varphi_h(X) = N_a(X)x_a, \quad \delta \varphi_h(X) = N_a(X)\delta x_a, \quad \dot{\varphi}_h(X) = N_a(X)\dot{x}_a, \quad (3.17)
\]

where \(N_a\) is the conventional shape function corresponding to node \(a \in [1, N]\), \(x_a\) is the vector of current nodal positions, and \(N\) is the number of nodes. The weak form (3.14) therefore reduces to the set of ordinary differential equations:

\[
M_{ab}\ddot{x}_b + \mathbf{f}^i_a(x) + \mathbf{f}^s_a(x) = \mathbf{f}_a^e, \quad \forall t \in T, \quad (3.18)
\]

where, the inertial, internal, interface and external forces are respectively defined by

\[
M_{ab}\ddot{x}_b = \int_{\partial t_{Boh}} \rho_0 N_a N_b dV \quad \ddot{x}_b, \quad (3.19)
\]

\[
\mathbf{f}^i_a = \int_{\partial t_{Boh}} \mathbf{P} : \nabla_0 N_a dV, \quad (3.20)
\]

\[
\mathbf{f}^s_a = \pm \int_{\partial t_{Boh}} (1 - \alpha) \left< \frac{\beta_s}{h_s} \right> \left[ [x_b] \otimes N^- \right] \cdot N^- N^S_a N^S_b dS
\]

\[
\pm \int_{\partial t_{Boh}} (1 - \alpha) \left[ \left< \frac{\beta_s}{h_s} \right> \left[ [x_b] \otimes N^- \right] \cdot N^- N^S_a N^S_b dS
\]

\[
\pm \int_{\partial t_{Boh}} \alpha T([x]) N^S_a dS, \quad (3.21)
\]

\[
\mathbf{f}^e_a = \int_{\partial t_{Boh}} \rho_0 \mathbf{B} N_a dV + \int_{\partial_{NBoh}} \mathbf{T} N^S_a dS, \quad (3.22)
\]
where \( M_{ab} \) is the mass matrix, and \( \pm \) refers to the boundaries of the two elements sharing the same interface.

The numerical implementation of the DG/cohesive framework is based on the use of interface elements introduced between the volumetric finite elements as depicted in Figure 2-3. The main advantage of using interface elements is the ability to integrate both the DG interface forces as well as the cohesive law (3.21). In this formulation, the conventional finite elements inside the volume of the domain can be used without modification. In this paper, 10-node quadratic tetrahedral elements are used, resulting in 12-node quadratic interface elements. Tetrahedral elements are integrated using a 4-point reduced quadrature rule, while the interface elements require full 6-point integration in order to prevent spurious penetration modes [69].

The interface elements are inserted between the two volume elements \( \Omega_0^+ \) and \( \Omega_0^- \) by splitting the shared nodes, leading to independent problem unknowns. This new element encompasses the surface elements \( \partial_1 \Omega_0^+ \) and \( \partial_1 \Omega_0^- \), which coincide in the reference configuration. In the reference configuration the interpolation of the position, the deformation mapping and its jumps are computed using the standard shape functions of the surface element \( N_a^S(\xi) \), \( a \in [1,n] \), where \( \xi = (\xi_1, \xi_2) \) are the natural coordinates. For example, the reference configuration of the element is thus described by the expression:

\[
X^\pm(\xi) = \sum_{a=1}^{n} N_a^S(\xi) X_{a^\pm},
\]

where \( X_{a^\pm}, a \in [1,6] \) are the nodal coordinates of the surface elements in the reference configuration. The reference interelement outer surface normal \( N^- \) corresponding to element \( \Omega_0^- \) evaluated on the middle surface is obtained from the expression:

\[
N^-(\xi) = \frac{G_1(\xi) \times G_2(\xi)}{|G_1(\xi) \times G_2(\xi)|},
\]

in which

\[
G_\alpha(\xi) = \langle X_{\alpha} \rangle = \sum_{a=1}^{n} N_{a,\alpha}^S(\xi) \langle X_{a} \rangle
\]

72
are the tangent basis vectors, \( \alpha \in [1, 2] \), \( \langle \mathbf{X}_a \rangle = \frac{\mathbf{x}_a^+ + \mathbf{x}_a^-}{2} \). To compute the equivalent geometric quantities in the deformed configuration, as required by the TSL, the same expressions can be used by simply replacing the undeformed nodal positions with those in the deformed configuration.

Time integration of the dynamics equations (3.18) is effected using a conventional second-order central-difference scheme [99] with mass lumping, which is suitable for the fast dynamic fracture processes of interest in this paper.

### 3.4 Parallel Implementation using MPI

The parallel implementation of the computational framework encompasses the following steps:

1. **Loading of a coarse, conforming finite element mesh into memory.**

   The full nodal coordinate and connectivity information describing an initial coarse and conforming finite element mesh are loaded from the file system on each processor and stored as one-dimensional arrays.

2. **Partitioning of the original conforming finite element mesh into as many processors as form part of the calculation.**

   The mesh is decomposed spatially into as many geometrically connected partitions as there are processors taking part in the calculation. Each processor retains only the portion of the coordinate and connectivity arrays corresponding to their assigned partition and discards the rest of the mesh from memory. The spatial decomposition is performed using the popular METIS library [100].

3. **Distributed mesh refinement based on element subdivision.**

   In this step, every element on each processor is subdivided into 8 subtetrahedron using the SUB8 algorithm from [101], reducing the size of each element by a factor of one half (see Figure 3-1). The element subdivision scheme is applied to the mesh as many times as necessary to reach the desired level of
mesh refinement. After each subdivision step, the nodal coordinate and element connectivity arrays are updated on each processor to reflect the addition of new nodes and elements in the mesh. This subdivision refinement algorithm allows for the use of exceedingly large meshes without the limitation of needing to load a large mesh on each processor prior to partitioning.

4. **Identification of the topology of the partitioned and refined mesh to determine neighbors of each partition and generation of the nodal communication maps with neighbors.**

Although these are later discarded after the DG mesh is created, they provide very useful information to identify partition boundaries in the DG mesh.

5. **Generation of the data structures of the new volumetric finite element mesh corresponding to the partition.**

In the DG formulation, volumetric elements have their own nodes which are not shared with adjacent elements. This can be done in a straightforward manner from the continuous partition finite element mesh by iterating through all the elements, replicating all the original nodes shared with the adjacent elements and updating the connectivity table to point to the new node identifiers. It is then clear that a node in the original mesh gets replicated as many times as its number of adjacent volume elements. A map from the new to the old node ID is created as each new node is generated for later convenience when generating the interface elements and the interprocessor communication maps.

6. **Generation of interface elements in the interior of the partition mesh.**

This can be efficiently accomplished in linear time on the total number of faces in the mesh by employing the WingedFace data structure [102] applied to the original partition mesh. For each face, this stores information about the adjacent volume elements and their face adjacency. Through the new volume element connectivity table one can then directly create the connectivity map for the interface elements. It should be noted that this step does not require
the insertion of any extra nodes in the mesh.

7. **Generation of interface elements at interprocessor boundaries.**

For faces on interprocessor boundaries, it is clear that one of the two neighboring processors needs to define an interface element. Real domain boundary is distinguished from interprocessor boundary by use of the old communication maps and the new-to-old nodal ID mapping as follows: once a partition boundary face is found, the face’s old node IDs are searched in the communication map of the partition neighbors. When all face nodes are found on a neighbor communication map, that signals that the face is an interprocessor boundary face and that a new interface element should be inserted. An interface element is inserted if the processor ID is lower than that of the neighbor to avoid duplication. This potentially creates a small imbalance but in practice no impact on scalability has been observed as the imbalance is only of a surface-to-volume character.

8. **Update of the interprocessor nodal communication maps.**

In the framework proposed, the new nodal communication maps are simple to obtain from the existing information of the interprocessor boundary faces. Remarkably, this step can be done without the need for any communications, barring two exceptional cases as described below. The number of nodes involved in the pairwise map is equal to the number of faces on that interprocessor boundary multiplied by the number of nodes per face (six in the case of 10-node tetrahedral elements used here). A simple spatial sort procedure of the faces involved and their nodes is used to match up the nodal communication pairs while avoiding communications. The spatial sorting needs to be done in two levels (for all the nodes of each face first and then for face node groups), as there are spatially-coincident nodes in adjacent faces which can lead to incorrect pairwise node identification across the communication map due to machine round-off. The two exceptions which require communication occur when either:
- **Case 1:** A partition boundary node is shared by more than one face of the same volumetric element.

The face-based communication map then leads to a duplication of a node on the communication map. Fortunately, these nodes can be simply identified with minimal communication and either eliminated from the communication map or handled in the interprocessor assembly operation. We have chosen this latter approach for implementation efficiency. This requires a modification of the assembly operator. The modification of the assembly operator is described in the following.

- **Case 2:** A partition boundary node is shared between three or more processors and one or more pairs of these processors do not share a common face.

In this case, the face-based communication maps are incomplete because nodes will be shared among three or more processors, but will only be included in maps between processors which share a face. Hence, additional communication maps need to be created between processors that share a node, but which do not share a common face. This special case is handled by first identifying the problematic shared boundary nodes within the highest ID processor through searching the existing communication maps, and then sending information to the lower ID processors instructing them on which nodes are shared and by what additional processors. The lower ID processors then create an additional communication map which accounts for the shared boundary nodes.

9. **Modified parallel assembly operator.**

The usual approach of assembling global nodal quantities (e.g. masses, internal forces, etc.) by first adding the local processor contributions to all the processor nodes and then completing boundary-node values through an MPI reduce operation using the communication maps needs to be extended to contemplate the case of repeated nodes in the face-based communication maps. The exten-
Figure 3-1: (a) A schematic of the element subdivision scheme utilized for mesh refinement. The original tetrahedron \((t_0,t_1,t_2,t_3)\) is subdivided into 8 subtetrahedron, \((t_0,t_01,t_02,t_03)\), \((t_01,t_1,t_12,t_13)\), \((t_0,t_03,t_13,t_23)\), \((t_2,t_02,t_12,t_23)\), \((t_01,t_02,t_03,t_13)\), \((t_01,t_02,t_12,t_13)\), \((t_02,t_03,t_13,t_23)\), and \((t_02,t_12,t_13,t_23)\).

It bears emphasis that the steps in the parallel implementation are independent of the specific finite and interface element type employed. The outlined procedure is then equally applicable to higher-order or hexahedral elements, etc. Additional aspects of the parallel DG implementation may be found in [69]. However, it should be noted that the steps outlined above represent a significant improvement compared to the approach in this reference as it avoids the need of using and maintaining a global node ID, which, in turn, requires significant communication and spatial searches in the initialization stage of the overall algorithm. Once the new communication maps and modified assembly operator are established, the calculation proceeds in the same fashion as the well-established and widely-adopted approach of MPI-based explicit finite element calculations [103, 104].
3.5 Large-Scale Scalability Testing and Performance Analysis

In this section, we report the results of an extensive suite of testing to assess the parallel scalability, communication overhead and node performance of the DG/CZM implementation in the MIT Summit code. The testing was carried out on several different high performance computing platforms operated by Army Research Laboratory (ARL) and the Engineering Research and Development Center (ERDC) through the DoD Supercomputing Resource Center (DSRC). A small set of scalability runs was also completed on our group cluster at MIT. The platforms on which the testing was conducted are described below:

**MIT**

- Our group cluster which consists of 40 compute nodes (320 compute cores). Each compute node contains two Intel 2.26 GHz Xeon E5520 64-bit quad-core processors with 24Gb of memory. The nodes are interconnected via a 4x DDR Infiniband network. For the scalability tests, OpenMPI was used to build and run the code.

**ERDC DSRC**

- The **Diamond**\(^1\) system which is an SGI Altix ICE with an ideal peak performance of 172 TFLOPS, consisting of 1,920 compute nodes (15,360 compute cores). Each compute node contains two 2.8GHz Intel Xeon 64bit quadcore Nehalem processors with 24 Gb of memory. The nodes are interconnected in a HyperCube topology DDR 4X Infiniband network. For the scalability tests, SGI-MPT 1.26 was used to build and run the code.

**ARL DSRC**

- The **MJM**\(^2\) system which consists of 1100 compute nodes (4400 cores). Each

---


\(^2\)Since this testing was conducted, the MJM system has been decommissioned and is no longer in use.
compute node contains two Intel 3.0 GHz Woodcrest 64-bit dual-core processors with 8 Gb of memory each. The nodes are interconnected via a 4x DDR Infiniband network. For the scalability tests, the OpenMPI was used to build and run the code.

- The Pershing\(^3\) system which is an IBM iDataPlex with an ideal peak performance of 420 TFLOPS, consisting of 1,260 compute nodes (20,160 compute cores). Each compute node contains two 2.6-GHz Intel 8-core Sandy Bridge processors with 32 Gb of memory. The nodes are interconnected by an FDR-10 Infiniband network. For the FLOP and scalability testing, Intel-MPI 4.1.3 was used to build and run the code.

- The Hercules\(^4\) system which is an IBM iDataPlex with an ideal peak performance of 320 TFLOPS, consisting of 1,079 compute nodes (17,264 compute cores). Each compute node contains two 2.6-GHz Intel 8-core Xeon Sandy Bridge processors with 64 Gb of memory. The nodes are interconnected by an FDR-10 Infiniband network. For the FLOP and scalability testing, Intel-MPI 4.1.3 was used to build and run the code.

**Weak Scalability Tests**

In a weak scalability test (also known as a "scaled speed-up" test), both the problem size and the number of cores are increased proportionally so as to maintain a constant computational load per processor. If the weak scalability of the code is perfect, then the time to compute will not change as the problem size and processor count is increased. First in this section, we present the results of initial weak scaling tests reported in [4], which were conducted two years ago on the previous generation of supercomputing hardware. The platforms utilized for this initial round of weak scaling tests were the MIT, ARL MJM and ERDC Diamond systems described above. The particular boundary value problem considered is the impact of a rigid sphere on a


\(^4\)More information about Hercules is unavailable at this time because it has recently been transitioned to a classified system after our testing was completed
monolithic, elastic plate (a problem investigated at length in Chapters 5 and 6). A fixed computational load of 25,366 volume elements (corresponding to 760,980 degrees of freedom) per core was employed and the largest simulation was conducted on 4096 processors on the Diamond system with a total of 103 million elements (corresponding to 3 billion degrees of freedom). The results are summarized in Figure 3-2, which shows a plot of the CPU time per time step as a function of the number of cores used for each platform. The plot shows that the DG/CZM method maintains excellent weak scalability on the MIT system. On the MJM and Diamond systems, weak scalability is maintained up to some threshold processor count, at which point the computation time starts to increase. This can be attributed to initial load imbalance in the mesh partitioning which is exacerbated exponentially as the processor count and problem size is increased.

Although these initial weak scaling results were promising, it remained to be seen whether the code retained its high scalability for even larger problem sizes and processor counts using the next generation of computer clusters and MPI software. Recently, two new systems (called "Pershing" and "Hercules", summarized above) were released for an initial "Pioneer" testing phase in which a small group of users were provided with unallocated access to the machines for a short period of time. As part of the Pioneer test group for the ARL, we were able to complete a new set of weak scaling runs on Hercules and Pershing. On the Hercules system, a fixed computational load of 52,285 volume elements (corresponding to 1,568,550 degrees of freedom) per core was considered. The largest simulation conducted on Hercules used 12,288 processors and a total mesh size of 642 million volume elements, or 19.3 billion degrees of freedom. On the Pershing system, a fixed computational load of 19,607 elements per core was considered. The largest simulation on Pershing used 1,536 processors and a total mesh size of 10 million tetrahedral elements, or 300 million degrees of freedom. In Figure 3-3, we show the results of weak scalability tests on the Pershing and Hercules systems. In contrast to the previous study, the code retains almost perfect weak scalability on both platforms up to the largest problem and size and processor count considered. In future work, it remains to be seen whether such
Figure 3-2: Initial demonstration of weak scalability of the DG/CZM method on up to 4096 processors and problems of upward of 3 billion degrees of freedom: Scaled speed-up results given as compute time per time step as a function of number of processors for all three platforms tested.
scalability is retained on the next generation of "superclusters" which feature millions of compute cores.

Figure 3-3: Recent demonstration of weak scalability of the DG/CZM method on new ARL systems Hercules and Pershing. The plot shows simulation time per time step as a function of the number of cores used.

**Strong Scalability Tests**

In a strong scalability test, the total problem size is held fixed and the number of cores is increased. If the code retains perfect strong scalability, then as the number of cores is increased, the computation time should decrease proportionally to the increase in processor count. During the Pioneer testing phase of the Hercules system, we were able to complete an extensive set of strong scalability test runs. Two mesh sizes were considered in the tests, consisting of 10,038,784 and 80,310,272 tetrahedral elements, respectively (corresponding to 300 million and 2.4 billion degrees of freedom). Owing the small number of Pioneer users testing the machine, we were able to complete strong scalability runs using up to the full machine (17,264 processors). The test results for strong scalability are provided in Figure 3-4. Similar to the weak scaling
results, we find that the code retains almost perfect strong scalability for the two different problem sizes up to the largest processor count considered.

Figure 3-4: Recent demonstration of strong scalability on up to 17,264 processors for two different problem sizes on new ARL system Hercules. The plot shows simulation time per time step as a function of the number of cores used.

Analysis of Communication Efficiency
In order to illustrate the efficiency of the parallel scheme and its low communication overhead, we conducted a series of impact damage simulations on ARL Pershing in which we measured the relative contributions of communication and computation to the overall simulation time. The total communication time consisted of corresponding MPI_Send and MPI_Recv operations between neighboring processors in which information was exchanged for their shared boundary nodes. The timing data was obtained on each processor using MPI_Wtime() and then overall values were determined using an MPI_Allreduce(). In Figure 3-5 we show the relative costs of communication and computation for three different problem sizes using increasing processor counts. We find that the relative communication overhead is dwarfed by the time spent for frac:
ture mechanics computation inside of each processor. Indeed, for all three problem sizes, the relative cost of communication is no more than a few percent of the overall time for processor counts up to the point at which the computational load becomes less than 10,000 elements per processor. When the mesh size per processor is reduced beyond this limit, the relative cost of communication increases to approximately 20% of the overall simulation time.

![Graph showing the relative costs of communication versus computation for impact damage simulations using different numbers of processors for three mesh sizes.](image)

Figure 3-5: Analysis of the relative costs of communication versus computation for impact damage simulations using different numbers of processors for three mesh sizes.

From this result it becomes clear that the superior scalability of the DG/CZM framework is enabled by the method’s extremely low communication overhead. This efficiency can be explained by the automated and highly localized nature of interprocessor communication. Since only stress, stiffness, and residual information is exchanged between processors, there is no need to communicate the location of cracks as they cross processor boundaries. Therefore, arbitrary numbers of cracks can propagate in the mesh, impasive to the presence of processor boundaries, without incurring any additional communication overhead. Furthermore, since information is only exchanged for nodes occupying the boundary between neighboring processors in
a spatial decomposition of the mesh, the relative cost of communication to computation scales with the ratio of the number of surface to volume nodes in each mesh partition. This guarantees that the code will retain excellent scaling up to arbitrarily large processor counts as long as the size of each mesh partition is sufficiently large.

Node Performance Analysis

As part of the Poineer testing phase on the Pershing and Hercules systems, we collaborated with ARL computational scientists to conduct a detailed analysis of the floating point operations per second (FLOPS) achieved by the DG/CZM method for different problem sizes and processor counts. To complete the FLOP profiling, we used the Performance Application Programming Interface (PAPI) package which provides a suite of utilities for collecting raw performance data from hardware counters. Using function calls to the PAPI library directly in our application driver, we collected data on floating point operations completed and computation time, from which we were able to extract the FLOPS achieved by our code. The data collected during the testing period proved valuable for the ARL team in assessing the performance characteristics of the Pershing and Hercules systems.

A comprehensive analysis of all the FLOPS data collected during the Pioneer phase is too extensive to be reported in this manuscript. In order to give some idea of the FLOPS performance of the DG/CZM method in a large-scale simulation, instead we present the data collected for one full machine run (17,264 processors), using a finite element mesh consisting of 642,482,176 tetrahedral elements (19.3 billion degrees of freedom). The simulation was run for 100 time steps and the FLOPS was measured on each processor for each computational step. In Figure 3-6 we show the minimum, maximum, and average FLOPS measured on each processor during the 100 step run. A maximum performance of approximately 700 megaFLOPS was achieved on some processors while on the average, approximately 650 megaFLOPS was achieved. The small spread in the measured FLOPS across the different processors shows that the mesh partitioning resulted in good overall load balancing. Summing the FLOPS over

5More information available at http://icl.cs.utk.edu/papi/
all the cores at each step provides an estimate of the total overall FLOPS achieved over the course of the simulation. Doing so, we measured a peak total performance of 11.4 teraFLOPS during the simulation which is approximately 3.6% of the ideal peak machine performance. Improving these performance numbers through software and hardware optimization (e.g. using hybrid CPU/GPU nodal configurations and software optimized with petaBricks\textsuperscript{6}) is an important topic for future research.

![Graph showing Measured FLOPS per processor for full machine run on ARL Hercules](image)

**Figure 3-6:** Measured FLOPS performance for each processor in a full machine run on new ARL system Hercules. The plot shows the measured minimum, maximum, and average FLOPS for a run of 100 timesteps.

In summary, from the extensive scalability, communication efficiency, and node performance testing presented in this section, it can be concluded that the proposed algorithm and its implementation in the MIT Summit code enables extreme scale

\textsuperscript{6}More information available at [http://projects.csail.mit.edu/petabricks/](http://projects.csail.mit.edu/petabricks/)
simulation of dynamic fracture.
Chapter 4

Validation Studies

In this chapter, we investigate the ability of the DG/CZM method to capture experimentally observed fracture patterns in brittle materials subjected to projectile impact loadings. In sections 4.1.1 and 4.1.2 we simulate PMMA and ceramic edge-on impact (EOI) experiments from the literature. For the EOI tests, the simulation results are compared qualitatively to the cracking patterns observed in experiments and quantitative comparisons are also made using several metrics including crack propagation lengths over time and crack tip speeds. Following this initial example, in section 4.2 we present the results of a preliminary exploratory simulation in which we assess the ability of the DG/CZM code to capture fracture patterns in ceramic tiles subjected to normal impact loading.

4.1 Edge-On Impact Simulations

The edge-on impact test was first developed by Schardin [105, 106] in the 1930s, who used the technique to investigate fracture patterns in glass. The technique was revived years later by Hornemann et al [107] and has since been employed extensively by Rosakis and co-workers to investigate interfacial crack growth in bimaterials [108, 109]. One of the advantages of the EOI test is that it can be used in conjunction with Coherent Gradient Sensing (CGS) techniques [110] to extract detailed quantitative information about the crack propagation including the evolution of the near-tip...
displacement gradients and the crack tip speed (see e.g. the work of Singh et. al [109]).

4.1.1 Edge-On Impact of PMMA

Umberger and Love conducted an extensive set of EOI tests in which they used a gas gun to accelerate an aluminum cylinder against a pre-notched PMMA plate in a one-point bend configuration (see Figure 4-1). The impact velocity was tailored in the tests to induce the propagation of a single Mode I crack from the notch tip and the evolution of crack tip speed and surface displacement gradients were measured using CGS. In most cases, the authors observed that the crack accelerated quickly to a steady state speed before either branching, arresting, or traversing the entire specimen. For increasing projectile velocity, Umberger and Love found that the steady state crack tip speed increased overall, although there was significant statistical spread in the experimentally measured values. The experimental crack tip speeds are shown in Figure 4-3.

In this section, the DG/CZM method is applied to model the EOI tests of Umberger and Love. For simplicity, the cylindrical aluminum projectile is modeled as a rigid surface with the same dimensions and the same mass as that used in [111]. Contact forces are applied between the projectile and the target plate using the penalty contact algorithm described in Appendix A. In the simulations, the bulk material behavior in the PMMA plate is assumed to be elastic. To model fracture, we first employ the rate-independent cohesive law presented in detail in Section 3.2 and we adopt the parameter values used by Zhou and co-workers [112] for simulation of Mode I crack propagation in a thin PMMA strip. The PMMA plates considered in the simulation study, which have the same dimensions as those tested by Umberger and Love, were meshed in ABAQUS with an advancing front algorithm. Using uniform element sizes of roughly 600 \( \mu m \), the meshes produced by ABAQUS contained between 270,000 and 550,000 volumetric tetrahedra, depending on the thickness of the specimen. The problem parameters and material properties utilized in the simulations are provided in Table 4.1.1.
Figure 4-1: A schematic of the specimen and impactor geometry from the Mode I EOI tests of Umberger and Love on pre-notched PMMA plates [111]

**Rate-Independent CZM**

Using the rate-independent cohesive model of Ortiz and Pandolfi, we simulated the impact experiments of Umberger and Love on pre-notched PMMA plates for specimen thicknesses of 6 and 13mm and impact velocities between 20 and 60 m/s. To illustrate some of the basic physics captured by the simulations, we provide visualization and analysis of the simulation results corresponding to a 6mm tile thickness and an impact velocity of 30 m/s. In Figures 4-2(a)–4-2(f) we provide a series of snapshots which show the evolution of the $\sigma_{yy}$ stress in the sample over the course of the simulation. In Figure (4-2(a)) we see the incident longitudinal stress wave which has a compressive $\sigma_{yy}$ component at its wavefront. There is also a tensile component of the $\sigma_{yy}$ stress in the wake of the longitudinal wave along the impact axis which corresponds to the propagating shear wave. Figure (4-2(b)) shows the $\sigma_{yy}$ field just after reflection of the longitudinal wavefront from the boundary opposite of the impacting cylinder,
Table 4.1: Problem Parameters and Material Properties for simulations of the EOI tests of Umberger and Love [111].

<table>
<thead>
<tr>
<th>Properties</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Plate Geometry and Material Properties</strong></td>
<td></td>
</tr>
<tr>
<td>Length</td>
<td>$L = 15.54$ cm</td>
</tr>
<tr>
<td>Width</td>
<td>$w = 10.16$ cm</td>
</tr>
<tr>
<td>Thickness</td>
<td>$t = 0.6$ and $1.3$ cm</td>
</tr>
<tr>
<td>Notch Length</td>
<td>$1.905$ cm</td>
</tr>
<tr>
<td>Density</td>
<td>$\rho_0 = 1180$ kg$\cdot$m$^{-3}$</td>
</tr>
<tr>
<td>Elastic Modulus</td>
<td>$E = 3$ GPa</td>
</tr>
<tr>
<td>Poisson's Ratio</td>
<td>$\nu = 0.35$</td>
</tr>
<tr>
<td>DG Stability Parameter</td>
<td>$\beta_s = 4.0$</td>
</tr>
<tr>
<td>Critical Cohesive Strength</td>
<td>$\sigma_c = 75.0$ MPa</td>
</tr>
<tr>
<td>Fracture Energy</td>
<td>$G_c = 300$ J$\cdot$m$^{-2}$</td>
</tr>
<tr>
<td>Tension/Shear Weighting Parameter</td>
<td>$\gamma = 1.0$</td>
</tr>
<tr>
<td>Interface Viscosity Parameter</td>
<td>$\upsilon = 0$ MPa$\cdot$s$\cdot$m$^{-1}$</td>
</tr>
<tr>
<td><strong>Projectile Geometry and Material Properties</strong></td>
<td></td>
</tr>
<tr>
<td>Radius</td>
<td>$R_c = 0.95$ cm</td>
</tr>
<tr>
<td>Mass</td>
<td>$m_c = 67$ g</td>
</tr>
<tr>
<td>Initial Velocity</td>
<td>$V_c = 20 - 60$ m$\cdot$s$^{-1}$</td>
</tr>
</tbody>
</table>

with the slower moving shear wave propagating some distance behind. The reflected longitudinal wave then interacts with the oncoming shear wave in Figure (4-2(c)) which induces a predominantly tensile $\sigma_{yy}$ field in the sample, with a localized stress concentration at the tip of the notch. The symmetric and localized $\sigma_{yy}$ field at the crack tip eventually induces the propagation of a single Mode I crack from the notch tip which is seen in Figure (4-2(d)) which eventually branches into two cracks as seen in Figure (4-2(e)). Both upper and lower branches continue to propagate and eventually the upper branch undergoes secondary branching event as seen in Figure (4-2(f)).

In order to quantitatively compare the results of simulations directly to experiments, an algorithm was implemented into the DG/CZM code for tracking the crack tip location over time during the simulations. The crack tip tracking algorithm is described in detail in Appendix B. In Figure (4-3) we compare the crack tip speeds extracted from simulations directly to the experimentally observed crack tip speeds reported by Umberger and Love in [111]. We find that the DG/CZM framework is
Figure 4-2: Snapshots showing crack propagation and the evolution of $\sigma_{yy}$ for simulations of the EOI tests on pre-notched PMMA plates from [111]
Figure 4-3: A comparison of the steady state crack tip speeds extracted from simulations with the experimentally observed values from the EOI experiments of Umberger and Love on pre-notched PMMA plates [111]. Results are shown for simulations using both rate-independent and rate-dependent CZM.

able to capture the steady state crack tip speeds reasonably well at lower impact velocities but generally over-estimates the crack tip speeds at higher impact velocities using the rate-independent cohesive formulation.

Rate-Dependent CZM

In previous work, Zhou and co-workers have shown for simulations of Mode I crack propagation in PMMA using the rate-independent extrinsic cohesive law employed here, that crack tip speeds predicted in simulations were larger than experimentally measured values [112]. The authors suggested that the discrepancy between the simulated crack tip speeds and the experimentally measured values can possibly be attributed to the fact that the rate-independent cohesive law assumes a value of the fracture energy which is constant irrespective of the dynamic conditions of crack
propagation. This assumption is in conflict with experimental observations that for most brittle materials, and especially PMMA, the fracture energy is a monotonically increasing function of the crack tip speed [113, 114]. In order to develop a CZM which captures the dependence of the fracture energy on the crack tip speed, Zhou and co-workers extended the cohesive law of Ortiz and Pandolfi [48] to include a dependence of the fracture energy parameter on the local rate of opening across the cohesive zone [112]. With this formulation, the fracture energy depends indirectly on the crack tip speed and can, in principle, be calibrated to experimental results to capture the dependence of the fracture energy on crack tip speed.

In order to investigate the effects of rate-dependence of the fracture energy on the predicted crack propagation behavior for simulations of the Mode I EOI tests, we implemented the rate-dependent model of Zhou and co-workers into the DG/CZM code. The model is based on a cohesive free energy density function, \( \phi \), which depends on both an effective opening, \( J \) (defined by Equation 2.33), and its time derivative. Hence, the rate-dependent free energy function has the following form:

\[
\phi = \phi \left( \delta, \dot{\delta} \right) \tag{4.1}
\]

This leads to the definition of an effective traction which now depends on both the effective opening and its time derivative, i.e. \( T = T \left( \delta, \dot{\delta} \right) \). The authors then assume a particular functional form for the effective traction which is identical to the form chosen by Ortiz and Pandolfi [48] with the exception that the critical opening parameter is no longer a constant, but rather depends on the effective opening rate (i.e. \( \delta_c = \delta_c \left( \dot{\delta} \right) \)). Hence \( T \) is given by:

\[
T \left( \delta, \dot{\delta} \right) = \sigma_c \left( 1 - \frac{\delta}{\delta_c \left( \dot{\delta} \right)} \right) \tag{4.2}
\]

where \( \sigma_c \) is the critical cohesive strength and \( \delta_c \left( \dot{\delta} \right) \) is the critical opening function.
which has the following form:

$$\delta_c(\dot{\delta}) = \delta_0 \left( 1 + \left( \frac{\dot{\delta}}{\dot{\delta}_0} \right)^n \right).$$

(4.3)

In this expression, $\delta_0$ is the value of the critical opening for a stationary crack and $\dot{\delta}_0$ and $n$ are fitting parameters which control the effects of rate-dependency in the fracture response. The reasoning behind this particular formulation is that it naturally captures the effect of rate-dependence in the fracture energy through the relationship between the work of separation and the critical opening corresponding to the definition of the effective traction in Equation 4.2:

$$\phi_s = \frac{1}{2} \sigma_c \delta_c(\dot{\delta}) = \frac{1}{2} \sigma_c \delta_0 \left( 1 + \left( \frac{\dot{\delta}}{\dot{\delta}_0} \right)^n \right).$$

(4.4)

For the simulation study with the rate-dependent model we use same the problem parameters as given in Table 4.1.1, and for the additional rate-dependency parameters we choose the values suggested by Zhou and co-workers in [112] (i.e. $n = 1.0$, $\dot{\delta}_0 = 5.0$ m/s) for which the authors obtained good agreement between simulated and experimental crack tip speeds for Mode I crack propagation in a thin PMMA strip. Using the same mesh as before for the simulations with the rate-independent model, we simulated the Mode I EOI tests for different impact velocities with the rate-dependent model and extracted the steady state crack tip speeds from simulations using the crack tip tracking algorithm discussed in Appendix B. In Figure 4-3 we compare the steady state crack tip speeds from the rate-dependent simulations with the values from rate-independent simulations and experiments. We find for all impact velocities, that the steady state crack tip speeds obtained with the rate-dependent model are lower than the values observed in simulations with the rate-independent model and provide a better fit to the experimental results for higher impact velocities. This result confirms the intuitive notion that the crack-tip velocity dependence of the dynamic fracture energy has a strong effect on the crack tip dynamics and suggests, as was shown by Molinari and co-workers, that the rate-dependent cohesive model
provides a better representation of the dynamic fracture physics since it captures this effect.

4.1.2 Edge-On Impact of Ceramics

In [115], Strassburger and Senf conducted an extensive experimental study of dynamic fracture in thin glass and ceramic plates subjected to edge-on impact by steel cylinders. In the experiments, high speed video was used to visualize the evolution of cracking patterns on the plate surfaces. The high speed video images revealed that for ceramics, the key failure mechanisms are the propagation of cone cracks from the edges of the contact zone and the development of an expanding damage front composed of many cracks nucleating and propagating along planes directed outwardly in the radial direction. Experiments conducted at different impact velocities showed that the overall dynamic fracture pattern depends strongly on the loading rate. Visualization of the experimental cracking patterns also provided a basis for extracting quantitative information such as the average propagation distance of the radial fracture front over time.

In this section, we simulate the edge-on impact experiments from [115] conducted using Al₂O₃ tiles at different impact velocities. In order to validate the computational approach, the simulation results are compared directly to the experiments. The EOI configuration considered in the simulations is depicted in Figure 4-4. For simplicity, the cylinder is treated as a rigid body in the simulations and contact between the cylinder and the plate is modeled using a simple penalty contact formulation. The finite element mesh, obtained with Gmsh software [116], was optimized to eliminate structure that might bias the predicted crack paths [50]. The mesh containing 266,000 volumetric tetrahedra, or a total of 7.98 million degrees of freedom, is shown in Figure 4-4. In the simulations, test velocities corresponding to 85 and 150 m·s⁻¹ are considered. In this relatively low velocity regime, the effects of plasticity can be neglected as the impact velocities are below free surface velocities recorded at the Hugoniot elastic limit in flyer plate experiments on Al₂O₃ [117]. As a result, the bulk material is assumed to behave elastically up until fracture. The material properties
Figure 4-4: The initial configuration for simulations of the edge-on impact tests of Strassburger and Senf [115]. The figure shows the finite element discretization used in the simulations, consisting of 266,000 volumetric elements.

and problem parameters used in simulations are provided in Table 4.2.

In Figures 4-5(a)-4-5(d), the computed evolution of the crack patterns for $V_c = 85 \text{ m-s}^{-1}$ is compared to high speed video images of the experimental observations. In the early stages of the impact event, cone cracks nucleate and propagate into the specimen from the upper and lower edges of the contact zone boundary accompanied by crack nucleation some distance away from the contact zone along the centerline of the impactor. Eventually, cracks nucleate away from the centerline of the impactor which propagate along radial planes, creating a fan-shaped damage front in the specimen. Cracks also form in localized regions on the upper and lower edges of the specimen. The images in Figures 4-5(a)-4-5(d) show a good qualitative agreement between simulations and experiments for both the spatial and temporal occurrence of crack nucleation and the subsequent crack propagation paths.

Quantitative information describing the crack propagation from simulations for the lower impact velocity was obtained by measuring the furthest extent of the radial damage front along the centerline of the specimen and the length of the upper cone crack over time. The results were compared to the average damage front location and cone crack propagation length reported by Strassburger [115], Figure 4-6. As the plot shows, the simulated cone crack length and the damage front location are
### Properties Values

<table>
<thead>
<tr>
<th>Properties</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Al$_2$O$_3$ Plate</strong></td>
<td></td>
</tr>
<tr>
<td>Length</td>
<td>$L = 10$ cm</td>
</tr>
<tr>
<td>Thickness</td>
<td>$t = 1$ cm</td>
</tr>
<tr>
<td>Initial density</td>
<td>$\rho_0 = 3690$ kg·m$^{-3}$</td>
</tr>
<tr>
<td>Elastic Modulus</td>
<td>$E = 370$ GPa</td>
</tr>
<tr>
<td>Poisson's Ratio</td>
<td>$\nu = 0.22$</td>
</tr>
<tr>
<td>DG Stability Parameter</td>
<td>$\beta_s = 4$</td>
</tr>
<tr>
<td>Cohesive Strength</td>
<td>$\sigma_c = 380$ MPa</td>
</tr>
<tr>
<td>Fracture Energy</td>
<td>$G_c = 40$ J·m$^{-2}$</td>
</tr>
<tr>
<td>Tension/Shear Weighting</td>
<td>$\gamma = 5$</td>
</tr>
<tr>
<td>Interface Viscosity Parameter</td>
<td>$c_v = 10^{-2}$ MPa·s·m$^{-1}$</td>
</tr>
<tr>
<td><strong>Rigid Cylinder</strong></td>
<td></td>
</tr>
<tr>
<td>Radius</td>
<td>$R_c = 1.5$ cm</td>
</tr>
<tr>
<td>Mass</td>
<td>$m_c = 0.126$ kg</td>
</tr>
<tr>
<td>Velocity</td>
<td>$V_c = 85, 150$ m·s$^{-1}$</td>
</tr>
</tbody>
</table>

Table 4.2: Plate and projectile dimensions and material properties used for simulations of the EOI experiments of Strassburger and Senf [115].

in reasonable quantitative agreement with the experiments. However, the average speeds of propagation for the cone crack and the damage front were 5045 and 6829 m·s$^{-1}$, respectively, which are higher than the experimentally reported values of 3860 and 5320 m·s$^{-1}$. This can be attributed to uncertainty in material parameters and in the actual locations of the crack tips.

Figures 4-7(a)-4-7(e) compare the results of simulations and experiments for a cylinder impact at 150 m·s$^{-1}$. Similarly to the lower impact velocity case, the simulations capture the propagation of a complex front of cracks traveling in radial directions away from the impact point. However, while the fracture front at 85 m·s$^{-1}$ features the propagation of just a few dominant cracks, at 150 m·s$^{-1}$ many more cracks are nucleated. Notably, a large number of short, closely-spaced cracks nucleate in between longer, more dominant cracks with much more extended cracking off the impact axis in proximity to the specimen boundaries. The simulations also capture cracks appearing on the opposing edge of the specimen, Figures 4-7(d) and 4-7(e).

As in the case of the lower velocity, the extent of the damage front was measured
Figure 4-5: Comparison of simulated (left) and experimental (right) crack propagation patterns for 85 m·s⁻¹ impact along the centerline of the specimen and compared to the experimentally reported values, Figure 4-8. Overall, the simulated and experimental damage front locations are in reasonable agreement considering the uncertainty in the material parameters used in the simulations and in the experimental measurements. However, the average speed of the simulated damage front was 7600 m·s⁻¹ which is significantly higher than the average experimental speed of 5635 m·s⁻¹ reported in [115].

Using shadowgraph analysis, it has been shown that shear waves drive the nucleation and propagation of damage fronts in EOI experiments on glass plates [115]. In this regard, it is interesting to use the full field description provided by the simulations to confirm and further refine explanations for crack pattern formation based on experimental results. To this end, in Figures 4-9(a)–4-9(d) we examine the evolution of the maximum principal stress and its direction from the simulations.

In the early stage of damage formation (Figures 4-9(a) and 4-9(b)) the incident longitudinal wave propagates first from the impact site, forming a strong compressive front. In its wake, a complex region of tensile stress forms due to the interaction of shear waves originating from the upper and lower edges of the contact zone and from
Figure 4-6: Comparison of simulated and experimental crack propagation lengths over time for the cone crack and the damage front for an impact speed of 85 m·s⁻¹
Figure 4-7: Comparison of simulated (left) and experimental (right) crack propagation patterns for 150 m·s⁻¹ impact.
Figure 4-8: Comparison of the average radial location of the damage front over time from simulations and experiments for an impact velocity of 150 m·s$^{-1}$

the reflection of the leading longitudinal wave at the lateral specimen boundaries. A key feature of the shear waves revealed by the simulation results is that their propagation and subsequent interaction leads to strong tensile fronts with semi-circular shapes (see Figures 4-9(b) and 4-9(c)). As these fronts propagate along the plate, they expand radially and the maximum tensile stress aligns at each point approximately with the front tangent. Since a traction-based fracture criterion is utilized in the simulations, cracks nucleate in radial planes perpendicular to the maximum principal stress direction. Thus, the simulations confirm that the formation of radial damage fronts in EOI experiments is controlled by the evolution of the maximum principal stress resulting from the propagation of shear waves. In Figures 4-9(b), 4-9(c), and 4-9(d) the simulations also show that when the shear waves reach the upper and lower specimen edges, the maximum principal stresses are aligned parallel to the edges. This causes the localized nucleation of cracks at the boundaries which propagate nearly perpendicular to the specimen edges.

Comparing the response predicted at 85 and 150 m·s$^{-1}$, we find that the evolution
of the maximum principal stress distribution and its direction are similar. However, one key difference is that tensile stresses are substantially stronger at locations away from the centerline of the impactor for the higher velocity case. This drives the nucleation of more cracks in the expanding radial damage front, many of which do not propagate far before arresting, which results in a profusion of short cracks. This crack arrest can be explained by the larger crack density which leads to a fast release of the driving force for crack growth. This explains why as the impact velocity increases there is a transition from the propagation of a few dominant cracks to the propagation of many shorter and more closely-spaced cracks.

4.2 Normal Impact On Thick Ceramic Tiles

In this section, we apply the numerical framework in a preliminary investigation of the main problem of interest in this thesis: Normal impact of high velocity projectiles on ceramic tiles. The ceramic material that we choose to model in this study is Al$_2$O$_3$, one of the most commonly used materials in the design of armor systems for vehicles [5]. Because of its wide use in armor applications, there has been significant interest in better understanding the fracture mechanisms in impacted alumina tiles through experimental studies of normal impact [14, 16, 17, 23, 24, 22]. These studies have shown that for impact velocities of 1 km/s or lower, the dominant failure modes observed in experiments are conical and radial cracks. The damage response of alumina tiles subjected to normal projectile impact has also been investigated through 2D axisymmetric simulations using cohesive elements [3] and continuum damage models [22]. In both of these works, numerical simulations were able to capture the nucleation and propagation of conical cracks, but due to the axisymmetric restriction, radial cracks could not be modeled. This underscores the need for a computational analysis that can capture not only axisymmetric damage features, but also the failure mechanisms which are inherently three-dimensional.

The study conducted in this section will assess the ability of the DG/CZM method to capture three-dimensional aspects of crack propagation in impacted ceramic tiles.
Figure 4-9: Evolution of the simulated maximum principal stress and associated principal direction. For each time snapshot, the results for the lower impact velocity ($85 \text{ m}\cdot\text{s}^{-1}$), shown on the left of the figure, are compared to those for the higher impact velocity ($150 \text{ m}\cdot\text{s}^{-1}$), shown on the right of the figure.
that have not yet been demonstrated previously in 3D simulations using cohesive elements. The particular tile geometry chosen for this study corresponds to a thick ceramic specimen and the projectile is chosen to be a sphere. The first impact velocity considered is 500 m/s which is approaching the "high" velocity range used in ballistic impact tests [16, 17]. At this speed, it is possible for the projectile to experience a large shape change due to localized plastic flow, especially for steel or metal projectiles [118, 119, 24]. In finite element simulations of impact where the projectile undergoes large plastic deformations, there is usually a need to re-mesh the projectile adaptively during the simulation in order to avoid issues of mesh distortion [120]. However, it is well-known that adaptive remeshing algorithms suffer from robustness issues and are not easily parallelized, especially in three dimensions [120]. As a result of these issues, modeling the complex response of the projectile in an impact problem is a challenging research topic in its own right. Since our main objective is to investigate fracture patterns in the brittle ceramic tile, we neglect deformation of the projectile and treat it instead as a rigid surface. This greatly simplifies the complexity of the finite element model since both re-meshing of the projectile and enforcement of contact between two arbitrarily deforming bodies are completely avoided. It also significantly reduces the computational cost since the motion of the projectile can be obtained analytically at each integration step without the need for a costly numerical solution. Given the assumption that the projectile remains rigid, the contact forces are applied between the projectile and the ceramic tile the penalty contact algorithm described in Appendix A.

For simplicity, the ceramic is assumed to behave elastically up until fracture, which is assumed to be governed by the cohesive law described in Section 3.2. The material properties and problem parameters are given in Table 4.2.

The results of this simulation are shown in the sequence of Figures 4-10-4-15 and in Figure 4-16. Each Figure of the sequence 4-10-4-15 shows snapshots of the simulation at different times consisting of a montage of: (a) hydrostatic stress contours on the impacted surface of the plate, (b) instantaneous maximum principal stress contours on the back face of the plate, (c) contours of the instantaneous maximum stress.
<table>
<thead>
<tr>
<th>Properties</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Plate Geometry and Material Properties</strong></td>
<td></td>
</tr>
<tr>
<td>Length</td>
<td>$L = 5.8$ cm</td>
</tr>
<tr>
<td>Thickness</td>
<td>$t = 0.12$ cm</td>
</tr>
<tr>
<td>Density</td>
<td>$3700$ kg·m$^{-3}$</td>
</tr>
<tr>
<td>Elastic Modulus</td>
<td>$400$ GPa</td>
</tr>
<tr>
<td>Poisson's Ratio</td>
<td>$0.22$</td>
</tr>
<tr>
<td>DG Stability Parameter</td>
<td>$\beta_s = 4.0$</td>
</tr>
<tr>
<td>Cohesive Strength</td>
<td>$\sigma_c = 1.4$ GPa</td>
</tr>
<tr>
<td>Fracture Energy</td>
<td>$G_c = 25$ J·m$^{-2}$</td>
</tr>
<tr>
<td>Tension/Shear Weighting Parameter</td>
<td>$\gamma = 1.0$</td>
</tr>
<tr>
<td>Interface Viscosity Parameter</td>
<td>$c_v = 0$ MPa·s·m$^{-1}$</td>
</tr>
<tr>
<td><strong>Projectile Geometry and Material Properties</strong></td>
<td></td>
</tr>
<tr>
<td>Radius</td>
<td>$R_s = 3.7$ mm</td>
</tr>
<tr>
<td>Mass</td>
<td>$m_s = 1.7$ g</td>
</tr>
<tr>
<td>Initial Velocity</td>
<td>$V_s = 500$ m·s$^{-1}$</td>
</tr>
</tbody>
</table>

Table 4.3: Problem parameters and material properties for simulations of rigid sphere impact on alumina.

principal stress in a through-thickness cross section, (d) the trace of the fracture surfaces on a through-thickness cross section and (e) a three-dimensional rendering of the fracture surfaces. Figure 4-10 corresponds to time $t = 0.52\mu$s soon after the time of impact. It shows the incipient stress waves, including a strong state of hydrostatic compression directly underneath the impact point, a Rayleigh wave on the surface and a compression wave propagating through the cross section of the plate hemispherically. Figure 4-11 corresponds to time $t = 2.09\mu$s, soon after the stress wave has reached the back face. The maximum principal stress in the reflected wave is large enough to initiate the first cracks on the back surface, which is accompanied by a stress release on the newly-created free crack surfaces and behind the wave front propagating radially outwards. As Figure 4-11(e) shows, there is an ensuing fracture ring on the front face originated at the interface between the precursor surface wave propagating outwardly which and the hydrostatic-compression region underneath the impactor. At time $t = 3.4\mu$s, Figure 4-12, the development and propagation of radial cracks at the back face (b) and of conical cracks in the cross section (c),(d) can be clearly observed. The radial cracks are clearly driven by the high in-plane biaxial
tensile stress wave propagating outwardly. The stress level drops significantly behind
the radial crack tips. Similarly, the conical crack propagation is driven by the high
principal stress localizing on the conical surface (c). At time $t = 5.46\mu s$, Figure 4-13,
it becomes clear that the crack patterns are dominated by several radial cracks at the
back face still propagating toward the edges and a principal conical crack stemming
from underneath the impact zone. It can also be observed that the radial crack which
initially propagates towards the bottom left corner in Figure 4-13(b) branches out
into two cracks which propagate away from the corner and toward the adjacent edges.
Later on in the simulation, Figure 4-14, the fracture pattern continues to develop and
both the radial and conical cracks become clearly visible as crack openings grow to
a macroscopic scale. Figure 4-15 shows the final computed snapshot, after which no
further fragmentation is experienced by the plate. The trace of the conical and radial
cracks on the face at the end of the simulation is shown in Figure 4-16.

From this simulation, it can be seen that the fracture and fragmentation process
in an unconfined ceramic plate impacted by a high velocity penetrator is clearly deter-
mined by the propagation of stress waves in the material. Conversely, the propagation
of stress waves is substantially affected by the ensuing cracks, as the sequence of pres-
sure contours on the surface of the plate in Figures 4-10-4-15(a) clearly shows. More
specifically, stress concentrations at crack tips, stress release and posterior multiple
reflections at newly-created crack surfaces lead to very heterogeneous stress distribu-
tions. It is thus clear the importance of properly describing the propagation of stress
waves in the fracturing material. In particular, we highlight the potential importance
of post-fracture transmission of compressive waves across closed or closing cracks, a
challenge for the CZM approach which is clearly affected by the artificial compliance
issue discussed throughout this paper. As explained in Section 3.2, this is properly
handled in the proposed method by falling back to the normal component of the DG
fluxes when cracks close and the interfaces go into compression, thus guaranteeing
the transmission of longitudinal compressive waves as in the continuum uncracked
problem.

We finally show in Figure 4-17 a view of the back face in the deformed configuration
Figure 4-10: Snapshot of simulation results at $t = 0.52 \times 10^{-6}$s
Figure 4-11: Snapshot of simulation results at $t = 2.09 \times 10^{-6}$s
Figure 4-12: Snapshot of simulation results at $t = 3.4 \times 10^{-6} \text{s}$
(a) Hydrostatic stress contours on external surface of the plate

(b) Maximum principal stress contours on back face of the plate

(c) Contours of maximum principal stress in cross section

(d) Trace of fracture surfaces on cross section

(e) 3D rendering of fracture surfaces

Figure 4-13: Snapshot of simulation results at $t = 5.46 \times 10^{-6}$s
Figure 4-14: Snapshot of simulation results at $t = 10.9 \times 10^{-6}$s
(a) Hydrostatic stress contours on external surface of the plate

(b) Maximum principal stress contours on back face of the plate

(c) Contours of maximum principal stress in cross section

(d) Trace of fracture surfaces on cross section

(e) 3D rendering of fracture surfaces

Figure 4-15: Snapshot of simulation results at $t = 15.6 \times 10^{-6} \text{s}$
Figure 4-16: Radial cracks and trace of dominant conical crack on plate back face at the end of the simulation
at the end of the simulation with the different mesh partitions in different colors. The purpose of this figure is to demonstrate that in the parallel DG/CZM method proposed, cracks are able to propagate freely across the processor boundaries without the need of communicating any topological information between the processors.

Figure 4-17: Deformed back face view at the end of the simulation showing radial cracks propagating across mesh partitions
Chapter 5

Parametric Exploration of Damage Mechanisms in Normal Impact of Ceramic Tiles

Building on the results of the previous section, in this chapter we simulate the failure of monolithic Al₂O₃ plates subjected to normal impact by spherical projectiles. The main goal of this study is to investigate the dependence of fracture patterns on impact velocity and tile thickness using the full-field description provided by the simulations. This complements parametric experimental studies of normal impact on ceramic plates [2, 11, 14, 15, 16, 17, 18]. In general, experimental configurations for this type of experiments involve different degrees of confinement of the ceramic plate with a backing and lateral frame for two main reasons: 1) to keep sample in place and facilitate recovery and post mortem analysis, and 2) because practical uses of hard brittle plates in ceramic armor are known to have a much increased performance under confinement conditions. The effect of different type of boundary conditions provided by different types of confinement on the response of ceramic plate impact including the case of no confinement was investigated by Sherman, [15]. The effect of the confinement usually results in a more complex response. In this work, we focus on the impact response of unconfined and unsupported plates in order to decouple the fracture response of the plate with the complexities of the interaction with the
confinement system.

A spherical projectile with a mass density corresponding to steel is considered in the simulations, and is modeled as a rigid surface. The projectile radius \((r = 3.7 \text{mm})\) is small compared to the plate length used in all simulations \((l = 100 \text{mm})\) leading to a localized impact. In section 5.1 we consider three different velocities spanning the range 100–700 m·s\(^{-1}\) for a fixed tile thickness of 8mm, while in section 5.2 tile thicknesses of 6mm and 12mm are compared at a fixed velocity of 700 m/s. The resulting range of impact energies corresponding to these velocities are representative of experimental studies employing metallic projectiles (see e.g. [2, 11, 14, 15, 16, 17, 18, 22]). Since the upper range of velocities considered here greatly exceeds particle velocities observed at the Hugoniot elastic limit for Al\(_2\)O\(_3\) [117], we account for the possibility of inelastic deformation by employing the plasticity model for Al\(_2\)O\(_3\) proposed by Deshpande and co-workers [22]. Details on the model and its parameters are provided in Appendix C.

One finite element mesh was generated for each different plate thickness with ABAQUS CAE software using an advancing front algorithm. In order to enhance the randomness of the internal mesh structures, the meshes were optimized using HealMesh\(^1\) software. The final finite element meshes used in calculations had an average element size of approximately 700 \(\mu\text{m}\) and were composed of 679,029, 853,410, and 1,137,911 tetrahedra for plate thicknesses of 6, 8, and 12mm, respectively. These mesh sizes correspond to 20.8, 25.6, and 34.1 million degrees of freedom, respectively. Due to the large scale of these problem sizes, the simulations in this section were run on either the ARL Harold or the ERDC Diamond supercomputers on up to 512 processors. Owing to the high scalability of the parallel framework (as demonstrated in Sec. 3.5), each run took less than 48 hours to achieve a simulation time of 20 \(\mu\text{s}\).

\(^1\)HealMesh ©Parasim Inc.
5.1 Effect of Impact Velocity

Figures 5-1–5-3 show predicted cracking patterns in an 8mm tile for the three impact velocities after a simulation time of approximately 18 $\mu$s. The response for the lowest velocity, Figure 5-1, shows four radial cracks on the back face of the tile which were arrested before reaching the specimen edges. The lack of any visible cracking on the top surface of the specimen shows that these cracks did not propagate through the thickness. The nearly perpendicular arrangement of the four radial cracks is similar to "cruciform" type cracking patterns which have been the subject of several prior analytical studies, e.g. [121, 122]. Other than radial cracking, no other fracture mode is observed.

At the intermediate velocity approximately seven radial cracks are observed on the back face, Figure 5-2. The increased driving force also activates additional failure modes including the formation of circumferentially oriented cracks on the top surface of the specimen. This fracture mode closely resembles nested ring cracking patterns observed in previous impact experiments on ceramics and glasses, e.g. Figure 1-2. The lack of much visible ring cracking on the back surface of the tile suggests that the ring cracks are localized at the top surface, which has also been observed experimentally in some cases [1, 12, 13]. A fragmented zone with a conical shape is also observed in a localized region under the impactor.

At the highest impact velocity, Figure 5-3, the predicted cracking patterns share similar features with those for the intermediate velocity, including the formation of a fragmented zone under the impactor and the propagation of radial cracks. In contrast to the intermediate velocity, however, the predicted long range damage is significantly different. Instead of ring cracks, a non-axisymmetric fracture mode is visible on both the front and back surfaces of the tile in which cracks nucleate in a "box-shaped" pattern. Localized cracking is also observed normal to the specimen boundaries. These localized boundary cracks intersect with the specimen edges on one end and coalesce with the box-shaped cracks on the other end leading to the formation of rectangular fragments at the specimen boundary.
Figure 5-1: Simulated damage patterns in an Al₂O₃ tile impacted at 100 m·s⁻¹ showing a slice taken near the front surface of the tile (left) and a slice taken near the back surface of the tile (right). The displacements have been magnified by a factor of 10.

Figure 5-2: Simulated damage patterns in an Al₂O₃ tile impacted at 400 m·s⁻¹ showing a slice taken near the front surface of the tile (left) and a slice taken near the back surface of the tile (right). The displacements have been magnified by a factor of 10.
Recent high velocity sphere impact experiments by Compton and Zok on unconfined Al$_2$O$_3$ tiles [24] exhibit fracture and fragmentation features that are similar to the simulation results presented here for the highest impact velocity. Specifically, Figure 5-4 shows the post-mortem fracture patterns in a recovered Al$_2$O$_3$ specimen which was impacted at 1052 m·s$^{-1}$ by a 304 stainless steel sphere and supported by a 2mm steel backing plate. The experiments show the same fracture modes observed in the simulation, including a localized fragmented zone under the impactor, radial cracks, a box-shaped cracking pattern, and localized cracking at the specimen edges which results in rectangular fragments.

From Figures 5-1-5-3 we conclude that the extent of both short and long range damage changes dramatically as the impact velocity is increased. These results are qualitatively similar to the observations of Normandia [18] for experiments of WC sphere impact on confined SiC plates. In particular, the simulations show the same trend in which additional failure modes are activated with increasing impact velocity. Using the full field description provided by the simulations, we attempt to gain some insight into the mechanisms responsible for the impact velocity dependence of
damage. In Figures 5-5-5-11, we show montages of simulation results comparing the tile response to three impact velocities: 100, 400 and 700 m·s⁻¹. Each figure shows the results at a fixed time for the three different simulations to facilitate the comparison of the response for each case. The three leftmost images show contours of the maximum principal stress on the front face (top left), back face (top right), and in a through-thickness planar cross section below the impact point (bottom). Finally, the rightmost images show a top view of the fully fractured interface elements.

As the sphere impacts the plate surface longitudinal, shear and Rayleigh waves propagate from the impact point into the sample. The stress field analysis begins in Figure 5-5 which shows the response predicted at each impact velocity after reflection of the longitudinal and shear waves from the back face of the plate. At this point strong tensile principal stresses under the impactor have caused the localized nucleation of cracks under the impact site for each velocity. For the two larger impact velocities, the increased intensity of the impact leads to higher tensile stresses and the formation of a fragmented zone. For the smallest impact velocity, the tensile stresses
directly under the impactor are smaller and the extent of localized cracking is much reduced.

As the simulations progress, circular tensile fronts expand radially on the front and back faces of the samples and radial cracks appear, propagating in localized regions near back face behind the tensile front (Figures 5-6-5-7). For all cases, the maximum principal stress on the back face ahead of the propagating radial cracks is aligned with the hoop direction. This intuitive result confirms previous hypotheses that tensile hoop stresses are the main driving force for radial crack nucleation and propagation [16]. It is also evident from these figures that the spatial and temporal evolution of the maximum principal stress depends strongly on impact velocity and, in turn, determines the extent of long range cracking in the sample.

For the low velocity case, the maximum principal stress aligns with the hoop direction in localized regions near the back face. On the other hand, the expanding tensile front on the front face is aligned with the radial direction. The cross section images additionally reveal that the radially directed tensile front on the top face is localized near the surface. These characteristics of the stress field suggest that the tensile response of the top face is dominated by the propagation of Rayleigh waves, while tensile stresses near the back face are due to shear waves. The propagation fronts of the radial cracks remain localized near the back face, coinciding with regions of material that are under hoop tension. Eventually, the radial cracks are arrested before reaching the lateral specimen boundaries. This can possibly be attributed to geometric decay of the peak tresses in the expanding hoop stress front on the back face which would lead to a diminishing driving force for radial cracking to continue. At longer times (Figures 5-8-5-11) stress waves reach the lateral specimen boundaries and reflect, breaking the axisymmetry of the in-plane stress field and inducing weak tensile stresses near the boundary which are not of sufficient magnitude to activate additional fractures.

At the intermediate velocity, the maximum principal stresses in the leading tensile front are initially aligned with the hoop direction throughout the entire cross section (Figures 5-5-5-6). This suggests that at early stages in the impact event, the tensile
response is dominated by the propagation of shear waves. In contrast to the low velocity case, the stress field on the top surface undergoes a complex evolution at later stages in the simulation in which the rotational symmetry of the hoop stress front is significantly disturbed (Figures 5-7-5-9). This is mainly due to three effects: The appearance of non-uniform tensile radial stresses from Rayleigh waves, stress relief waves from the propagation of radial cracks which have reached the top surface, and stress wave reflections from the lateral specimen boundaries. In Figure 5-8, the tensile radial stresses on the top surface are of sufficient amplitude to cause the nucleation of circumferential cracks. Due to the non-uniformity of the radial stress front, these circumferential cracks nucleate at irregular spacings away from the impact point, and take the form of nested ring cracks. After activation of the circumferential failure mode, several radial cracks are arrested. Some radial cracks seem to be arrested through the interaction with stress relief waves from the newly formed ring cracks while others are arrested by propagating into and coalescing with the circumferentially oriented crack surfaces. Late in the simulation, some radial cracks continue propagating beyond the circumferential fracture front and towards the specimen boundaries, driven by strong tensile stresses from reflected waves. In these regions, the maximum principal stress aligns parallel to the specimen boundaries. Correspondingly, when the radial cracks encounter this boundary-affected zone, they alter their path of propagation, either by branching, or by following a curved path and they propagate perpendicular to the boundary.

For the highest velocity, the response of the plate shares some similar features observed at the intermediate velocity including the propagation of an axisymmetric hoop stress front over the entire cross at early stages (Figures 5-5-5-6) followed by the emergence of a strongly non-axisymmetric stress state caused by boundary reflections and unloading waves from radial cracks (Figure 5-7-5-8). Tensile radial stresses are also observed on the top face in Figure 5-8, possibly attributable to the propagation of Rayleigh waves. However, material regions experiencing radial tension appear to be significantly reduced in extent by radial cracks which have propagated quickly to the top surface, relieving the driving force for fracture and almost completely suppressing
the nucleation of a circumferential cracking mode. Another effect of the increased velocity is that tensile stresses resulting from boundary reflections are significantly amplified throughout the entire cross section, resulting from the increased intensity of compressive longitudinal waves from the impact. The strong state of tension arising from reflected waves leads to the activation of through-thickness spall cracks (Figure 5-8). Interestingly, the overall shape of the spallation fracture front mimics the square shape of the lateral specimen boundaries. In Figures 5-9–5-11, extended cracking continues in localized regions near the specimen boundaries, with cracks propagating perpendicular to the edges of the specimen.

5.2 Effect of Tile Thickness

In this section we build on the results of Section 5.1 and use large-scale simulations to investigate the influence of tile geometry on the resulting cracking patterns. The simulations compare the response of a thin tile (dimensions of 100x100x6mm) and a thick tile (dimensions of 100x100x12mm) subjected to high velocity sphere impact in the ballistic range \( V = 700 \text{ m/s} \). The final predicted cracking patterns for the 6 and 12mm tiles are shown in Figures 5-12–5-14 after 20 \( \mu \text{s} \) of simulation time.

For the thin tile, Figure 5-12 shows that the extent of predicted damage is quite severe due to the activation of a number of distinct failure modes including localized fragmentation under the impactor, radial cracks, and spallation near the specimen boundaries. A unique aspect of the damage sustained by the thin tile is that nearly all of the extended cracking (i.e. radial cracks, spallation fronts, boundary cracking) which visible on the top surface is also visible on the back surface, with seemingly little variation through the thickness. This leads to a damage distribution for the thin tile which is almost two-dimensional in nature, varying only in the plane of the plate. For the thick tile, the increase in cross section thickness seems to significantly reduce the overall damage sustained by the tile with fewer cracks visible on both the top and back surfaces (Figure 5-13). Another important effect of increasing the tile thickness is that through-thickness crack propagation becomes more difficult. From the thick
Figure 5-5: Snapshot of simulation results at $t = 4.0\mu$s showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right)
Figure 5-6: Snapshot of simulation results at $t = 6.0\mu s$ showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right).
Figure 5-7: Snapshot of simulation results at $t = 8.0\mu s$ showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right)
Figure 5-8: Snapshot of simulation results at $t = 10.0 \mu s$ showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right).
Figure 5-9: Snapshot of simulation results at $t = 12.0 \mu s$ showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right).
Figure 5-10: Snapshot of simulation results at $t = 14.0\mu s$ showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right).
Figure 5-11: Snapshot of simulation results at \( t = 16.0 \mu s \) showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right).
tile simulation, this is supported by the fact that there are more radial cracks visible on the back surface than on the top surface. Furthermore, a spallation failure mode can be seen on the top surface which is completely absent from the back face.

In Figure 5-14 we compare the cross section damage predicted for the two different thicknesses. In both cases extensive localized cracking occurs under the impactor which spans the thickness between the front and back faces. However, for the thin tile the material under the penetrator is completely pulverized and comminuted. The complete loss of material resistance due to this localized comminution allows the projectile to penetrate the tile. When the tile thickness is increased to 12mm, the failure mode under the penetrator changes from complete comminution of material to nested cone cracking and lateral cracking, resembling the experimental cracking patterns shown in Figure 1-1. This change in failure mechanism allows the thick tile to be more resistant to through-penetration by the fact that localized damage under the penetrator is more diffuse and spread over a larger volume. As a result, the thick tile retains its strength and defeats the penetrator, aided by connected regions of material under the penetrator which remain undamaged.

Several aspects of the fracture response predicted for the thin tile including the complete comminution of material under the penetrator, and the extensive fragmentation caused by extended cracking have never before been demonstrated in a 3D impact simulation using cohesive elements. In order to further illustrate these unique aspects of the simulation results presented in this section, we created several 3D renderings of the fracture patterns for the thin tile using ParaView. The renderings were completed using client-server mode whereby an interactive ParaView session was launched as a parallel job on the ERDC GPU visualization cluster and the manipulation of 3D simulation results in ParaView was done from my laptop. From this session, we produced several images illustrating the ejection of fragments from the back face (Figure 5-15) and the extensive fragmentation caused by extended cracking (Figure 5-16).

Next we attempt to explain the strong effects of tile thickness on the predicted cracking patterns using a visualization of the maximum principal stress and its di-
Figure 5-12: Simulated damage patterns in a 6mm thick Al$_2$O$_3$ tile impacted at 700 m·s$^{-1}$ showing a slice taken near the front surface of the tile (left) and a slice taken near the back surface of the tile (right). The displacements have been magnified by a factor of 10.

Figure 5-13: Simulated damage patterns in a 12mm thick Al$_2$O$_3$ tile impacted at 700 m·s$^{-1}$ showing a slice taken near the front surface of the tile (left) and a slice taken near the back surface of the tile (right). The displacements have been magnified by a factor of 10.
Figure 5-14: Fracture patterns in the cross section after a simulation time of $t = 20.0\mu s$ for tiles of (a) 6 and (b) 12mm thickness impacted at a velocity of 700m/s.
Figure 5-15: 3D renderings of the 6mm plate impacted at 700 m/s showing fragments being ejected from the back of the tile: (top) Back face view, (bottom) Side view. The displacements have been magnified by a factor of 2.
Figure 5-16: 3D renderings of the 6mm plate impacted at 700 m/s showing fragmentation due to significant extended cracking which has propagated through the thickness: Top face view (top), Back face view (bottom). Ejected fragments were filtered out during the rendering process in order to provide a full view of the extended cracking patterns. The displacements have been magnified by a factor of 15.
rection. In Figures 5-17–5-22 we compare the response of the 6 and 12mm tiles, using the same arrangement of views employed in Figures 5-5–5-11. Similar to the 8mm tile response discussed at length in Section 5.1, the response of both thicknesses is dominated by strong axisymmetric hoop stress fronts which span the entire cross section and propagate laterally towards the specimen boundaries (Figures 5-17 and 5-18). However, the cross section slices reveal that the for the thin tile, the hoop stress front is more uniform over the thickness and has a larger amplitude. Conversely, for the thick tile, the driving hoop stress front is diffused over a larger area and as a result, it has a smaller amplitude and a non-uniform distribution over the cross section. In Figure 5-19 the compressive longitudinal wave front reflects from the lateral boundaries, breaking the axisymmetry of the stress field and inducing large tensile stresses near the boundaries in both plates. In the thin tile, the uniformity of the stress field over the cross section prior to wave reflection produces an almost identical state of elevated tension on both the front and back faces. As a result, spallation cracks nucleate on both the top and back faces and propagate towards the opposing surfaces (Figures 5-20–5-22). Due to the large driving force available for fracture and the relatively small dimension of the thickness, a majority of these spall cracks propagate through the thickness. For the thick tile however, the non-uniformity of tensile stresses over the thickness previous to wave reflection leads to tensile stresses after wave reflection which are comparably larger on the top surface than on the back surface (Figure 5-20). Correspondingly, spallation cracks nucleate on the top surface only, and propagate towards the back face (Figures 5-21–5-22). Due to the larger thickness dimension and the comparably lower driving force available, the spallation cracks are not able to propagate fully through the thickness and never reach the back face. This explains the key differences in the extended cracking patterns observed previously in Figures 5-12 and 5-13.
Figure 5-17: Snapshot of simulation results at $t = 4.0 \mu s$ showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right)
Figure 5-18: Snapshot of simulation results at $t = 6.0 \mu s$ showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right)
Figure 5-19: Snapshot of simulation results at $t = 8.0\mu s$ showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right)
Figure 5-20: Snapshot of simulation results at $t = 10.0\mu s$ showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right).
Figure 5-21: Snapshot of simulation results at $t = 12.0\mu s$ showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right)
Figure 5-22: Snapshot of simulation results at $t = 14.0\mu$s showing the maximum principal stress contours on the front face (left), back face (middle), and in the cross section (bottom left), the fully fractured interface elements (right)
Chapter 6

Billion-size Simulations of Radial Cracking: Mesh Dependency and Convergence

Due to high computational cost of simulating 3D problems coupled with the paucity of scalable three-dimensional cohesive element algorithms for fracture, it had not been previously possible to assess the convergence properties of the cohesive element method in 3D. However, the high scalability of the DG/CZM approach enables the ability to investigate mesh dependence and convergence through large 3D simulations at an unprecedented scale. To this end, in this section we conduct the first systematic large-scale 3D investigation of mesh dependence of fracture patterns predicted by the cohesive element method. The driver problem that we choose to investigate is low velocity sphere impact on thin brittle plates. In Section 5.1 we found that for the material properties and problem parameters chosen, the dominant failure mode predicted for an impact at 100 m/s is the nucleation of radial cracks on the back face of the plate which propagate partially through the thickness and are arrested before reaching the specimen boundaries. This problem provides a convenient test-bed for investigating the convergence properties of the method for a 3D problem involving multiple cracks which is of great practical importance in a number of applications.
6.1 Mesh Refinement Scheme

In this chapter, impact damage calculations are carried out on a sequence of refined meshes obtained by recursive application of the distributed subdivision refinement algorithm described in section 3.4. This approach to mesh refinement allows for the systematic study of crack path dependence on the mesh because at each successive level of element subdivision, the number of possible crack path directions increases while the possible crack paths from all previous subdivision levels remain in the mesh. Furthermore, according to [101], the minimum quality measure of all meshes obtained by recursive application of the 8-subtetrahedron subdivision algorithm is bounded below by one half of that measure for the initial mesh. This guarantees that the method preserves the quality of the initial mesh at each level of mesh refinement and ensures that the time step for explicit dynamics calculations remains as large as possible.

The initial coarse mesh used in the calculations was obtained with Gmsh software [116] and was optimized to eliminate unwanted internal structure that might bias the predicted crack paths [50] (see Figure 6-1). After optimization, a baseline coarse mesh with 19,607 tetrahedral elements was obtained. Using this mesh as a starting point, impact damage calculations were carried out after applying a number of subdivisions, \( s = 0, \ldots, 4 \), to achieve different levels of mesh refinement. Table 6.1 provides details about the finite element meshes at different levels of subdivision. Utilizing the full machine access provided to us through the Pioneer testing program on the Hercules system, we were able to run a billion-size simulation in support of the convergence study which is the most highly-resolved fracture simulation ever attempted in two or three dimensions (corresponding to the \( s = 4 \) case in Table 6.1).

In Table 6.2 we provide a detailed account of the computational systems used for the different simulations, the number of processors employed in each case, and the total sustained compute time and corresponding number of CPU-hours used for each simulation.
Figure 6-1: The initial coarse mesh used in the mesh dependency calculations.

<table>
<thead>
<tr>
<th>s</th>
<th>E</th>
<th>D.O.F.</th>
<th>$h_{\text{min}}$ ($\mu$m)</th>
<th>$h_{\text{max}}$ ($\mu$m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>19,607</td>
<td>588,210</td>
<td>1080</td>
<td>8939</td>
</tr>
<tr>
<td>1</td>
<td>156,856</td>
<td>4,705,680</td>
<td>540</td>
<td>4470</td>
</tr>
<tr>
<td>2</td>
<td>1,254,848</td>
<td>37,645,440</td>
<td>270</td>
<td>2235</td>
</tr>
<tr>
<td>3</td>
<td>10,038,784</td>
<td>301,163,520</td>
<td>135</td>
<td>1117</td>
</tr>
<tr>
<td>4</td>
<td>80,310,272</td>
<td>2,409,308,160</td>
<td>67.5</td>
<td>558.5</td>
</tr>
</tbody>
</table>

Table 6.1: Details on the finite element meshes used in the convergence runs at different levels of subdivision. The variables $h_{\text{min}}$ and $h_{\text{max}}$ refer to the minimum and maximum edge size of the volumetric elements, respectively.

<table>
<thead>
<tr>
<th>s</th>
<th>D.O.F.</th>
<th>Platform(s)</th>
<th>Processors</th>
<th>Compute Time [hrs]</th>
<th>CPU-hrs</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>588K</td>
<td>Harold</td>
<td>64</td>
<td>4</td>
<td>256</td>
</tr>
<tr>
<td>1</td>
<td>4.7M</td>
<td>Harold</td>
<td>128</td>
<td>8</td>
<td>1,024</td>
</tr>
<tr>
<td>2</td>
<td>37.6M</td>
<td>Harold</td>
<td>512</td>
<td>17</td>
<td>8,074</td>
</tr>
<tr>
<td>3</td>
<td>301M</td>
<td>Diamond, Hercules</td>
<td>1,000(D); 17,264(H)</td>
<td>135(D); 7(H) 135K(D); 121K(H)</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>2.4B</td>
<td>Hercules</td>
<td>17,264</td>
<td>64</td>
<td>1.1M</td>
</tr>
</tbody>
</table>

Table 6.2: Details on the computational platforms, the number of processors, the total compute time, and the corresponding CPU-hours used for the convergence runs at different levels of subdivision.
Table 6.3: Details on the average output file sizes produced per time step at different levels of subdivision

<table>
<thead>
<tr>
<th>s</th>
<th>D.O.F.</th>
<th>Volume File Size (GB)</th>
<th>Interface File Size (GB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>588K</td>
<td>0.039</td>
<td>0.035</td>
</tr>
<tr>
<td>1</td>
<td>4.7M</td>
<td>0.248</td>
<td>0.237</td>
</tr>
<tr>
<td>2</td>
<td>37.6M</td>
<td>1.9</td>
<td>1.8</td>
</tr>
<tr>
<td>3</td>
<td>301M</td>
<td>15.2</td>
<td>14.4</td>
</tr>
<tr>
<td>4</td>
<td>2.4B</td>
<td>121.6</td>
<td>115.2</td>
</tr>
</tbody>
</table>

6.2 Visualization of Large-Scale Simulation Results

As a means for visualizing the simulation results, two sets of output files were produced at regular intervals during the simulations: 1) ParaView\(^1\) Multi-block datasets containing all of the volume elements with the stress and displacement fields saved at every nodal location, and 2) Tecplot\(^2\) datasets containing all of the interface elements with the displacement jumps and interface damage state (Eqn. 3.16) saved at every nodal location. In Table 6.3 we provide a summary of the average output file sizes per time step at the different levels of mesh refinement.

Examining the file sizes reported in Table 6.3, it is clear that as we push the boundaries of large-scale simulation to the billion-size, visualization of simulation results presents an exciting and mostly unexplored challenge of dealing with "big data". Summing the total file size for all simulation outputs at three and four levels of subdivision over 50 time steps, a total of 1.5TB and 11.84TB of data were produced, respectively. When the output files grow to this size, file transfers become exceedingly cumbersome, even on a high speed network. More importantly, conventional laptop or desktop machines do not possess enough physical memory to visualize the data for a single time step.

Faced with this "big data" challenge, we developed an advanced visualization approach using ParaView's parallel batch processing capabilities which allowed us to visualize the huge outputs produced in this convergence study. Our new visualization

\(^1\)More information available at [http://www.paraview.org](http://www.paraview.org)
\(^2\)More information available at [http://www.tecplot.com](http://www.tecplot.com)
technique leverages the existing "pvbatch" utility in which ParaView rendering can be controlled from a python script in an off-screen mode. The main advantage of doing so is that the off-screen rendering can be run as a parallel job on a conventional computing cluster, thus completely avoiding the need for file transfers and providing a tangible solution for dealing with huge file sizes. Extensive details describing our new parallel batch rendering techniques are provided in Appendix D.

Using parallel batch processing, we were able to produce full 3D renderings for most of the simulations which show a view of the simulated radial cracking patterns on the back face of the damaged specimens. The 3D renderings for the \( s = 0, \ldots, 2 \) cases were done directly on the compute nodes on Harold using up to 128 processors. For the \( s = 3 \) simulation, which was run on Diamond, the compute nodes on that machine did not possess enough physical memory to complete the rendering process and attempts at 3D rendering failed due to memory allocation errors. The files were then transferred to the ERDC utility server and the 3D rendering was completed successfully using a special "large-memory" visualization cluster which contains 32 compute cores and 256GB of shared memory per node. For the \( s = 4 \) case, the output files were so large that the 3D rendering could not be completed, even using large-memory visualization nodes. Harnessing the full power of ParaView in a parallel batch environment for rendering very large datasets is an important topic for ongoing research.

In addition to full 3D volume renderings of the damaged plates, a visualization of the radial crack surfaces was produced for each simulation by rendering the fractured interface elements in ParaView. For the \( s = 0, \ldots, 3 \) cases, the Tecplot outputs containing all of the interface elements were converted into ParaView files, and the fractured interface elements were extracted directly in ParaView using the interface damage measure (Eqn. 3.16) and the "Threshold" filter. Similar to the 3D volume renderings, the interface element renderings for \( s = 0, \ldots, 3 \) were done in parallel batch mode directly on the Harold and Diamond compute nodes.

The same process was also attempted for the \( s = 4 \) case. However, the conversion

---

of full outputs from Tecplot to ParaView took an impractical amount of time (up to 12 hours for one time step) and once the conversion process was finally completed, our attempts at rendering failed due to insufficient memory on the compute nodes of Hercules. In order to workaround these difficulties, a strategy was developed for filtering and downsizing the interface output files which enabled us to render the fracture surfaces for the $s = 4$ case. To do so, we wrote a python script which processed the Tecplot output files and extracted from them all of the fractured interface elements. The script then generated a new set of Tecplot outputs containing only the fractured interface elements, thus reducing the file sizes by more than an order of magnitude (see Appendix E for the python source code). The filtered Tecplot files were then converted to ParaView format and the fracture surface rendering could be completed. Anecdotally, we note that after filtering, the file size was so drastically reduced that we were able to do the rendering on a laptop machine.

In Figures 6-2–6-6 we show the different renderings of the cracking patterns that were produced for the convergence runs.

### 6.3 Mesh Dependency and Convergence Analysis

Examining Figures 6-2–6-6, it is clear that the number of radial cracks, the paths of radial crack propagation, and the final radial crack lengths are strongly dependent on the mesh size. From these renderings we can identify the following qualitative trends with mesh refinement:

- The number of dominant radial cracks initially increases and then saturates at 6 cracks
- The propagation length of the dominant radial cracks increases
- The number of crack branching events increases and the side branches propagating further before arresting
- The final length of the dominant radial cracks becomes more uniform with less overall deviation
Figure 6-2: \( s = 0 \) case: Full 3D rendering of final radial cracking patterns on the back face (top) and the trace of the radial crack surfaces on the back face produced from a rendering of the fractured interface elements at a time of \( t = 12 \mu s \). Displacements have been magnified by a factor of 150 in the 3D rendering.
Figure 6-3: $s = 1$ case: Full 3D rendering of final radial cracking patterns on the back face (top) and the trace of the radial crack surfaces on the back face produced from a rendering of the fractured interface elements at a time of $t = 12\mu s$. Displacements have been magnified by a factor of 150 in the 3D rendering.
Figure 6-4: $s = 2$ case: Full 3D rendering of final radial cracking patterns on the back face (top) and the trace of the radial crack surfaces on the back face produced from a rendering of the fractured interface elements at a time of $t = 12\mu s$. Displacements have been magnified by a factor of 150 in the 3D rendering.
Figure 6-5: $s = 3$ case: Full 3D rendering of final radial cracking patterns on the back face (top) and the trace of the radial crack surfaces on the back face produced from a rendering of the fractured interface elements at a time of $t = 12\mu s$. Displacements have been magnified by a factor of 150 in the 3D rendering.
Figure 6-6: \( s = 4 \) case: The trace of radial crack surfaces on the back face produced from a rendering of the fractured interface elements at a time of \( t = 12\mu s \). Because the file size was so large, a 3D rendering could not be produced.

- The propagation paths of the radial cracks become smoother and more physically realistic

The trend of increasing final crack length with mesh refinement can be explained by the fact that as the mesh is refined, more potential crack path directions become available in the mesh which align more closely with the theoretical crack propagation trajectory dictated the maximum principal stress direction (shown to be the hoop direction in Chapter 5). When the mesh is too coarse (Figures 6-2–6-4), exact radial paths are not available in the mesh and so the propagating crack tips follow jagged, oscillatory paths in an attempt to propagate in the radial direction. When the crack path is jagged, more energy is expended per unit of crack advance in the radial direction. This effect, known as "mesh-induced toughness", is reduced as the mesh is refined because the cracks tips are able to find paths in the mesh which are better aligned with the radial direction. Hence, for the same rate of energy flow into the crack tip, less energy is consumed by mesh-induced toughness for a finer mesh, and therefore, more energy is available for driving crack propagation in the
radial direction. As a result, the cracks propagate further before arrest as the mesh is refined.

The fracture surface renderings in Figures 6-2-6-6 provide a useful basis for extracting quantitative information describing the radial crack paths as a function of mesh refinement. Using Datathief, the spatial coordinates of the radial crack paths were extracted from these images using upwards of 50-80 data points to represent each crack. In Figure 6-7 the predicted radial crack paths for all mesh sizes are overlayed in the same plot, allowing a direct visual comparison to be made. Interestingly, we find that almost all of the dominant radial cracks for different mesh sizes follow similar paths with only slight deviations. This is most likely an artifact of the element subdivision scheme used for mesh refinement, for which all possible fracture paths from lower levels of resolution remain in the mesh as it is refined. Hence, even though more fracture paths become available with each successive subdivision, the signature of the initial mesh is inherently embedded for subsequent refinements.

Using the quantitative representation of the crack paths, we were able to extract precise measurements of the radial crack lengths for all the different mesh sizes, allowing a direct quantitative assessment of convergence of the fracture patterns. In particular, convergence is studied using two quantitative metrics describing the crack length, computed for only the dominant radial cracks at each mesh size:

1. The average total radial crack length \(L_t\).

2. The average Euclidean radial crack length \(L_e\), defined as the average distance from the radial crack locus to each radial crack tip.

In Figure 6-9 we plot the average total and Euclidean radial crack lengths as a function of the number of degrees of freedom. The plot confirms the general trend observed qualitatively in Figures 6-2-6-6 that the final length of the dominant cracks increases with mesh refinement. Interestingly however, the curve begins to level off between the \(s = 3\) and \(s = 4\) suggesting that the simulations are starting to approach the point of convergence, albeit in the sense of an average crack length. These results suggest that in order to predict relatively converged fracture patterns
for 3D problems involving multiple cracks, billions of degrees of freedom are required at a minimum.

Another important metric which has been used in previous work to assess the convergence rate of cohesive fracture simulations is the dissipated fracture energy [59]. In Figure 6-10 we compare the time evolution of the dissipated fracture energy for the different mesh sizes. This plot reveals that the energy dissipation is much more sensitive to mesh size than the average crack length and increases greatly with mesh size. The strong trend of increasing energy dissipation is confirmed in Figure 6-11 which shows the final dissipated cohesive energy as a function of the number of degrees of freedom. The increased energy dissipation with mesh refinement can be attributed to the increasing number of secondary cracks nucleating through crack branching, and the increased propagation length of these secondary cracks with mesh refinement. The final metric used to assess convergence in this simulation study is the time of complete crack arrest, which is extracted from the data in Figure 6-10 as the point at which the slope of the energy-time curve becomes zero. Figure 6-12 shows that similar to the average crack length, the arrest time is highly sensitive to the mesh size up to $s = 3$, at which point it seems to converge as the mesh is refined for $s = 4$. Although this convergence result is promising, the large variability in the predicted crack paths and the strong mesh dependence of the dissipated cohesive energy up to $s = 4$ demands that the mesh resolution be pushed even further, perhaps to the trillion-size. This is also confirmed by Table 6.1 which shows that the minimum and maximum edge lengths for the finite element mesh with $s = 4$ are 67.5 and 558.5$\mu$m, respectively. Hence, even at the billion-size, the finite element mesh is unable to resolve the size of the fracture process zone in ceramics. To do so, at least three more levels of subdivision will be required, corresponding to a mesh size of more than 1.2 trillion degrees of freedom.
Figure 6-7: Final radial cracking patterns after complete crack arrest extracted from Figures 6-2–6-6 using Datathief and overlaid on top of each other using a commercial plotting package.
Figure 6-8: The radial cracking data in Figure 6-7 is replotted here with the envelope of the average radial crack length overlayed on the plot for each value of $s$. This provides a qualitative visual depiction of convergence of the average radial crack length.
Figure 6-9: Average total radial crack length and average Euclidean (end-to-end) radial crack length after complete crack arrest as a function of the number of degrees of freedom. The diminishing slope of the plot suggests that the average radial crack lengths are starting to converge as the calculations reach the billion-size.
Figure 6-10: Temporal evolution of the dissipated cohesive energy during the radial cracking process at different levels of mesh refinement.
Figure 6-11: Total cohesive energy dissipated by the radial cracking process after complete crack arrest as a function of the number of degrees of freedom. The increased dissipation with mesh refinement indicates that we are far from convergence in this metric.
Figure 6-12: Time at complete arrest of radial cracks as a function of the number of degrees of freedom
Chapter 7

An Analytical Theory for Radial Crack Propagation

7.1 Motivation

In chapters 4 and 5 of this thesis, we used large-scale simulations to study how brittle material plates fail when they are subjected to localized impact loads. One of the main applications of this computational work is the damage response of armor tiles subjected to ballistic or fragment impact loadings. For this application area, predicting fundamental characteristics of the failure process in terms of the load, material and geometry is important in assessing structural integrity and protection performance. Clearly, large-scale numerical simulation is an important tool for predicting the impact response of structures and armor materials. However, analytical approaches based on fracture mechanics theory can also provide valuable physical insight.

Of the previous analytical work on predicting characteristics of impact fracture patterns, most studies have focused on the axisymmetric modes of failure. Some examples include the work of Bowden and Field [1] on explaining the nucleation radius of ring cracks in glass, the work of Evans and co-workers [11] on predicting cone crack lengths and lateral crack depth in ceramics, and the work of Iyer [20] on predicting cone crack lengths in SiC. While significant progress has been made...
in characterizing axisymmetric fracture modes, radial cracking in plates subjected to impact has received much less attention in the literature. This is likely due to the fact that it is an inherently three-dimensional effect in most situations involving localized impact.

Much of the previous analytical work on radial cracking has focused on the problem of pressurized star-shaped cracks in thin bodies (Figure 7-1a) [123, 124, 125, 126], which is relevant in the failure of pressure vessels. Somewhat less attention has been focused on the problem of traction-free star-shaped cracks in brittle materials subjected to external loads, which is relevant in the case of localized impact. Some notable exceptions can be found in the works of Sih and Irwin [127] and Rassoulova [128] on dynamic star-crack propagation under uniform far-field tension and Nakasa and Nakastuka [129] on radial crack branching. In [127], Sih and Irwin approximated a suddenly appearing traction-free star-crack under far-field tension as an expanding circular locus of stress relief and studied the near-tip stresses in comparison to the static case. An exact solution for the same problem was later derived by Rassoulova [128] who obtained an expression for the dynamic stress intensity factor. Nakasa and Nakastuka [129] studied the problem of radial cracking in thin brittle plates subjected to indentation loading. Treating radial cracking patterns as the result of repeated symmetric branching events, the authors developed theoretical expressions which implicitly defined the number of cracks that would propagate for given loading conditions and material properties. However, the branching assumption necessarily limited the possible range of crack numbers to powers of two. As a result the theory showed poor agreement with experiments.

Our interest is in developing an analytical theory to estimate the number of radial cracks in brittle material plates subjected to contact loads as a function of the loading, material, and geometric parameters. As a first step towards this goal, in this chapter we tackle the problem of predicting the number of radial cracks that will propagate in an infinite membrane under the action of axisymmetric, time-independent transverse boundary tractions. The purpose of developing this two-dimensional model is that it captures the effect of radial variation of the in-plane stress field on the resulting
7.2 Radial Cracking Theory for Membranes

To this end, consider a traction-free star-shaped crack with \( n \geq 2 \) segments in an infinite linear elastic membrane (see Figure 7-1a). A polar coordinate system attached to the center of the star-crack describes the geometry of the cracking pattern, with the initial crack surfaces defined by \( 0 \leq r \leq a \). The boundary tractions are applied as axisymmetric transverse loads with the center of the load application aligned with the star-crack center. The analytical solution to this problem can be constructed by superposition of the solutions of the following boundary value problems: A) the elastic solution for a thin body subjected to the same transverse loads but without a crack. This implies that the stress field for this problem is smooth; B) the elastic solution for a star-shaped crack subjected to normal tractions on the crack faces. The general solution for this case was derived by Williams [124]. Denoting the stress fields from problems A) and B) as \( \sigma^A \) and \( \sigma^B \), the solution to the general problem is obtained by superposition as \( \sigma = \sigma^A + \sigma^B \). The sought solution is obtained by imposing that the crack-face tractions vanish. This requires choosing the crack face tractions in problem B) so as to cancel the hoop stresses arising along the potential crack lines in the membrane.
problem A). Since the stress field in problem A) contains no singularities, the stress intensity factor in the resulting solution is simply \( K_I = K_I^B \).

Denoting the traction distribution acting in the direction of the outward unit normal on the crack faces as \( T = T(r) \), \( K_I^B \) is given by the following contour integral, [124]:

\[
K_I = K_I^B = \lim_{r \to a^+} \sqrt{2\pi (r-a)} \sigma_\theta^B = \frac{1}{i} \sqrt{\frac{a}{2\pi n}} \int_\Gamma \tilde{T}[T(r), a, s] \phi(a, n, s) \, ds \tag{7.1}
\]

In this expression, \( \phi(a, n, s) \) is an involved function of the flaw size, the number of cracks, and the contour integration variable \( s \) [124]. The integration contour, \( \Gamma \), runs parallel to the imaginary axis in the complex plane with \( s \) satisfying \( 0 < \text{Re} [s] < 1 \). Finally, the functional \( \tilde{T}[T(r), a, s] \) contains the dependence on the applied loading and is given by

\[
\tilde{T}[T(r), a, s] = \int_0^a -T(r) r^{s-1} \, dr \tag{7.2}
\]

where the traction-free crack condition requires that \( T(r) \) satisfy

\[
T(r) = -\sigma_\theta^A(r) \text{ for } 0 < r < a \tag{7.3}
\]

According to linear elastic fracture mechanics, a necessary condition for the propagation of the existing star-shaped cracks is

\[
K_I = \frac{1}{i} \sqrt{\frac{a}{2\pi n}} \int_\Gamma \tilde{T}[T(r), a, s] \phi(a, n, s) \, ds = K_c \tag{7.4}
\]

where \( K_c \) is the fracture toughness. Since \( n \) is the only unknown, Eq. 7.4 defines the number of cracks as a function of the applied loading, with a parametric dependence on \( K_c \) and the initial flaw size. The propagation of these \( n \) cracks will be unstable if the loading and fracture properties satisfy [130]

\[
\frac{\partial K_I}{\partial a} > 0 \tag{7.5}
\]
7.3 Uniform In-Plane Stress Approximation

Unfortunately, it does not seem possible to invert Eq. 7.4 for general loading conditions and solve for $n$ in closed form, nor does it seem possible to use Eq. 7.5 to derive a condition on the applied loading for which crack propagation would be unstable. However, a practicable solution can be found by noting that the number of cracks, as defined by Eq. 7.4, depends on the applied loading only through the local hoop stress distribution in Problem A) for $0 \leq r \leq a$ and does not depend on the details of the stress distribution outside of this region. In many practical cases of interest, the stress components for problem A) have a characteristic spatial scale associated with the loading which determines their spatial variation (e.g. the contact radius in Hertzian spherical indentation). If the initial flaw size is sufficiently small compared to the loading length scale, then the hoop stress from problem A) will have a slow variation over $0 \leq r \leq a$.

Based on this fact, we propose to approximate the hoop stress distribution $\sigma^A_\theta (r)$ for $0 \leq r \leq a$ as a uniform field. The stress intensity factor then becomes identical to that of a traction-free star-crack subjected to equibiaxial far-field tensile loading in the plane and $K_I$ reduces to a substantially simpler form. Specializing Eq. 7.1 to the case of an equibiaxial in-plane loading (denoted as $\sigma$), we have $\sigma^A_\theta (r) = \sigma$ and hence, $T (r) = -\sigma$. The elastic solution for the equibiaxial loading case was first derived by Sih and Irwin [127] based on the work of Westmann [123]. An equivalent, and more compact form of the solution can be found by adopting the expression provided by Williams [124] for Eq. 7.1 under the condition of uniform applied stress. Hence, we find

$$K_I = \sqrt{2\pi a} \sigma \psi (n)$$

where the function $\psi (n)$ is given by

$$\psi (n) = 2^{1-2n-1} n^{-1/2} \exp \left( -\frac{1}{\pi} \int_0^\infty F (x, n) \, dx \right)$$
\[ F(x, n) = \frac{\log[1 + x \sin(2\pi n^{-1}) \cosech(2\pi xn^{-1})]}{1 + x^2} \quad (7.8) \]

Combining Eq. 7.6 with the K-criterion, we can write
\[ \psi(n) = \frac{1}{\lambda} \quad (7.9) \]

where the non-dimensional parameter \( \lambda \) has been defined as
\[ \lambda = \frac{\sqrt{2\pi a\sigma}}{K_c} \quad (7.10) \]

Since the left hand side of Eq. 7.9 is only a function of \( n \), we find that the number of cracks is governed by the single non-dimensional parameter, \( \lambda \), which intuitively represents the ratio of the total \( K \)-field available to drive the propagation of a single Mode I crack, divided by the fracture toughness for a single crack. We also note that in the case of slow variation of the hoop stress over the initial flaw size for which Eq. 7.9 applies, the instability condition Eq. 7.5, is satisfied automatically.

Even though it is still not possible to invert Eq. 7.9 and solve for \( n \), the simplified form of this relation allows for approximate closed-form solution. Following Ouchterlony [125], we replace the star-shaped crack geometry by that of a hypocycloidal hole (see Figure 7-1b). Evidently, the stress intensity factor at the tips of the hypocycloidal hole has the same form as Eq. 7.6, with a geometry function \( \tilde{\psi}(n) \) which very closely approximates \( \psi(n) \), albeit with a much simpler form. For the case of the hypocycloid, the geometry function is given by \( \tilde{\psi}(n) = \sqrt{\frac{2}{n}} (1 - \frac{1}{n}) \). According to Clements and Widana [131], the approximation is almost exact for all practical values of \( n \). In particular, they showed that for \( n \) in the range \( n = 2, ..., 400 \) the maximum value of the relative error, \( \frac{|\psi - \tilde{\psi}|}{\psi} \), is at worst only 0.4%. Inserting \( \tilde{\psi}(n) \) in place of \( \psi(n) \) in Eq. 7.9, we obtain
\[ \sqrt{\frac{2}{n}} (1 - \frac{1}{n}) = \frac{1}{\lambda} \quad (7.11) \]

This expression leads to a quadratic equation for \( n \). Solving this equation and using
the requirement that \( n \geq 2 \) yields

\[
n = \lambda \left( \lambda + \sqrt{\lambda^2 - 2} \right)
\]

(7.12)

Hence we have derived a simple expression for the number of radial cracks that will propagate in a thin infinite body under equibiaxial in-plane stress conditions. We expect this to be a good approximation to predict radial cracking patterns in an infinite membrane subjected to out-of-plane contact forces when the hoop stress in the uncracked body has a slow variation over the initial flaw size. Inserting the expression for \( \lambda \) in terms of the problem parameters and simplifying gives

\[
n = \frac{2\pi a^2}{K_c^2} \left( 1 + \frac{K_c}{\sqrt{2\pi a}} \sqrt{2 \left( \frac{\pi a^2}{K_c^2} - 1 \right)} \right)
\]

(7.13)

From this solution we obtain the precise scaling behavior for the number of cracks:

\[
n \sim \lambda^2 \sim \frac{\pi a^2}{K_c^2} \sim \frac{\pi a^2}{EG_c}
\]

(7.14)

where we have introduced the fracture energy and the elastic modulus via the scaling relation \( G_c \sim \frac{K_c^2}{E} \). In Eq. 7.14 we can identify \( \frac{\pi a^2}{E} \) as the energy release rate for a Mode I through crack of length \( 2a \) in an infinite plate under plane stress conditions.

Equation 7.14 then states that the number of radial cracks is proportional to the total rate of energy flow available to drive the propagation of a single Mode I crack, divided by the fracture energy for a single crack, \( G_c \).

Next we note from the form of Eq. 7.13 that \( \frac{\pi a^2}{K_c^2} \geq 1 \) in order for a solution to exist which implies that the far-field loading must satisfy

\[
\sigma \geq \frac{K_c}{\sqrt{\pi a}} = \sigma_{2t}
\]

(7.15)

In this expression, \( \sigma_{2t} \) has been defined as the minimum value of the applied stress for propagation of a star-crack. Substitution into Eq. 7.13 shows that this value of the applied stress corresponds to the development of a single straight crack, or \( n = 2 \). On
the other hand, Rassoulova [128] obtained that for a suddenly appearing star-crack traveling at constant velocity under equibiaxial in-plane stress, the number of cracks must be at least three. The applied stress corresponding to the propagation of three cracks, denoted $\sigma_{3t}$, can be evaluated from Eq.’s 7.10 and 7.11 and is given by

$$\sigma_{3t} = \frac{3K_c}{2\sqrt{2\pi a}}$$  (7.16)

### 7.4 Extension to 3D Problems and Applications

Next we propose to apply the theory derived for membranes to situations involving axisymmetric normal loads on three-dimensional bodies. To this end, we assume that radial cracks will propagate in a three-dimensional body on the particular plane where the maximum hoop stress occurs in the uncracked continuum. Adopting a cylindrical coordinate system, the plane containing the maximum hoop stress is denoted as $z = z^*$ and the hoop stress evaluated along this plane is $\sigma_\theta^*(r) = \sigma_\theta(r, z = z^*)$. Given the hypothesis that radial cracking will initiate on the plane of maximum hoop stress, we can apply the radial cracking theory for the membrane problem as a first approximation to predict the number of radial cracks in three-dimensional situations by inserting $\sigma_\theta^*(r)$ into Eq. 7.3 and inverting Eq. 7.4 to solve for $n$. It is possible to do this numerically. However, a more insightful situation arises when the hoop stress features a slow variation over the initial flaw size, in which case Eq. 7.13 applies directly.

As a means for testing the accuracy of this approach, we consider the case of Hertz spherical contact on an elastic half space. The characteristic spatial scale which determines the variation of the stresses is the contact radius, $a_c$, and the relevant non-dimensional length scales are the non-dimensional radius $R = r/a_c$ and the non-dimensional depth $Z = z/a_c$. The maximum hoop stress, denoted $\sigma_\theta^{\text{max}}$, occurs at $R = 0$ (i.e. $\sigma_\theta^{\text{max}} = \sigma_\theta^*(0)$) [132, 20]. In order to interrogate the variation of the hoop stress over the initial flaw size, we expand the Hertz solution in the neighborhood of
$R = 0$ and for any $Z > 0$, which gives:

$$\sigma_\theta (R, Z) = F_0 (Z) + F_1 (Z) R^2 + O \left( R^4 \right) \quad (7.17)$$

Since we are concerned with the variation of the hoop stress only over $0 \leq r \leq a$, we introduce the “fast” non-dimensional length scale $\tilde{R} = r/a$ associated with the flaw size, and the perturbation parameter $\epsilon = a/a_c$. Then we can rewrite the previous expression as:

$$\sigma_\theta (\tilde{R}, Z) = F_0 (Z) + \epsilon^2 F_1 (Z) \tilde{R}^2 + O \left( \epsilon^4 \tilde{R}^4 \right) \quad (7.18)$$

Hence, if the initial flaw size is sufficiently small compared to the contact radius, then $\epsilon \ll 1$ and $\sigma_\theta (\tilde{R}, Z) \approx F_0 (Z)$ for $0 \leq \tilde{R} \leq 1$. This implies that on the plane of the maximum hoop stress, defined by $Z^* = z^*/a_c$, we have

$$\sigma^*_\theta (\tilde{R}) \approx F_0 (Z^*) \text{ for } 0 \leq \tilde{R} \leq 1 \quad (7.19)$$

where the term $F_0 (Z^*) = \sigma^\text{max}_{\theta}$ is identified as the maximum hoop stress. In conclusion, we find that we can employ the equibiaxial in-plane stress approximation to compute the number of cracks for Hertzian spherical indentation by setting $\sigma = \sigma^\text{max}_{\theta}$ in Eq. 7.13.

To illustrate the predictive features of the method in the context of a problem involving cracks due to contact loads, we apply the theory to the micro-indentation experiments of Wereszczak et al [133] who used 300 µm diameter diamond indentors to probe the deformation and failure response of several different armor grades of silicon carbide. An example of the final indentation patterns on SiC-SC-1R produced for different indentation loads are reproduced in Figure 7-2. According to the authors, radial cracks became visible on the top surface of the specimen for some critical indentation load and increased in number as the load increased. The trace of these radial cracks on the top surface of the sample can be clearly seen in Figure 7-2. A few caveats should be taken into consideration when applying our theory to this problem. First, the true nature of the experimentally observed cracks as surface radial cracks or
median cracks developing in the volume and propagating to the surface was unclear. Second, other important effects such as plastic deformation under the indenter and the development of conical cracks, are ignored. Finally, it is important to note that Eqn. 7.13 applies strictly to membrane configurations and hence, significant errors maybe introduced in applying it to three-dimensional situations. In particular, the membrane theory neglects the effects of out-of-plane bending stresses in the formulation of the stress intensity factor criterion for fracture, making the strong assumption that the part of the stress field responsible for radial cracking can be modeled by a biaxial stress state.

For a given indentation load, the maximum hoop stress is determined by considering Hertz’ contact solution at $R = 0$ [132]:

$$\frac{\sigma_\theta (R = 0, Z)}{p_0} = \frac{1}{\nu} \left[ 1 - Z \cdot \tan^{-1} \left( \frac{1}{Z} \right) \right] + \frac{1}{2} \left( 1 + Z^2 \right)^{-1}$$  \hspace{1cm} (7.20)$$

where the peak contact pressure $p_0$ is defined as

$$p_0 = \frac{3}{2 \pi a_c^2} \frac{P}{\rho}$$  \hspace{1cm} (7.21)
<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Diamond Indenter</strong></td>
<td></td>
</tr>
<tr>
<td>$E_s$</td>
<td>1141 GPa</td>
</tr>
<tr>
<td>$\nu_s$</td>
<td>0.07</td>
</tr>
<tr>
<td>$R_s$</td>
<td>150 μm</td>
</tr>
<tr>
<td><strong>Silicon Carbide Target</strong></td>
<td></td>
</tr>
<tr>
<td>$E_t$</td>
<td>450 GPa</td>
</tr>
<tr>
<td>$\nu_t$</td>
<td>0.17</td>
</tr>
<tr>
<td>$K_c$</td>
<td>1.5 MPa·m$^{1/2}$</td>
</tr>
<tr>
<td>$a$</td>
<td>2-3 μm</td>
</tr>
</tbody>
</table>

Table 7.1: Parameter values used in calculations

In these expressions, $\nu_t$ is the test material’s Poisson ratio, $a_c = \left(\frac{3PR_s}{4E^*}\right)^{1/3}$ is the contact radius corresponding to indentation load $P$ and sphere radius $R_s$, and $E^*$ is the effective contact elastic modulus defined as $E^* = \left(\frac{1-\nu_t^2}{E_s} + \frac{1-\nu_s^2}{E_t}\right)^{-1}$ (subscripts $s$ and $t$ refer to sphere and test material properties, respectively). We then obtain the maximum hoop stress by first solving

$$\frac{\partial \sigma_\theta}{\partial Z} \bigg|_{R=0} = 0$$

(7.22)

for the plane $Z = Z^*$, and then computing $\sigma_{\theta}^{\text{max}} = \sigma_\theta(R=0, Z = Z^*)$. Then, the number of radial cracks can be computed from Eq. 7.13. The basic elastic and fracture properties used for theoretical calculations are provided in Table 7.1.

The value of fracture toughness used in our calculations, $K_c = 1.5 \, MPa\sqrt{m}$, is chosen to be smaller than values measured in experiments for bulk silicon carbide.\footnote{Vargas-Gonzalez and Speyer [134] reported average fracture toughness values of $K_c \sim 2.5-4.5 \, MPa\sqrt{m}$ for several grades of SiC measured from four-point bend tests.} In general, experimental values of fracture toughness measured from bulk specimens of SiC are enhanced by the effective dissipation of microcracking in the formation of the fracture process zone. Since the indentation experiments of Wereszczak et. al, [133] were carried at the micro-scale and the crack lengths were on the order of 50-100 μm, we expect the effects of microcracking, which are present in fracture toughness tests on bulk specimens, to be largely absent. Hence, we argue that a value of the fracture toughness smaller than bulk measured values is appropriate. The value of $K_c$
In Figure 7-3 we compare theoretical predictions for the number of cracks as a function of the indentation load with the experimental data from [133]. Evidently, the trend of increasing numbers of radial cracks with increasing load is captured accurately by the theory within a realistic physical range of initial flaw sizes for armor grade SiC. However, it is also clear from Figure 7-3 that the prediction for the number of radial cracks is very sensitive to the choice of the initial flaw size. As such, it should be emphasized that this result does not represent a strict validation of the theory. That would require comparison of theoretical predictions with a more complete set of indentation experiments with controlled initial flaw sizes, accompanied by statistical analysis of the number of radial cracks. Since a more complete set of validation data is not available in the literature, the predictions made in this chapter represent an initial attempt to validate the theory to the extent possible using available experimental data.
Chapter 8

Summary and Recommendations for Future Work

Understanding the failure response of brittle materials subjected to localized impact loading is a challenging problem of critical importance in a number of industrial, defense, and space applications. With the continued development of powerful supercomputing resources, direct numerical simulation will be a central tool for studying brittle impact damage for years to come. Although fracture mechanics theories have been developed which are amenable to implementation in numerical codes, a 3D method capable of simulating fracture and fragmentation at a large scale has until this point, remained elusive. As a result, most previous work on modeling impact damage was restricted to two dimensional simulations conducted in axisymmetric mode, and important 3D effects such as radial cracks, were ignored. However, this thesis has furnished the first solution to this problem. By combining the discontinuous Galerkin framework with cohesive zone models of fracture, we developed the first scalable framework for simulating dynamic fracture of brittle solids in three dimensions (an idea introduced in my Master’s research, [42, 4] and by Mergheim et al [136]). In the work for this PhD thesis we refined the DG/CZM formulation, improved its robustness, and implemented the algorithm in the MIT Summit research code for massively-parallel simulation. A unique feature of the DG/CZM framework is that it does not require topological changes to the mesh as cracks and fragments
develop, allowing for the seamless propagation of cracks across processor boundaries and thus, scalability in parallel computations. This was demonstrated in an extensive set of both strong and weak scalability runs on DoD supercomputers using up to 17,264 processors (full-machine size) and mesh sizes up to 19.3 billion degrees of freedom. Utilizing this new capability, we then applied the method to conduct the first systematic investigation of impact damage in brittle plates using large-scale 3D numerical simulation.

A critical component of this work was a strong effort to validate the numerical framework by direct simulation of impact experiments and qualitative and quantitative comparison of simulations against experimental results. In the first study to this end, we simulated the Mode I edge-on impact experiments of Umberger and Love [111] on notched PMMA plates. The simulations captured the essential qualitative features of crack propagation observed in the experiments including the propagation of a single Mode I crack from the notch tip which reached a steady propagation speed before eventually branching into multiple cracks. The crack tip location over time was tracked during the simulations using the algorithm described in Appendix B. From this data, the steady crack tip speed was extracted for comparison against experiments. Using both a rate-independent and more physically realistic rate-dependent cohesive model, we found that the numerical framework was able to accurately capture the experimentally observed Mode I crack speeds over a wide range of impact velocities. In a second validation test, we simulated the edge-on impact experiments of Strassburger and Senf on Al₂O₃ plates. This test presented a considerably more difficult challenge for validation as it involved the simultaneous nucleation of multiple cracks propagating in different directions. The simulated cracking patterns were compared to experiments at two different impact velocities and were shown to capture many qualitative features of the experimentally observed fracture patterns including the propagation of shear-driven cone cracks from edges of the contact zone and the nucleation and propagation of cracks in an expanding radial damage front. Quantitative comparisons were made between simulations and experiments for both the cone crack length and the extent of the damage front over time and reasonable agreement.
was obtained for both metrics, especially considering the inherent uncertainty in elastic and fracture properties and in the experimental results. A detailed analysis of the maximum principal stress and its direction were then used to gain insight into the driving force for fracture and its dependence on impact velocity.

The validation tests involving edge-on impact provide confidence in the ability of the DG/CZM approach to capture the crack speeds of both Mode I and mixed-mode cracks. However, significant work still remains to verify the accuracy of crack path and crack speed predictions against experimental results. More comprehensive validation studies involving EOI are required for both the Mode I configuration with notched specimens (section 4.1.1) and the mixed-mode configuration with un-notched specimens (section 4.1.2). For Mode I tests, future validation work should include comparison of the simulated crack paths to high-speed images of the experimental cracking patterns at different times and quantitative comparison of the crack tip speed evolution. For crack path comparisons, special attention should be paid to how well the numerical framework can capture crack path instabilities at high crack speeds such as the onset of crack branching and the subsequent branching angles and branched propagation paths. For the mixed-mode configuration, further work on validation should include comparisons of simulated and experimental cone crack and damage front trajectories over a greater range of impact speeds. Another critical, and yet challenging, metric for validation will be to compare the spatial location of crack nucleation sites and the subsequent propagation paths of cracks in the expanding radial damage front. This will likely require advanced experimental procedures using the state of the art high speed cameras to produce high resolution images of the surface cracking patterns in addition to extensive image analysis to compare simulations and experiments.

The final validation test considered in chapter 4 was a model problem involving normal impact on a thick ceramic tile. In this example, we illustrated the ability of the numerical approach to capture the fundamental failure modes observed in ceramic plate impact experiments including conical, radial and lateral cracking patterns. Since the predicted fracture patterns were not compared directly to experimental results,
this example did not represent a strict validation of the numerical approach for the normal impact problem. Nevertheless, it provided confidence that the approach can capture both the axisymmetric and three-dimensional fracture modes commonly observed in ceramic impact experiments. This example also proved that the DG/CZM approach is able to seamlessly propagate cracks across processor boundaries, a unique feature of the method which has never before been demonstrated for an approach based on CZM in three dimensions. Building on this example, in chapter 5 we conducted a systematic parametric study of normal impact on ceramic plates with the goal of understanding the parts of the stress field which drive different fracture modes as a function of impact velocity and tile thickness. Using a detailed visualization of the maximum principal stress and its direction, we were able to gain new insight into the basic driving forces for axisymmetric and non-axisymmetric cracking modes and how they depend on these problem parameters.

A central result of this simulation study is that prior to reflection of stress waves at the lateral boundaries, the damage response of the tile is driven by maximum principal stresses which are aligned primarily with the in-plane hoop direction. The analysis of the stress field evolution revealed that these hoop stresses propagate laterally in the tile as axisymmetric fronts, with radial crack tips propagating in their wake. This proved the intuitive hypothesis suggested previously in [16, 17] that crack propagation along radial paths must be driven by stresses aligned with the hoop direction. Although tensile hoop stress fronts were an essential feature observed in all simulations, the amplitude of these stresses and their distribution through the thickness depended strongly on both impact velocity and tile thickness, and as a result, the through-thickness propagation path of radial crack fronts depended strongly on these parameters. In particular, we found that for increasing impact velocity and decreasing tile thickness, there was a greater tendency for radial crack fronts to propagate completely through the thickness, driven by large tensile hoop stresses spanning the entire cross section. Another important result of the stress field analysis was an improved understand of the effects of stress wave reflections at the lateral boundaries of the tiles. To this end, we found that reflected stress waves induce non-axisymmetric ten-
sile stresses near the lateral specimen boundaries. Above a certain threshold impact velocity, the amplitude of reflected stress waves was sufficient to activate a spallation failure mode near the specimen boundaries, in close proximity to the propagating radial crack tips. This was shown to significantly alter the propagation paths of the radial cracks, causing them to either curve, branch, or be arrested.

While this investigation into the impact damage physics revealed some key insights, significant work remains to build a more complete understanding of the failure mechanisms in projectile impact scenarios. In the context of projectile impact damage in protective materials, further simulation studies should be conducted to probe the effects of other key problem parameters which were not considered in this thesis. Future work should focus on understanding the damage physics in more advanced structural configurations which are typically studied in impact experiments. Some examples include laterally confined plates and layered composite structures [16, 17]. Parametric simulation studies should also be used to investigate the role of basic elastic and fracture properties of both the target plates and the projectiles. In particular, the deformation and failure of the projectile should be explicitly modeled in the simulations as it represents a key aspect of the impact physics which was ignored in the simulation studies for this thesis. Another critical direction of future work is the development of cohesive zone models which provide a better description of dynamic fracture processes. To this end, models should be developed which include dynamic effects, such as rate-dependence of both the nucleation and propagation fracture toughness. A more accurate description of mixed-mode crack propagation is also needed which captures the variation of the fracture energy as a function of the relative amount of opening and sliding at the crack tip. A more advanced fracture criterion must also be developed which better characterizes fracture initiation in brittle materials under combined states of compression and shear.

Another important contribution of this thesis is the first systematic large-scale simulation study of mesh dependence and convergence of the cohesive element method in 3D. In this mesh dependency study, convergence of radial cracking patterns was studied for a sequence of refined meshes using up to 2.4 billion degrees of freedom.
and run on up to 17,264 processors. The ability to run full simulations at such a large scale was enabled both by the scalability of the DG/CZM framework and by access to several supercomputing platforms from ARL and ERDC. In support of this convergence study, a new parallel batch rendering process was also developed to visualize the radial cracking patterns using ParaView’s "pvbatch" utility. This allowed us to avoid cumbersome file transfers and provided a tangible solution for rendering the huge files produced for simulations up to the billion-size. Convergence was studied by comparing the results at different mesh sizes for several metrics including the number of cracks, the crack paths, the average total and Euclidean radial crack lengths, the dissipated cohesive energy, and the time at complete crack arrest. A promising result of this study is that for both the average crack length and the crack arrest time, the results started to converge for the two largest mesh sizes. However, the number of cracks, the crack paths, and the dissipated cohesive energy all depended sensitively on the mesh size and failed to converge, even for the billion-sized mesh. The lack of convergence in these metrics indicates that even finer meshes are needed, perhaps up to the trillion-size. This underscores the critical importance of the scalability offered by the DG/CZM framework, as it is the only computational tool currently available that can simulate fracture at such an unprecedented scale. Building on the results of this initial study, future efforts to assess convergence should focus not only on mesh size, but also on the effect of mesh topology. In order to gain a more fundamental understanding of how CZM-based methods converge in 3D, studies should be conducted in the future for a range of different boundary value problems, including both static and dynamic loadings.

In the final chapter, we attempted to develop a model for predicting the number of radial cracks in brittle materials subjected to contact loadings. For simplicity, we considered the model problem of an infinite elastic membrane containing a star-shaped crack subjected to axisymmetric transverse loads. Using the principle of superposition, we derived an expression for the stress intensity factor at the radial crack tips. Combining this relation with the critical stress intensity factor condition, we obtained an involved expression which implicitly defined the number of radial
cracks that would propagate for a given loading, fracture toughness, and initial flaw size. Although this expression was too complicated to obtain a closed form solution for the number of cracks, it did provide a critical insight; the number of cracks depends on the stress field only through the local hoop stress distribution along the initial crack lines. Based on this fact, we argued that if the flaw size is sufficiently small compared to the length scale associated with the spatial variation of the stress field, then the membrane problem with out-of-plane loading can be approximated by a star shaped-crack in a thin sheet subjected to equibiaxial in-plane loading. An approximate solution was then obtained for this simplified boundary value problem from which we obtain a closed-form expression giving the number of radial cracks in terms of the applied loading, the initial flaw size, and the fracture toughness. This key result revealed some of the basic physics underlying the radial cracking mechanism. In particular, we found that the number of cracks is governed by just a single non-dimensional parameter. This parameter can be interpreted physically as the ratio of the total K-field available to drive the propagation of a single Mode I crack divided by fracture toughness for a single crack. We then proposed to use the membrane theory as an approximate model for predicting the number of cracks in three-dimensional problems involving contact loads. In a preliminary validation test for this approach, we used the approximate theory to predict the spherical indentation experiments of Wereszczak et al. [133] and reasonable agreement was obtained for the number of radial cracks as a function of indentation load.

The theoretical developments presented in chapter 7 represent an important contribution to the scientific understanding of radial cracking patterns. However, this is only a small step towards the long-term goal of developing a theory to characterize radial cracking patterns in brittle plates subjected to projectile impact loads. Nevertheless, the fracture mechanics approach used to obtain the number of cracks for the membrane problem can be used as a general framework for treating more complicated boundary value problems. To reach the goal of a radial cracking theory in projectile impact scenarios, future work should aim to derive stress intensity factors for a series of boundary value problems of increasing complexity. For example, a logical next step
would be to consider the problem of predicting the number of radial cracks which will propagate in a thin sheet subjected to expanding in-plane stress wave loading. A solution for the stress intensity factor for a star-shaped crack subjected to this dynamic loading will likely involve a superposition procedure similar to the approach employed by Freund [137] for a Mode I crack subjected to plane wave loading. Another important problem that should addressed in future work is the development of a theory for characterizing the dynamics of radial crack propagation under time-varying dynamic loadings. Such a theory would be critical for understanding dynamic crack path instabilities such as crack branching and for predicting crack arrest.
Appendix A

Penalty Contact Algorithm

In all of the impact simulations presented in this thesis, we employ a simple penalty-based algorithm to apply contact forces between the projectile and the impacted surface. The algorithm encompasses the following steps:

- Before the time step loop begins, a surface mesh of 2D triangular elements is created, representing all of the facets which lie on the impacted surface. The surface elements are endowed with quadrature points, allowing for fields to be defined over them with a quadratic interpolation.

- At each step within the time step loop contact forces are applied to the surface and the projectile using the following process (illustrated schematically in Figure A-1):
  - The projectile is allowed to penetrate the sphere by a trial displacement of \( \Delta s = V_p \Delta t \) where \( V_p \) is the velocity of the projectile and \( \Delta t \) is the current time step.
  - After the trial projectile displacement, the subsurface penetration distance \( \delta \), is measured at every quadrature point for each surface element.
  - From the penetration distance defined at each quadrature point, a corresponding contact force field is calculated at the quadrature points by
multiplying by a penalty factor, $K$:

$$F = K\delta$$ \hspace{1cm} (A.1)

- The penalty contact force field defined at the quadrature points is then interpolated to the nodes so it can be applied as an external traction in the Newmark integration.

- The total resultant contact force applied to the impacted surface is then computed by integrating the penalty contact force field over all the surface elements on each processor and then summing across the processors using an MPI reduction.

- An equal and opposite resultant force is then applied to the projectile.

- The rigid body motion of the projectile is then computed analytically and its velocity is updated for the next integration step.
Figure A-1: A schematic illustrating the penalty-based algorithm used to enforce the contact constraint in impact simulations in this thesis.
Appendix B

Crack Tip Tracking Algorithm

For the Mode I edge-on impact simulations in Section 4.1.1, we developed an algorithm to estimate the location of the crack tip at each step in the calculation. The time history of the crack tip location was then used to extract the steady state crack tip speed for comparison against experiments. The algorithm for tracking the crack tip checks every interface element on each processor after the explicit Newmark integration step to find the node with full damage which corresponds to the furthest extent of the main crack in the propagation direction. In order to discern between the main crack and cracks nucleating elsewhere in the body, the search for damaged nodes is confined to a small region around the current crack tip location (illustrated schematically in Figure B-1).

The crack tip tracking algorithm encompasses the following steps:

- A small box is defined which initially encompasses the notch tip at the first step in the calculation.

- After the completion of each explicit Newmark time integration step we do the following on each processor:
  - Loop over the nodes of each interface element and test to see if the damage measure, defined by Equation 3.16, has reached a value of 1 at any node.
  - If so, test to see if the coordinates of that node fall within the box.
- If this node falls within the box and exceeds the x-location of the current crack tip, set the new crack tip location to the coordinates of the node in question.

- Collect all of the trial crack tip locations into an array on processor 0 using a call to MPIGather().

- On processor 0, search through the trial crack tip locations and determine the one with the largest x-coordinate. This is the new global crack tip.

- Broadcast the coordinates of the new global crack tip to all processors using a call to MPIBcast() and update the local crack tip coordinates on each processor to the new global value.

- Update the position of the search box, moving it to the new global crack tip location.
Figure B-1: A schematic illustrating the search box used to estimate the crack tip location in Mode I edge-on impact simulations of section 4.1.1.
Appendix C

Deshpande-Evans Ceramic Plasticity Model

The ceramic constitutive model employed in chapter 5 is based on the model proposed by Deshpande, et. al in [22]. The model uses an additive decomposition of the logarithmic strain into an elastic and inelastic part

$$\mathbf{e} = \mathbf{e}^e + \mathbf{e}^p \quad (C.1)$$

where $\mathbf{e}$ is the total logarithmic strain, $\mathbf{e}^e$ is the elastic strain, and $\mathbf{e}^p$ is the inelastic strain. The deviatoric stress, $\mathbf{S}$, and the pressure, $p$ are obtained from the elastic part of the strain tensor via Hooke’s law with

$$\mathbf{S} = 2G (\mathbf{e}^e - \text{tr}\mathbf{e}^e \mathbb{I}) \quad (C.2)$$

and

$$p = -\kappa \text{tr}\mathbf{e}^e \quad (C.3)$$

where $G$ is the shear modulus, $\mathbb{I}$ is the identity tensor, and $\kappa$ is the bulk modulus. The total stress then follows from $\mathbf{\sigma} = \mathbf{S} - p\mathbb{I}$. 

Two regimes of rate-dependent plastic flow in Al$_2$O$_3$ are accounted for via a rate-dependent $J2$ flow rule: at low strain rates, resistance to plastic flow within the crystal
lattice governed by the Peierls stress, and at high strain rates, resistance governed by phonon drag. Based on these mechanistic considerations, Deshpande, et al propose the following flow rule for the equivalent plastic strain rate, $\dot{\varepsilon}_{eq}$:

$$
\dot{\varepsilon}_{eq} = \begin{cases} 
\left( \frac{\dot{\varepsilon}^0}{\dot{\varepsilon}} \right)^{(1-n)/n} \left( \frac{\bar{\sigma}}{\dot{\varepsilon}^0} - 1 \right) & 2\bar{\sigma} > \sigma_0 \left( \left( \frac{\dot{\varepsilon}^0}{\dot{\varepsilon}} \right)^{1/n} + 1 \right) \\
\left( \frac{\bar{\sigma}}{\dot{\varepsilon}^0} - 1 \right)^n & \sigma_0 < 2\bar{\sigma} \leq \sigma_0 \left( \left( \frac{\dot{\varepsilon}^0}{\dot{\varepsilon}} \right)^{1/n} + 1 \right) \\
0 & 2\bar{\sigma} < \sigma_0 
\end{cases}
$$

(C.4)

where $\sigma_0$ is the flow stress, $\bar{\sigma} = \sqrt{\frac{3}{2} S : S}$ is the von Mises stress, $\dot{\varepsilon}_0$ and $n$ are the reference strain rate and strain rate sensitivity exponent, and $\dot{\varepsilon}_t$ is the transition strain rate above which the plastic response becomes phonon drag limited. The flow stress is given by the conventional power law form,

$$
\sigma_0 = \frac{\sigma_Y}{2} \left[ 1 + \left( \frac{\dot{\varepsilon}_{eq}}{\dot{\varepsilon}_Y} \right)^M \right]
$$

(C.5)

where $M$ is the strain hardening exponent, $\sigma_Y$ is the uniaxial yield strength, and $\dot{\varepsilon}_Y$ is the value of the equivalent plastic strain at which $\sigma_0 = \sigma_Y$. Assuming a flow potential $G_p = \bar{\sigma}$, the inelastic strain rate tensor is then obtained from

$$
\dot{\varepsilon}^p = \dot{\varepsilon}_{eq} \frac{\partial G_p}{\partial \sigma}
$$

(C.6)

The material properties for the plasticity model used in the simulations in chapter 5 were taken from the work of Gamble et al [36] in which the model was fit to quasi-static indentation experiments on Corbit 98 Al$_2$O$_3$\(^1\), Table C.1.

\(^1\)Produced by Industrie Bitossi Sovigliana Vinci, Italy
<table>
<thead>
<tr>
<th>Properties</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( \text{Al}_2\text{O}_3 ) Plate</td>
</tr>
<tr>
<td>Length</td>
<td>( L = 100 \text{ mm} )</td>
</tr>
<tr>
<td>Thickness</td>
<td>( t = 6, 8, 12 \text{ mm} )</td>
</tr>
<tr>
<td>Initial density</td>
<td>( \rho_0 = 3864 \text{ kg} \cdot \text{m}^{-3} )</td>
</tr>
<tr>
<td>Elastic Modulus</td>
<td>( E = 366 \text{ GPa} )</td>
</tr>
<tr>
<td>Poisson's Ratio</td>
<td>( \nu = 0.239 )</td>
</tr>
<tr>
<td>Yield Stress</td>
<td>( \sigma_Y = 5.75 \text{ GPa} )</td>
</tr>
<tr>
<td>Strain Hardening Exponent</td>
<td>( M = 0.1 )</td>
</tr>
<tr>
<td>Strain Rate Dependence Exponent</td>
<td>( n = 34 )</td>
</tr>
<tr>
<td>Reference Plastic Strain Rate</td>
<td>( \dot{\varepsilon}_0 = 0.001 )</td>
</tr>
<tr>
<td>Transition Shear Strain Rate</td>
<td>( \dot{\varepsilon}_t = 10^6 )</td>
</tr>
<tr>
<td>DG Stability Parameter</td>
<td>( \beta = 4 )</td>
</tr>
<tr>
<td>Cohesive Strength</td>
<td>( \sigma_c = 400 \text{ MPa} )</td>
</tr>
<tr>
<td>Fracture Energy</td>
<td>( G_c = 25 \text{ J} \cdot \text{m}^{-2} )</td>
</tr>
<tr>
<td>Tension/Shear Weighting</td>
<td>( \gamma = 1 )</td>
</tr>
<tr>
<td>Interface Viscosity Parameter</td>
<td>( c_v = 10^{-1} \text{ MPa} \cdot \text{s} \cdot \text{m}^{-1} )</td>
</tr>
<tr>
<td>Rigid Sphere</td>
<td></td>
</tr>
<tr>
<td>Radius</td>
<td>( R_s = 3.6 \text{ mm} )</td>
</tr>
<tr>
<td>Mass</td>
<td>( m_s = 1.5 \text{ g} )</td>
</tr>
<tr>
<td>Velocity</td>
<td>( V_s = 100, 400, 700 \text{ m} \cdot \text{s}^{-1} )</td>
</tr>
</tbody>
</table>

Table C.1: Plate and projectile dimensions and material properties used in normal impact simulations.
Appendix D

Strategy For Visualization of Very Large Datasets: Parallel Batch Rendering

In this appendix we provide details about the new batch rendering process that we developed in this thesis to visualize the very large datasets generated in the convergence study of Chapter 6. The process we developed is based on ParaView's "pvbatch" utility which allows the user to control ParaView's rendering process from a python script, in an off-screen mode. The first step in the rendering process is to produce a state file for the desired ParaView scene to be rendered. ParaView state files are XML formatted files which contain all of the commands executed "under the hood" by ParaView when the user creates a scene using the interactive GUI. Since it is generally not possible to interactively visualize a very large dataset on a conventional desktop using the ParaView GUI, we first produced a state file for the desired ParaView scene using a low resolution dataset. The basic idea we had then, is to modify the state file for the low resolution dataset so that it can be used in batch mode to visualize very large datasets that cannot be visualized interactively.

In the ParaView state file, the file path to the low resolution dataset is hard-coded throughout the file in several different places. To make this file amenable to visualizing other datasets, we replaced the hard-coded file path in each location with
a simple tag. This is illustrated in Figure D-1.

```xml
<Proxy>
  <Proxy group="sources" type="XMLP\'structuredGridReader" id="3443" servers="#1">
    <Property name="FileName" id="3443.FileName" number_of_elements="#1">
      <Element index="#0" value="/home/seagaves/dv/low_speed/old_results/u_sub0/fields-00021.pvtu"/>
      <Domain name="files" id="3443.FileName.files"></Domain>
    </Property>
  </Proxy>
</Proxy>
```

Figure D-1: A portion of the ParaView state file constructed using low resolution data showing the hard-coded file path to the low resolution dataset (top) and the same code snippet with the hard-coded file path replaced by a simple tag (bottom).

Using the modified state file as a template for rendering, we wrote a python script under the pvbatch formalism which loads the modified state file, searches for the file tag, replaces it with the file path to the high resolution data and then renders the ParaView scene producing an output .png image. The python source code for batch rendering is provided below.

```python
#!/usr/bin/env pvpython

import sys, os, glob, shlex, subprocess
from paraview import servermanager

def GetProxiesToDelete() :
  iter = servermanager.vtkSMProxyIterator()
  ...
```

198
iter.Begin()
tobedeleted = []

while not iter.IsAtEnd():
    if iter.GetGroup().find("prototypes") != -1:
        iter.Next()
        continue

    proxy = servermanager.getPyProxy(iter.GetProxy())
    proxygroup = iter.GetGroup()
    iter.Next()
    if proxygroup != 'timekeeper' and proxy != None and
       proxygroup.find("pq_helper_proxies") == -1:
        tobedeleted.append(proxy)

return tobedeleted

def findTimeSteps():
    timeSteps = []
    for f in glob.glob('fields -*-0000.vtu'):
        timeSteps.append(f.split('-')[1])
    timeSteps.sort()
    return timeSteps

def genTempStateFile(step, baseStateFile, tmpFile):
    solidFile = "fields " + step + "+step+.pvtu"
    f = open(baseStateFile, 'r')
    lines = f.readlines()
    g = open(tmpFile, 'w')
    for l in lines:
        nl = l.replace('PATH_TAG', solidFile)
if __name__ == "__main__":

    import string
    print 'hello'

    from optparse import OptionParser, make_option
    option_list = [
        make_option("-p", "--pvstate", action="store", dest="stateFile", help="name of the state file to load", default="tmp.pvsm"),
        make_option("-s", "--steps", action="store", dest="steps", help="steps given in list form [0000000,0001000,...]", default=""),
        make_option("-n", "--base-name", action="store", dest="base-name", help="base name for output file", default="image")
    ]

    print option_list

    parser = OptionParser(option_list=option_list)
    (options, args) = parser.parse_args()
    userSteps = options.steps
    baseStateFile = options.stateFile
    baseOutputName = options.base_name
# find out all timesteps available

timeSteps = findTimeSteps()

# print timeSteps

# see if user picked specific steps

if options.steps:
    aux = userSteps.split(',

    # check that user steps exist in simulation

    ok = True

    uTimeSteps = []

    for step in aux:

        if not step in timeSteps:

            ok = False

            print "User selected time step: ", step, ",

            not in simulation data"

        else:

            uTimeSteps.append(step)

            print "User selected time step: ", step, 

            found in simulation, added to list"

    # go ahead and replace available time steps with user

    selection

    if ok:
        timeSteps = uTimeSteps

    else:

        print "User selected time steps not in simulation

        data"

        print "Available time steps: ", timeSteps[0], ",", 

        timeSteps[1], ",...,", timeSteps[-1], ",",

        stopping here"

    exit(0)
print "Will work on steps: ", timeSteps

for ts in timeSteps:
    print 'Time step: ', ts
    stateFile = 'tmp.pvsm'
    genTempStateFile(ts, baseStateFile, stateFile)

    servermanager.Connect()
    servermanager.LoadState(stateFile)

    view=servermanager.GetRenderView()
    view.StillRender()

    imageFileName = baseOutputName+'-'+ts+''.png"
    view.WriteImage(imageFileName, "vtkPNGWriter",1)
    servermanager.Disconnect()

print 'Done'

Since the rendering is done using a python script, this process was easily generalized to process multiple time steps and multiple ParaView scenes within the same batch rendering run. Furthermore, utilizing the fact that the pvbatch utility natively supports parallel rendering with MPI, we were able to do the rendering jobs directly on the ARL and ERDC computing clusters using as many processors as was necessary to compensate for the large file sizes. The ability to do parallel batch rendering not only provided a significant speed-up, but it also completely eliminated the need for transferring the files over a network, a process which becomes extremely cumbersome when both the number of files and the file sizes become large. Below we provide a sample bash script used on the ARL Harold machine to launch a parallel rendering job which visualizes five time steps of output data, producing three ParaView scenes
for each time step. This particular rendering job used 64 processors.

```bash
#!/bin/bash
#PBS -A ARONC00723032
#PBS -q standard
#PBS -l select=8:ncpus=8:mpiprocs=8
#PBS -l walltime=6:00:00
#PBS -N vmps.8_100
#PBS -j oe
#PBS -M aseagravamit@mit.edu
#PBS -m bae

cd $WORKDIR/elastic_runs/mup8_100_1d5
cp $PBS_O_WORKDIR/pvsnapshots.py .
cp $PBS_O_WORKDIR/mps_mpd.*.pvsm .

. /usr/cta/modules/$MODULE_VERSION/init/bash
module load paraview/3.14.0
module load mpi/openmpi-1.4.4

declare -a steps=(0005 0010 0015 0020 0025)

for i in "${steps[@]}"
do
    mpirun -np 64 pvbatch --use-offscreen --rendering
    pvsnapshots.py -p mps_mpd_ztf.pvsm -n 8.100_mps_ztf -s $i
    mpirun -np 64 pvbatch --use-offscreen --rendering
    pvsnapshots.py -p mps_mpd_zbf.pvsm -n 8.100_mps_zbf -s $i
```

203
mpirun -np 64 pvbatch --use-offscreen --rendering
    pvsnapshots.py -p mps_mpd_x.pvsm -n 8_100.mps_x -s $i
done

clear

cmpvshots.py

clear

rm mps_mpd_*.pvsm

mv *.png $PBS_O_WORKDIR

exit
Appendix E

Script for Filtering Output Data

In this appendix we provide the python source code developed to filter the fractured interface elements from a set of Tecplot output files and create a new set of Tecplot outputs containing only the filtered data.

```python
import csv, sys, os, glob, re

def parsedata(filename, separator, name_out):
    with file(filename, 'rb') as file_obj:
        csvobj = csv.reader(file_obj, delimiter=separator, skipinitialspace=True)

        # Ignore first three header lines
        line1 = csvobj.next()
        line2 = csvobj.next()
        line3 = csvobj.next()

        # Parse header line to extract the number of nodes
        nodesblock = line3[2]
        nodestemp1 = nodesblock.split('=')
        nodestemp2 = nodestemp1[1]
        nodestemp3 = nodestemp2.split(',')
        nodes = int(nodestemp3[0])

        # Parse header line to extract the number of elements
```
elementsblock = line3[3]
elemtemp1 = elementsblock.split(’=’)
elemtemp2 = elemtemp1[1]
elemtemp3 = elemtemp2.split(‘,’)
elements = int(elemtemp3[0])
# Read and store the old nodal coordinates and nodal
# fields
oldnodedata = []
for i in range(nodes):
    nodedata_i = csvobj.next()
    for j in range(11):
        oldnodedata.append(nodedata_i[j])
newconn = []
newnodedata = []
newnodelist = []
# Read in the old element connectivity and the get
# the damage for each node of each interface element
# Keep only elements with partially, or fully
# damaged nodes
for e in range(elements):
    conn_loc = []
    D_loc = []
    elemdata_i = csvobj.next()

    # Get connectivity and nodal damage for this
    # interface element
    for i in range(3):
        conn_loc.append(int(elemdata_i[i]) - 1)
        D_loc.append(oldnodedata[11*conn_loc[i]+4])
    # Test to see if any node is damaged
if float(D_loc[0]) > 0 or float(D_loc[1]) > 0 or float(D_loc[2]) > 0:
    # Test to see if this node has been
    # previously added to the filtered node list
for ci in range(3):
    found = 0
    found_loc = 0
    for counter in range(len(newnodelist)):
        if conn_loc[ci] == int(newnodelist[counter]):
            found = 1
            found_loc = counter+1
    # If it hasn't been added yet, add the
    # node now
    if found == 0:
        found_loc = len(newnodelist)
        newnodelist.append(conn_loc[ci])
        for j in range(11):
            newnodedata.append(olddata[11*conn_loc[ci]+j])
        # Construct new element connectivity to
        # reflect the
        # updated location of each node in the
        # filtered nodes list
        newconn.append(found_loc)

# DONE THE FILTERING PROCESS!

# Extract the new number of nodes after filtering
newnodes = len(newnodedata)/11
# Extract the new number of elements after filtering
newelements = len(newconn)/3

# If the filtered dataset is not empty, create a new tecplot output with only the filtered data
if float(newnodes) > 0:
    # Open the output file
    fileout = open(name_out,"w")

    # Tecplot style header lines
    line = 'TITLE = "NYI"
    fileout.write(line)
    line = 'VARIABLES = x y z opening_x opening_y
               opening_z opening sliding_x sliding_y
               sliding_z sliding
    fileout.write(line)
    line = 'ZONE t="NYI", N='+str(newnodes)+', E='+str(newelements)+', F=FEPOINT, ET=TRIANGLE
    fileout.write(line)

    # Write the filtered nodal coordinates and nodal data
    for node in range(len(newnodedata)/11):
Write the connectivity of the filtered elements
for element in range(len(newconn)/3):
    line = str(newconn[3*element+0]+1)+" '+str(newconn[3*element+1]+1)+" '+str(newconn[3*element+2]+1)+"\n"
    fileout.write(line)
fileout.close()

if __name__ == '__main__':
    # Get input data from user
    timestepL = raw_input('Enter lower bound for time step: ')
    timestepU = raw_input('Enter upper bound for time step: ')
    # Locate all time steps and filter the desired range
    steps = []
    for steps in glob.glob('open-*-0000.tec '):
        base_file = steps.split('.')[0]
        timestepsplit = base_file.split('-')[1]
        timestep = int(re.sub("^0+","",timestepsplit))
        if timestep >= int(timestepL) and timestep <= int(timestepU):
            # This is a time step in the desired range so
            # find data from all procs
            basename_in = 'open-'+timestepsplit+'-*.*.tec '
            print 'Working on step '+timestepsplit
            for f in glob.glob(basename_in):
basename_file = f.split('.')[0]
procsplit = basename_file.split('-')[2]
print 'Extracting cracks for processor ' + procsplit
basename_out = 'cracks-' + timestepsplit + '-' + procsplit + '.tec'
print basename_out
parsedata(f, ' ', basename_out)
print 'COMPLETED SUCCESSFULLY'
Bibliography


