INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2010; **81**:1207–1241
Published online 1 September 2009 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/nme.2723

An adaptive spacetime discontinuous Galerkin method for cohesive models of elastodynamic fracture

Reza Abedi[1], Morgan A. Hawker[2,‡], Robert B. Haber[1,*,†] and Karel Matouš[3]

[1]*Department of Mechanical Science and Engineering, University of Illinois at Urbana-Champaign, 1206 West Green Street, Urbana, IL 61801, U.S.A.*
[2]*C&I Engineering, Louisville, KY 40218, U.S.A.*
[3]*Computational Science and Engineering, University of Illinois at Urbana-Champaign, 1304 West Springfield Ave., Urbana, IL 61801, U.S.A.*

SUMMARY

This paper describes an adaptive numerical framework for cohesive fracture models based on a spacetime discontinuous Galerkin (SDG) method for elastodynamics with elementwise momentum balance. Discontinuous basis functions and jump conditions written with respect to target traction values simplify the implementation of cohesive traction–separation laws in the SDG framework; no special cohesive elements or other algorithmic devices are required. We use unstructured spacetime grids in a h-adaptive implementation to adjust simultaneously the spatial and temporal resolutions. Two independent error indicators drive the adaptive refinement. One is a dissipation-based indicator that controls the accuracy of the solution in the bulk material; the second ensures the accuracy of the discrete rendering of the cohesive law. Applications of the SDG cohesive model to elastodynamic fracture demonstrate the effectiveness of the proposed method and reveal a new solution feature: an unexpected quasi-singular structure in the velocity response. Numerical examples demonstrate the use of adaptive analysis methods in resolving this structure, as well as its importance in reliable predictions of fracture kinetics. Copyright © 2009 John Wiley & Sons, Ltd.

Received 11 June 2008; Revised 15 June 2009; Accepted 7 July 2009

KEY WORDS: cohesive model; fracture mechanics; discontinuous Galerkin; spacetime; finite element; elastodynamics, adaptive remeshing

*Correspondence to: Robert B. Haber, Department of Mechanical Science and Engineering, University of Illinois at Urbana-Champaign, 1206 West Green Street, Urbana, IL 61801, U.S.A.
†E-mail: r-haber@illinois.edu
‡Formerly, Department of Theoretical and Applied Mechanics, University of Illinois at Urbana-Champaign, IL, U.S.A.

Contract/grant sponsor: NSF; contract/grant number: ITR/AP DMR 01-21695
Contract/grant sponsor: Center for Process Simulation and Design (CPSD); contract/grant number: ITR/AP DMR 03-25939
Contract/grant sponsor: Center for Simulation of Advanced Rockets (CSAR)
Contract/grant sponsor: U.S. Department of Energy; contract/grant number: B341494

1. INTRODUCTION

This paper presents a framework for implementing cohesive models of elastodynamic fracture within the spacetime discontinuous Galerkin (SDG) finite element method. This section reviews the use of cohesive models in computational fracture mechanics and summarizes the requirements for a successful numerical implementation. We emphasize the suitability and effectiveness of adaptive SDG methods in achieving high-resolution realizations of cohesive traction–separation laws (TSLs). Indeed, the implementation described in this work delivers extremely accurate numerical solutions that led to the discovery of quasi-singular velocity response in cohesive elastodynamic fracture.

1.1. Related work

Dynamic material failure is important in a number of scientific and engineering applications, including earthquake modeling [1] semiconductor chip manufacturing processes [2], impact and fragmentation studies [3], armor engineering [4], delamination of composite materials [5], and bone fracture [6]. A variety of numerical methods for modeling dynamic fracture have been proposed. For example, damage parameters describe the degradation of the bulk material's strength and stiffness in *smeared cracking* and *damage* models [7, 8]. *Molecular dynamics* (MD) simulations [9–12] have reproduced many features of experimental observations of crack propagation [13]. However, the high cost of MD simulations limits their application to length scales on the order of micrometers and time scales on the order of nanoseconds; hence; they are not useful for many scientific and engineering studies. *Lattice dynamics* models [14] also predict certain crack patterns observed in experiments, but are primarily intended for qualitative studies rather than simulations of specific material microstructures. They describe material as a collection of discrete point masses arranged in a lattice pattern and connected by linear springs that can fail suddenly in a brittle fashion. *Nodal release* finite element models [15, 16] represent crack growth by duplicating nodes near the crack tip and allowing them to move independently. Thus, the crack opens along the common face between two adjacent elements, resulting in discrete increments of crack advance. Non-physical artifacts and instabilities arising from this discontinuous representation of crack extension can be mitigated by gradually releasing the nodal forces at duplicated nodes. However, it is difficult to link these models to the microstructural properties of materials, and they are not currently in wide use for dynamic fracture mechanics.

Cohesive models are among the most effective, and are currently the most popular, class of continuum numerical models for dynamic fracture. They developed from the cohesive zone models first introduced by Dugdale [17] and Barenblatt [18]. In contrast to smeared damage models, cohesive fracture models represent crack surfaces as sharp material interfaces that can be implemented either between adjacent finite elements, as in [19], or within elements, as in the extended finite element method [20–23] and the generalized finite element method [24, 25]. Cohesive models simulate crack initiation and extension by modeling the macroscopic effects of various non-linear damage processes in the neighborhood of the crack tip. A constitutive relation, called a TSL, describes the tractions acting across a cohesive interface as non-linear, bounded functions of the interface separation. The TSL eliminates the crack-tip stress singularities that arise in classical fracture mechanics [26] and introduces a microscopic material length scale that is essential to continuum fracture models.

TSLs appear in various forms, such as piecewise linear [5, 27], trapezoidal [27], rigid-linear [28], polynomial [29], or exponential [19, 30, 31]. Recently, Matouš and co-workers introduced

DOI: 10.1002/nme

a multiscale cohesive model based on microscopic features and physics [32]. Brittle models for elastodynamic fracture are the focus of this paper, but cohesive models have also been proposed for other types of materials and physical settings, such as ductile fracture and shear band formation [33, 34]. Rate-dependent cohesive models [35, 36] address strain-rate hardening effects that have been observed experimentally [37]. Costanzo and Walton [35] include thermomechanical coupling effects in a cohesive model that attempts to capture the effects of the rapid rise in temperature induced by high strain rates in the vicinity of propagating crack tips, as observed experimentally by, for example, Zehnder and Rosakis [38]. Cohesive models that account for geometric nonlinearities are appropriate when large deformation gradients are present [31, 39], as is generally the case in the crack-tip region.

1.2. Implementation of cohesive models within the SDG method

There are two primary requirements that any numerical implementation of a cohesive fracture model must satisfy. First, the kinematic model must admit jumps in the displacement field across cohesive interfaces, and second, some means must be available to enforce the traction–separation relation. Secondary considerations that might favor one method over another include numerical efficiency and scalability, balance properties for momentum and energy, adaptive capabilities and ease of implementation.

This paper presents a scheme for implementing cohesive models within an adaptive SDG finite element method [40–42]. The SDG method is implemented on fully unstructured spacetime meshes with basis functions that are discontinuous across all spacetime element boundaries. The discrete SDG model enforces momentum balance to within machine precision over every spacetime element with respect to the appropriate Riemann fluxes on the inter-element boundaries. Although the method is dissipative, adaptive mesh refinement effectively limits the relatively mild numerical energy losses.

When implemented on suitable spacetime grids that satisfy a *causality constraint* [43], the SDG formulation supports a patch-by-patch,[§] advancing-front solution scheme with $O(N)$ computational complexity, where N is the number of elements in the spacetime mesh. The causal spacetime solution scheme has a coarse-grained parallel structure that facilitates scalable, *asynchronous* parallel implementations that incur relatively small communications overhead. The patchwise solution method also facilitates adaptive spacetime meshing algorithms wherein mesh refinement and coarsening operations are implemented locally on individual elements or patches [44]. The mesh adapts simultaneously in space and time; hence, smaller effective time steps are used where the spatial resolution is high, and larger effective time steps are used in elements with larger spatial diameters. Notably, there are no global constraints on the time-step size that would limit the method's efficiency. Thus, the adaptive SDG method efficiently captures shocks and other moving solution features, such as the near-tip fields of growing cracks, by concentrating mesh refinement along the spacetime trajectories of the moving fronts.

The discontinuous nature of the SDG formulation provides a framework that naturally incorporates cohesive models. Displacement discontinuities are admissible between elements, so there is no need for cohesive interface elements or other special algorithmic devices. The only modifications

[§] A patch is a small collection of contiguous finite elements, cf. Section 3.1.

DOI: 10.1002/nme

needed to introduce a cohesive interface in the SDG framework are to relax the weak enforcement of the kinematic compatibility conditions and to use the TSL to compute the target momentum fluxes (tractions) in place of the Riemann values. These modifications are easily accomplished, as described in 2.4. In cases where the bulk material model is linearly elastic, nonlinearities due to the TSL arise only in those patches that intersect cohesive surfaces. Thus, the expense of a non–linear solution can be avoided in most patches.

An effective adaptive analysis strategy is essential to resolve accurately the TSL and to ensure numerical stability. Adaptive algorithms typically use *a posteriori* error indicators for the bulk response to control the mesh size. However, to date and to the authors' knowledge, no method in the literature directly addresses the accuracy of the discrete representation of the TSL. In this work, we use two adaptive error indicators: an energy-based error indicator that limits the numerical dissipation in each element to control the accuracy of the bulk solution and a second cohesive error indicator that limits the error in the fracture energy along cohesive interfaces. This combination provides a robust scheme for high-resolution dynamic simulations of cohesive fracture with excellent shock capturing capabilities. In this paper, adaptive meshing is used only to ensure solution accuracy. However, adaptive spacetime meshing can also be used to track crack paths that are not known *a priori*. This application of adaptive spacetime meshing will be addressed in a subsequent paper.

Adouani *et al.* [45] present a stable, adaptive method for propagating cracks within a *time-discontinuous Galerkin* framework. Their method also uses spacetime finite elements, but differs from the fully discontinuous method described here in that the basis functions only admit discontinuities across constant-time interfaces between slabs of elements. They use special interface elements to model displacement jumps in space across crack surfaces, and employ meshing and solution strategies that are distinct from the ones described here.

In the remainder of this paper, we extend the SDG formulation for elastodynamics to incorporate cohesive models in a manner that supports most TSLs. After discussing novel aspects of our adaptive implementation, we study the impact of the proposed adaptive error indicator for the cohesive tractions. Numerical examples demonstrate the ability of adaptive SDG models to generate very high-resolution solutions that respond dynamically to changes in the process zone size and that accurately portray the response within the cohesive process zone. In particular, we present new findings relating to quasi-singular velocity response in the vicinity of cohesive crack tips that propagate along weak interfaces embedded in a stronger matrix (e.g. low-strength, bimaterial interfaces).

2. FORMULATION

In this section, we develop an SDG finite element method for linearized elastodynamics that incorporates a cohesive model. In particular, we show that the SDG formulation for linear elastodynamics presented in [41] requires only a modest extension of the target flux definitions to accommodate the cohesive model. We use the notation for differential forms on spacetime manifolds introduced in [41] to summarize the spacetime theory. Although less familiar to some readers, this approach delivers a direct, coordinate-free notation that greatly simplifies the formulation on unstructured spacetime grids. In contrast to tensor notation, for example, the Stokes Theorem expressed in forms notation does not require unit vectors 'normal' to spacetime d-manifolds. Such objects are not well defined, given the absence of an inner product for spacetime vectors in classical

DOI: 10.1002/nme

mechanics. Expositions of differential forms and the exterior calculus on manifolds can be found in [46, 47]. We provide conventional component expansions of the SDG equations to clarify the formulation.

2.1. Spacetime analysis domain and differential forms notation

Let D be the reference spacetime domain, an open $(d+1)$-manifold in $\mathbb{E}^d \times \mathbb{R}$, where d is the spatial dimension. The coordinates $(x^1, \dots, x^d, t) = (\boldsymbol{x}, t)$ in D are defined with respect to the ordered basis $(\mathbf{e}_1, \dots, \mathbf{e}_d, \mathbf{e}_t)$ and are understood to be material coordinates associated with the undeformed configuration of a body followed by the time coordinate. The dual basis is denoted $(\mathbf{e}^1, \dots, \mathbf{e}^d, \mathbf{e}^t)$. From here on, we adopt the standard summation convention with Latin indices ranging from 1 to d. We employ differential forms with scalar, vector and tensor coefficients and follow the convention that symbols displayed in italic bold fonts denote forms, while symbols in upright bold fonts denote vector or tensor fields. Thus, $\boldsymbol{\sigma}$ and $\boldsymbol{\sigma}$ denote the differential form for stress and the stress tensor field, respectively, as explained below.

The standard basis for spacetime 1-forms is $\{\boldsymbol{d}x^1, \dots, \boldsymbol{d}x^d, \boldsymbol{d}t\}$. The *spacetime volume element* is the $(d+1)$-form given by $\Omega := \boldsymbol{d}x^1 \wedge \cdots \wedge \boldsymbol{d}x^d \wedge \boldsymbol{d}t$, where '$\wedge$' is the *exterior product operator* on forms; cf [47–49]. We have the standard basis for d-forms, $\{\star\boldsymbol{d}x^j, \star\boldsymbol{d}t\}$, in which $\star$ is the Hodge star operator and the indices of $\star\boldsymbol{d}x^j$ shall, from here on, be treated as subindices for purposes of the summation convention. These satisfy $\boldsymbol{d}x^i \wedge \star\boldsymbol{d}x^j = \delta^i_j \Omega, \boldsymbol{d}t \wedge \star\boldsymbol{d}x^j = \mathbf{0}, \boldsymbol{d}t \wedge \star\boldsymbol{d}t = \Omega$ and $\boldsymbol{d}x^i \wedge \star\boldsymbol{d}t = \mathbf{0}$. For the case $d=2$, we have $\Omega = \boldsymbol{d}x^1 \wedge \boldsymbol{d}x^2 \wedge \boldsymbol{d}t, \star\boldsymbol{d}x^1 = \boldsymbol{d}x^2 \wedge \boldsymbol{d}t, \star\boldsymbol{d}x^2 = -\boldsymbol{d}x^1 \wedge \boldsymbol{d}t$ and $\star\boldsymbol{d}t = \boldsymbol{d}x^1 \wedge \boldsymbol{d}x^2$. The *temporal insertion* is defined in terms of the standard insertion (contraction) operator as, $\mathbf{i} := \mathbf{i}_{\bar{\mathbf{e}}_t}$, in which $\bar{\mathbf{e}}_t$ is a vector field on D with uniform value $\mathbf{e}_t$. For $d=2$, we have $\mathbf{i}\star\boldsymbol{d}x^1 = -\boldsymbol{d}x^2, \mathbf{i}\star\boldsymbol{d}x^2 = \boldsymbol{d}x^1$ and $\mathbf{i}\star\boldsymbol{d}t = \mathbf{0}$.

Let $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ be r- and s-forms on D, respectively, let $\mathbf{a}$ and $\mathbf{b}$ be m- and n-tensor fields on D with $m \geqslant n$, and let w be a scalar field on D. We write $\mathbf{a}\boldsymbol{\alpha}$ and $\mathbf{b}\boldsymbol{\beta}$ to denote an r-form and an s-form with tensor coefficients of order m and n, respectively. The exterior product of $\mathbf{a}\boldsymbol{\alpha}$ and $\mathbf{b}\boldsymbol{\beta}$ is the $(r+s)$-form with tensor coefficients of order $m-n$ given by

$$\mathbf{a}\boldsymbol{\alpha} \wedge \mathbf{b}\boldsymbol{\beta} := \mathbf{a}(\mathbf{b})(\boldsymbol{\alpha} \wedge \boldsymbol{\beta}) \tag{1}$$

We introduce a special 1-form with vector coefficients, $\boldsymbol{d}\mathbf{x} := \mathbf{e}_i \boldsymbol{d}x^i$, and a corresponding d-form with co-vector coefficients, $\star\boldsymbol{d}\mathbf{x} := \mathbf{e}^i \star\boldsymbol{d}x^i$.

2.2. Continuum formulation

2.2.1. Mechanical fields. Let the ordered set $\mathscr{P} = \{Q_\alpha\}_{\alpha=1}^M$ be a partition of the spacetime domain D into M open subdomains with regular boundaries. Let $L^2(D)$ and $H^1(Q)$ be the Hilbertian Sobolev spaces of order 0 on D and order 1 on Q, respectively. We define a broken Sobolev space on $\mathscr{P}$, $\mathscr{V} := \{\boldsymbol{w} \in L^2(D) : \boldsymbol{w}|_{Q_\alpha} \in H^1(Q_\alpha), \alpha = 1, \dots, M\}$, in which $\boldsymbol{w}$ is a vector field on D (i.e. a 0-form with vector coefficients), and note that $\mathscr{V}$ admits vector fields with jumps between adjacent subdomains.

Next, we define the kinematic fields for the SDG formulation. Let $\boldsymbol{u} \in \mathscr{V}$ denote the *displacement field on D*, a 0-form with vector coefficients (i.e. a vector field) with the component representation, $\boldsymbol{u} = \mathbf{u} = u_i \mathbf{e}^i$. The *velocity* $\boldsymbol{v}$ and the *linearized strain* $\boldsymbol{E}$ are 1-forms on D with co-vector coefficients given by $\boldsymbol{v} := \mathbf{v}\boldsymbol{d}t = \dot{\mathbf{u}}\boldsymbol{d}t = \dot{u}_i \mathbf{e}^i \boldsymbol{d}t$ and $\boldsymbol{E} := \mathbf{E} \wedge \boldsymbol{d}\mathbf{x} = E_{ij} \mathbf{e}^i \otimes \mathbf{e}^j \wedge \mathbf{e}_k \boldsymbol{d}x^k = E_{ij} \mathbf{e}^i \boldsymbol{d}x^j$, where $E_{ij} := \frac{1}{2}(u_{i,j} + u_{j,i})$. The *strain–velocity* is the 1-form defined by $\boldsymbol{\varepsilon} := \boldsymbol{E} + \boldsymbol{v}$.

DOI: 10.1002/nme

The force-like fields are d-forms on D with vector coefficients. We introduce the *stress*,

$$\boldsymbol{\sigma}=\boldsymbol{\sigma}\wedge\star\boldsymbol{dx}=\sigma^{ij}\mathbf{e}_i\otimes\mathbf{e}_j\wedge\mathbf{e}^k\star\boldsymbol{d}x^k=\sigma^{ij}\mathbf{e}_i\star\boldsymbol{d}x^j \tag{2}$$

and the *linear momentum density*,

$$\boldsymbol{p}=\mathbf{p}\star\boldsymbol{dt}=\rho\dot{u}^i\mathbf{e}_i\star\boldsymbol{dt} \tag{3}$$

These combine to form the *spacetime linear momentum flux*,

$$\boldsymbol{M}=\boldsymbol{\sigma}-\boldsymbol{p} \tag{4}$$

The *body force density* is the $(d+1)$-form given by

$$\mathbf{b}=\mathbf{b}\Omega=b^i\mathbf{e}_i\Omega \tag{5}$$

In the above expressions, $\mathbf{p}$ and $\mathbf{b}$ denote the vector fields for linear momentum density and body force density on D, and $\boldsymbol{\sigma}$ denotes the stress tensor field on D.

2.2.2. Kinematic compatibility. Kinematic compatibility requires the traces of the displacement field, $\boldsymbol{u}^+$ and $\boldsymbol{u}^-$, from opposing sides of any spacetime d-manifold $\Gamma\subset\overline{D}$ to match. That is,

$$(\boldsymbol{u}^+-\boldsymbol{u}^-)|_\Gamma=\mathbf{0} \tag{6}$$

In lieu of (6), we enforce an equivalent system of jump conditions [40] on the boundaries of all subdomains $Q\subset D$:

$$(\boldsymbol{\varepsilon}^*-\boldsymbol{\varepsilon})|_{\partial Q}=\mathbf{0} \tag{7a}$$

$$(\boldsymbol{u}_0^*-\boldsymbol{u}_0)|_{\partial Q}=\mathbf{0} \tag{7b}$$

in which a subscript '0' indicates a local projection of the subscripted quantity onto $\mathscr{V}_0^Q:=\{\boldsymbol{w}\in H^1(Q):\boldsymbol{\varepsilon}(\boldsymbol{w})=\mathbf{0}\}$, that is, the zero-energy subspace of $\mathscr{V}|_Q$ characterized by vanishing velocity and strain [41]. A superscript '$*$' denotes a physically consistent target value that is uniquely defined on any d-manifold embedded in $\overline{D}$ by prescribed boundary or initial data, or by the solution to a local Riemann problem; cf. [41]. Specific definitions for the target values are presented in Section 2.2.5.

2.2.3. Momentum and energy balance. Balance of linear momentum requires that

$$\int_{\partial Q}\boldsymbol{M}+\int_Q\rho\boldsymbol{b}=\mathbf{0}\quad\forall Q\subset D \tag{8}$$

in which ρ is the mass density per unit volume in the reference configuration. Equation (8) also implies balance of angular momentum when the stress tensor $\boldsymbol{\sigma}$ is symmetric [41].

Let Γ^{J} be the jump set of $\boldsymbol{M}$ on D. The system

$$(\boldsymbol{dM}+\rho\boldsymbol{b})|_{Q\setminus\Gamma^{\mathrm{J}}}=\mathbf{0}\quad\forall Q\subset D \tag{9a}$$

$$(\boldsymbol{M}^*-\boldsymbol{M})|_{\overline{Q}\cap(\Gamma^{\mathrm{J}}\cup\partial Q)}=\mathbf{0}\quad\forall Q\subset D \tag{9b}$$

DOI: 10.1002/nme

enforces (8) via the Stokes theorem while accounting for possible jumps in $\boldsymbol{M}$. It also enforces compatibility with a target momentum flux $\boldsymbol{M}^*$ that is defined uniquely on every d-manifold embedded in $\overline{D}$; cf. 2.2.5. The component form of (9a) is $[\sigma^{ij}_{,j}+\rho(b^i-\ddot{u}^i)]\mathbf{e}_i\Omega=\mathbf{0}$ on $Q\setminus\Gamma^{\mathrm{J}}$. Thus, (9a) is the familiar equation of motion.

Energy balance requires

$$\int_{\partial Q}\boldsymbol{N}+\int_{Q}\dot{\boldsymbol{u}}\wedge\rho\boldsymbol{b}=\mathbf{0}\quad\forall Q\subset D \tag{10}$$

where the *spacetime flux of total energy* $\boldsymbol{N}$ is the d-form with scalar coefficients given by $\boldsymbol{N}=-\mathbf{i}\Phi+\dot{\boldsymbol{u}}\wedge\boldsymbol{\sigma}$, in which the *total energy density*, $\Phi:=(U_0+T_0)\Omega$, is a $(d+1)$-form whose scalar coefficient is the sum of the internal and kinetic energy densities, U_0 and T_0.

2.2.4. Constitutive model. From here on, we assume linear elastodynamic response. That is, we assume there exists a spacetime total energy density field of the form

$$\Phi(\boldsymbol{\varepsilon})=\tfrac{1}{2}\boldsymbol{\varepsilon}\wedge\mathbf{C}\boldsymbol{\varepsilon}=\tfrac{1}{2}(E_{ij}C^{ijkl}E_{kl}+\rho v_i\delta^{ij}v_j)\Omega \tag{11}$$

where $\mathbf{C}$ is a linear transformation of 1-forms into d-forms such that

$$\boldsymbol{\sigma}+\boldsymbol{p}=\frac{\partial\Phi}{\partial\boldsymbol{\varepsilon}}\Rightarrow\boldsymbol{\sigma}+\boldsymbol{p}=\mathbf{C}\boldsymbol{\varepsilon} \tag{12a}$$

$$\boldsymbol{\sigma}=\frac{\partial\Phi}{\partial\boldsymbol{E}}\Rightarrow\boldsymbol{\sigma}=\mathbf{C}\boldsymbol{E}\ \text{ or }\ \sigma^{ij}=C^{ijkl}E_{kl} \tag{12b}$$

$$\boldsymbol{p}=\frac{\partial\Phi}{\partial\boldsymbol{v}}\Rightarrow\boldsymbol{p}=\mathbf{C}\boldsymbol{v}\ \text{ or }\ p^i=\rho\delta^{ij}v_j \tag{12c}$$

in which δ^{ij} is the Kronecker delta and the components C^{ijkl} of the positive-definite elasticity tensor exhibit the usual major and minor symmetries—a structure that imposes symmetry on the coefficients of $\boldsymbol{\sigma}$. The momentum flux is related to the strain–velocity via the linear transformation

$$\boldsymbol{M}=\hat{\mathbf{C}}\boldsymbol{\varepsilon}:=\mathbf{C}(\boldsymbol{E}-\boldsymbol{v}) \tag{13}$$

and the spacetime flux of total energy for linear elastodynamic response reduces to

$$\boldsymbol{N}=\tfrac{1}{2}(\dot{\boldsymbol{u}}\wedge\boldsymbol{M}+\boldsymbol{\varepsilon}\wedge\mathbf{i}\boldsymbol{\sigma}) \tag{14}$$

2.2.5. Boundary and initial conditions. Consider a subdomain $Q\subset D$. The target values $\boldsymbol{u}_0^*,\boldsymbol{\varepsilon}^*$, and $\boldsymbol{M}^*$ in (7a), (7b) and (9b) provide a unified mechanism for enforcing boundary conditions consistent with prescribed boundary and initial data on $\partial Q\cap\partial D$ or with causality on $\partial Q\setminus\partial D$. We use undecorated symbols and symbols decorated with a superscript '+' to denote, respectively, the interior and exterior traces of a quantity on ∂Q. A superscript 'R' indicates a value on ∂Q that solves a local Riemann problem, as described in [41].[¶] An underlined symbol denotes prescribed

[¶]These values are called *Godunov values* in [41], but the nomenclature, *Riemann values*, adopted here is more appropriate.

DOI: 10.1002/nme

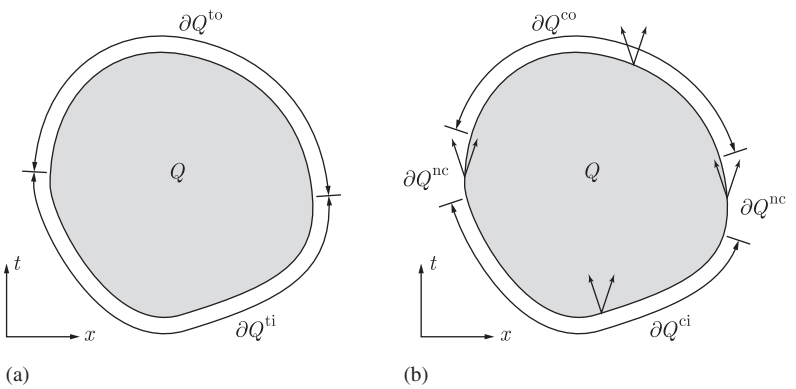

Figure 1. Alternative partitions of ∂Q. (a) temporal partition and (b) causal partition.

initial or boundary data on ∂D. Next, we introduce two partitions that determine how the target value is computed on different parts of ∂Q.

The *temporal partition of* ∂Q helps to determine the target value $\boldsymbol{u}_0^*$ on ∂Q (see Figure 1(a)). We define the *temporal inflow boundary of* Q as $\partial Q^{\mathrm{ti}} := \{\mathbf{x} \in \partial Q : \mathbf{i}\Omega(\mathbf{a}_{\partial Q}(\mathbf{x})) < 0\}$ in which the volume vector $\mathbf{a}_{\partial Q}$ encodes the local orientation of ∂Q. The *temporal outflow boundary* is $\partial Q^{\mathrm{to}} = \partial Q \setminus \partial Q^{\mathrm{ti}}$. The earlier value of $\mathbf{u}_0$ always determines $\boldsymbol{u}_0^*$. Thus, $\boldsymbol{u}_0^* = \boldsymbol{u}_0$ on ∂Q^{to} (so (7b) is trivially satisfied), and $\boldsymbol{u}_0^*$ is determined on ∂Q^{ti} either by outflow values from an earlier subdomain or by prescribed initial data:

$$\boldsymbol{u}_0^*|_{\partial Q^{\mathrm{ti}}} = \begin{cases} \boldsymbol{u}_0^+ & \text{on } \partial Q^{\mathrm{ti}} \setminus \partial D^{\mathrm{ti}} \\ \bar{\boldsymbol{u}}_0 & \text{on } \partial Q^{\mathrm{ti}} \cap \partial D^{\mathrm{ti}} \end{cases} \tag{15}$$

We use a separate *causal partition of* ∂Q to determine the target values for $\boldsymbol{\varepsilon}^*$ and $\boldsymbol{M}^*$ based on prescribed initial/boundary data or on the characteristics associated with elastic wave propagation (see Figure 1(b)). We say that a spacetime d-manifold is *causal* if and only if it has a zero intersection with the union of the dynamic domains of influence of all its points. Causality implies that all characteristic directions at all points on a causal subdomain of ∂Q are either all outward or all inward relative to Q. We partition ∂Q into its causal and noncausal parts, ∂Q^{c} and ∂Q^{nc}, and we further partition ∂Q^{c} into a *causal inflow* part ∂Q^{ci} and a *causal outflow* part ∂Q^{co} according to whether the local characteristic directions are all inward or all outward relative to Q (see Figure 1(b)).

In general, the target values $(\boldsymbol{M}^*, \boldsymbol{\varepsilon}^*)$ depend on the interior and exterior traces on ∂Q of $\boldsymbol{M}$ and $\boldsymbol{\varepsilon}$, as well as on the local orientation of ∂Q. The earlier limiting values control the target values on ∂Q^{c}, while Riemann values control on noncausal interfaces on the interior of D. In the latter case, we write $(\boldsymbol{M}^*, \boldsymbol{\varepsilon}^*) = (\boldsymbol{M}^R(\boldsymbol{M}, \boldsymbol{M}^+, \mathbf{a}_{\partial Q}), \boldsymbol{\varepsilon}^R(\boldsymbol{\varepsilon}, \boldsymbol{\varepsilon}^+, \mathbf{a}_{\partial Q}))$, in which the volume vector $\mathbf{a}_{\partial Q}$ encodes the local orientation of ∂Q. For simplicity, we assume either pure Dirichlet or pure Neumann conditions at each point on the noncausal domain boundary. That is, we partition the noncausal domain boundary, ∂D^{nc}, into disjoint Neumann and Dirichlet parts. A prescribed momentum flux $\bar{\boldsymbol{M}}$ determines $\boldsymbol{M}^*$ on the Neumann part, $\partial D^{\boldsymbol{M}}$, where $\boldsymbol{\varepsilon}$ is unconstrained. A prescribed strain–velocity $\bar{\boldsymbol{\varepsilon}}$ determines $\boldsymbol{\varepsilon}^*$ on the Dirichlet part, $\partial D^{\boldsymbol{\varepsilon}}$, where $\boldsymbol{M}$ is unconstrained. For each $Q \subset D$,

DOI: 10.1002/nme

we write

$$\boldsymbol{M}^* = \begin{cases} \boldsymbol{M} & \text{on } \partial Q^{\text{co}} \cup (\partial Q \cap \partial D^{\boldsymbol{\varepsilon}}) \\ \boldsymbol{M}^+ & \text{on } \partial Q^{\text{ci}} \setminus \partial D^{\text{ci}} \\ \boldsymbol{M}^{\text{R}}(\boldsymbol{M}, \boldsymbol{M}^+, \mathbf{a}_{\partial Q}) & \text{on } \partial Q^{\text{nc}} \setminus \partial D^{\text{nc}} \\ \bar{\boldsymbol{M}} & \text{on } \partial Q \cap (\partial D^{\text{ci}} \cup \partial D^{\boldsymbol{M}}) \end{cases} \tag{16}$$

$$\boldsymbol{\varepsilon}^* = \begin{cases} \boldsymbol{\varepsilon} & \text{on } \partial Q^{\text{co}} \cup (\partial Q \cap \partial D^{\boldsymbol{M}}) \\ \boldsymbol{\varepsilon}^+ & \text{on } \partial Q^{\text{ci}} \setminus \partial D^{\text{ci}} \\ \boldsymbol{\varepsilon}^{\text{R}}(\boldsymbol{\varepsilon}, \boldsymbol{\varepsilon}^+, \mathbf{a}_{\partial Q}) & \text{on } \partial Q^{\text{nc}} \setminus \partial D^{\text{nc}} \\ \bar{\boldsymbol{\varepsilon}} & \text{on } \partial Q \cap (\partial D^{\text{ci}} \cup \partial D^{\boldsymbol{\varepsilon}}) \end{cases} \tag{17}$$

Detailed expressions for the Riemann values are provided in [41]. They provide physically and mathematically correct values on material interfaces $\partial Q^{\text{nc}} \setminus \partial D^{\text{nc}}$ that hold whether or not a shock is present.

2.2.6. Continuum problem statement. Our SDG finite element method derives from a continuum problem in which the displacement field $\mathbf{u}$ is the only independent unknown solution field. Let the kinematic relations between $\mathbf{u}$ and $\boldsymbol{\varepsilon}$ (cf. Section 2.2.2) and the spacetime constitutive relation (13) be strongly enforced, and let $\boldsymbol{u}_0^*, \boldsymbol{M}^*$ and $\boldsymbol{\varepsilon}^*$ be defined as in (15), (16) and (17). Then we seek $\boldsymbol{u} \in \mathscr{V}$ such that the jump conditions for kinematic compatibility (7) and the equation of motion (9) are satisfied. Satisfaction of (7) and (9) implies energy balance in the continuum setting, hence, we do not explicitly enforce (10).

2.3. Discrete formulation

We obtain a discontinuous Galerkin finite element method from the continuum problem by associating the partition $\mathscr{P}$ with a spacetime finite element mesh and by considering a finite-dimensional subspace,

$$\mathscr{V}_h = \{\mathbf{w} \in \mathscr{V} : \mathbf{w}|_Q \in \mathscr{V}_h^Q \ \forall Q \in \mathscr{P}\}$$

in which $\mathscr{V}_h^Q = \mathbf{P}^{k_Q}(Q)$, where $\mathbf{P}^{k_Q}(Q)$ is the space of vector fields on element Q whose components are complete polynomials of order k_Q. The polynomial order k_Q can be adjusted on a per-element basis in a hp-adaptive scheme. However, this option is not exercised in the h-adaptive results reported in Section 4. The same relations are enforced strongly as in the continuum problem, while the following discrete-weighted residual statement weakly enforces (7) and (9). Find $\boldsymbol{u}_h \in \mathscr{V}_h$ such that, for all $Q \in \mathscr{P}$,

$$\int_Q \dot{\boldsymbol{w}} \wedge (\mathbf{d}\boldsymbol{M}_h + \rho \boldsymbol{b}) + \int_{\partial Q} \{\dot{\boldsymbol{w}} \wedge (\boldsymbol{M}_h^* - \boldsymbol{M}_h) + (\boldsymbol{\varepsilon}_h^* - \boldsymbol{\varepsilon}_h) \wedge \mathbf{i}\hat{\boldsymbol{M}}\}$$

$$+ \int_{\partial Q^{\text{ti}}} \kappa \boldsymbol{w}_0 \wedge (\boldsymbol{u}_h^* - \boldsymbol{u}_h) \star \boldsymbol{dt} = 0 \quad \forall \boldsymbol{w} \in \mathscr{V}_h^Q \tag{18}$$

DOI: 10.1002/nme

in which a subscript 'h' denotes an explicit function of $\boldsymbol{u}_h$ and its first-order partial derivatives, and κ is a constant introduced for dimensional consistency [41]. The Stokes theorem applied to (18) leads to the discrete weak form that defines our finite element method:

$$\int_Q (-\mathbf{d}\dot{\boldsymbol{w}} \wedge \boldsymbol{M}_h + \dot{\boldsymbol{w}} \wedge \rho \boldsymbol{b}) + \int_{\partial Q} \{\dot{\boldsymbol{w}} \wedge \boldsymbol{M}_h^* + (\boldsymbol{\varepsilon}_h^* - \boldsymbol{\varepsilon}_h) \wedge \mathbf{i}\hat{\boldsymbol{M}}\}$$

$$+ \int_{\partial Q^{\mathrm{ti}}} \kappa \boldsymbol{w}_0 \wedge (\boldsymbol{u}_h^* - \boldsymbol{u}_h) \star \boldsymbol{d}t = 0 \quad \forall \boldsymbol{w} \in \mathscr{V}_h^Q \tag{19}$$

The discrete solution to (19) exactly satisfies the integral forms of balance of linear momentum and balance of angular momentum over every spacetime element $Q \in \mathscr{P}$ [41].

To facilitate implementation, and for the benefit of readers who might not be familiar with differential forms notation, we present an alternative statement of (19) using conventional tensor component expressions. First, however, we must introduce a mapping of the spacetime from $\mathbb{E}^d \times \mathbb{R}$ to $\mathbb{E}^{d+1}$ to enable the construction of normal vector fields on spacetime d-manifolds. This mapping establishes an arbitrary metric on spacetime that is not intrinsic to elastodynamics. While it compromises the formal invariance of our formulation, it introduces no error. Specifically, we introduce an arbitrary reference velocity magnitude c and define a *pseudo-time*, $\tilde{t} = ct$, whose dimension is length. All spacetime fields are now written as functions of $(\boldsymbol{x}, \tilde{t})$, and all temporal derivatives are taken with respect to the pseudo-time $\tilde{t}$. The component expansion of Equation (19) is

$$\int_Q \{-\dot{w}_{i,j}\sigma_h^{ij} + \rho \ddot{w}_i \dot{u}_h^i + \rho \dot{w}_i b^i\} \mathrm{d}\mathbf{V} + \int_{\partial Q} \{\dot{w}_i (\sigma_h^{ij})^* n_j - \dot{w}_i (p_h^i)^* n_{\tilde{t}} - [\![E_{ij}]\!] \hat{\sigma}^{ij} n_{\tilde{t}} + [\![\dot{u}_i]\!] \hat{\sigma}^{ij} n_j\} \mathrm{d}\mathbf{S}$$

$$+ \int_{\partial Q^i} \kappa (w_0)_i [\![u^i]\!] n_{\tilde{t}} \mathrm{d}\mathbf{S} = 0 \quad \forall \mathbf{w} \in V^Q \tag{20}$$

where $[\![f]\!] := f^* - f$, $\mathrm{d}\mathbf{S}$ and $\mathrm{d}\mathbf{V}$ are spacetime surface and volume differentials in $\mathbb{E}^{d+1}$, and $\mathbf{n} = n_i \mathbf{e}^i + n_{\tilde{t}} \mathbf{e}^{\tilde{t}}$ is the outward spacetime unit normal vector field on ∂Q in which $\mathbf{e}^{\tilde{t}}$ is the orthonormal basis vector in the pseudo-time direction.

2.4. Extension of the SDG framework to incorporate a cohesive model

2.4.1. Modifications to the SDG model. In this section, we extend the SDG formulation to incorporate a cohesive model. There are three basic requirements to accomplish this objective: (i) identification of the set of material cohesive interfaces, (ii) a provision to admit displacement discontinuities across cohesive interfaces and (iii) a means to enforce a traction–separation relation across cohesive interfaces. We accomplish the first requirement by aligning the spacetime mesh with the trajectories of the cohesive interfaces; we use Γ^{C} to denote the collection of time-like d-manifolds that cover these trajectories. Since the SDG formulation already accommodates jumps across inter-element boundaries, the second requirement is accomplished simply by writing $\boldsymbol{\varepsilon}^* = \boldsymbol{\varepsilon}$ on Γ^{C} in (17) to relax the compatibility constraint. We address the third requirement by extending the definition of the target momentum flux in (16) to match the cohesive traction from the TSL on Γ^{C}.

We modify the disjoint partition of ∂D^{nc} to include cohesive interfaces: $\partial D^{\mathrm{nc}} = \overline{\partial D^{\boldsymbol{M}} \cup \partial D^{\boldsymbol{\varepsilon}} \cup \partial D^C}$, where $\partial D^{\mathrm{C}} = \partial D \cap \Gamma^{\mathrm{C}}$. We introduce ∂D^{C} to accommodate cohesive interfaces on the domain

DOI: 10.1002/nme

boundary that can arise in models with symmetry or periodic boundary conditions. The target values for $\boldsymbol{M}$ and $\boldsymbol{\varepsilon}$ are redefined as

$$\boldsymbol{M}^* = \begin{cases} \boldsymbol{M} & \text{on } \partial Q^{\text{co}} \cup (\partial Q \cap \partial D^{\boldsymbol{\varepsilon}}) \\ \boldsymbol{M}^+ & \text{on } \partial Q^{\text{ci}} \setminus \partial D^{\text{ci}} \\ \boldsymbol{M}^{\text{R}}(\boldsymbol{M}, \boldsymbol{M}^+, \mathbf{a}_{\partial Q}) & \text{on } \partial Q^{\text{nc}} \setminus (\partial D^{\text{nc}} \cup \varGamma^{\mathbf{C}}) \\ \bar{\boldsymbol{M}} & \text{on } \partial Q \cap (\partial D^{\text{ci}} \cup \partial D^{\boldsymbol{M}}) \\ \boldsymbol{M}^{\text{C}} & \text{on } \partial Q \cap \varGamma^{\mathbf{C}} \end{cases} \tag{21}$$

$$\boldsymbol{\varepsilon}^* = \begin{cases} \boldsymbol{\varepsilon} & \text{on } \partial Q^{\text{co}} \cup (\partial Q \cap (\partial D^{\boldsymbol{M}} \cup \varGamma^{\mathbf{C}})) \\ \boldsymbol{\varepsilon}^+ & \text{on } \partial Q^{\text{ci}} \setminus \partial D^{\text{ci}} \\ \boldsymbol{\varepsilon}^{\text{R}}(\boldsymbol{\varepsilon}, \boldsymbol{\varepsilon}^+, \mathbf{a}_{\partial Q}) & \text{on } \partial Q^{\text{nc}} \setminus (\partial D^{\text{nc}} \cup \varGamma^{\mathbf{C}}) \\ \bar{\boldsymbol{\varepsilon}} & \text{on } \partial Q \cap (\partial D^{\text{ci}} \cup \partial D^{\boldsymbol{\varepsilon}}) \\ \boldsymbol{\varepsilon}^{\text{C}} & \text{on } \partial Q \cap \varGamma^{\mathbf{C}} \end{cases} \tag{22}$$

in which $\boldsymbol{M}^{\text{C}}$ is a solution-dependent value determined by the traction–separation relation on $\varGamma^{\text{C}}$. Since all cohesive interfaces are material surfaces, $\varGamma^{\text{C}}$ is comprised entirely of vertical (i.e. time-parallel) d-manifolds. In this case, $\boldsymbol{\varepsilon}^{\text{C}}|_{\varGamma^{\text{C}}} = \boldsymbol{\nu}|_{\varGamma^{\text{C}}}$. By assigning the interior trace of $\boldsymbol{\nu}$ to $\boldsymbol{\varepsilon}^\star$, we relax the compatibility condition on cohesive interfaces. Let $\mathbf{e}_{\text{n}}$ be the spatial vector field with unit magnitude that is everywhere normal to $\varGamma^{\text{C}}$. Then the restriction of the spacetime momentum flux to $\varGamma^{\text{C}}$ simplifies to $\boldsymbol{M}|_{\varGamma^{\text{C}}} = \boldsymbol{\sigma}|_{\varGamma^{\text{C}}} = \mathbf{t} \star \boldsymbol{d}x^{\text{C}}$, in which $\mathbf{t}$ and $\star \boldsymbol{d}x^{\text{C}}$ are the surface traction and the spacetime d-volume element on $\varGamma^{\text{C}}$. The target momentum flux on $\varGamma^{\text{C}}$ is then fully defined by

$$\boldsymbol{M}^{\text{C}} = \mathbf{t}_{\text{c}} \wedge \star \boldsymbol{d}x^{\text{C}} \tag{23}$$

in which $\mathbf{t}_{\text{c}}$ is the cohesive traction vector generated by the traction–separation relation.

2.4.2. Cohesive traction–separation relation. In principle, any traction–separation relation can be implemented within the above SDG framework. However, for definiteness and to support the numerical examples in Section 4, here we restrict our attention to two spatial dimensions ($d = 2$) and to the history-independent, exponential relationship developed by Xu and Needleman to describe brittle, elastodynamic fracture [19]:

$$\mathbf{t}_{\text{c}} = \frac{\partial \phi}{\partial \boldsymbol{\Delta}} \tag{24}$$

in which the potential ϕ is the free energy density per unit area and $\boldsymbol{\Delta} = \Delta_{\text{n}} \mathbf{e}^{\text{n}} + \Delta_{\text{t}} \mathbf{e}^{\text{t}}$ is the jump in the displacement field across $\varGamma^{\text{C}}$, where $\mathbf{e}^{\text{n}}$ and $\mathbf{e}^{\text{t}}$ are the spatial vector fields with unit magnitude that are everywhere normal and tangent to $\partial Q \cap \varGamma^{\text{C}}$.

DOI: 10.1002/nme

Let ϕ_n and ϕ_t denote the work of separation in the normal and tangential directions, and let δ_n and δ_t be the critical separations at which the cohesive traction components, $\mathbf{t}^n$ and $\mathbf{t}^t$, attain their maximum values, σ_C and τ_C. Then, we follow [19, 30] in which the scalar potential is

$$\phi(\mathbf{\Delta}) = \phi_n + \phi_n \exp\left(-\frac{\Delta_n}{\delta_n}\right)\left\{\left[1-r+\frac{\Delta_n}{\delta_n}\right]\frac{1-q}{r-1} - \left[q+\left(\frac{r-q}{r-1}\right)\frac{\Delta_n}{\delta_n}\right]\exp\left(-\frac{{\Delta_t}^2}{{\delta_t}^2}\right)\right\} \tag{25}$$

where $q = \phi_t/\phi_n$ and $r = \Delta_n^*/\delta_n$, with Δ_n^* being the value of Δ_n after complete shear separation when $t_n = 0$. This specialization of the potential leads to $\phi_n = e\sigma_C\delta_n$ and $\phi_t = \sqrt{e/2}\tau_C\delta_t$.

As in [19], we set $r = 0$, $\tau_C = \sqrt{2e}\sigma_C$ and $\delta_n = \delta_t$, so that $\phi_n = \phi_t \Leftrightarrow q = 1$. Combining the above relations, we obtain expressions for the cohesive traction components:

$$t_n = \sigma_C \frac{\Delta_n}{\delta}\exp\left(1-\frac{\Delta_n}{\delta}-\frac{{\Delta_t}^2}{\delta^2}\right) \tag{26a}$$

$$t_t = 2\sigma_C \frac{\Delta_t}{\delta}\left(1+\frac{\Delta_n}{\delta}\right)\exp\left(1-\frac{\Delta_n}{\delta}-\frac{{\Delta_t}^2}{\delta^2}\right) \tag{26b}$$

in which $\delta := \delta_n = \delta_t$. The smooth structure of (26) facilitates numerical quadrature as well as the solution of the resulting non-linear finite element equations by, for example, a Newton–Raphson procedure.

3. IMPLEMENTATION

In contrast to methods that use a sequence of global time steps to advance the solution in time, we implement our SDG method on unstructured spacetime meshes that admit nonuniform refinement in both space and time. We construct our spacetime meshes subject to a *causality constraint* that enables a direct, patch-by-patch, advancing-front solution procedure that interleaves mesh generation with patch-level solution procedures. The use of this constraint leads to an algorithm, with linear computational complexity in the number of patches, that is well-suited to parallel and adaptive implementation. The following subsections describe the basic SDG meshing and solution procedure, the implementation of the cohesive model within the SDG framework, and a spacetime adaptive refinement strategy that simultaneously controls overall numerical dissipation as well as the error in fracture energy.

3.1. Spacetime meshing and advancing-front solution procedure

We say that a spacetime mesh is *patchwise causal* if and only if the elements in the mesh can be organized into groups of contiguous elements called *patches*, such that the boundary of each patch consists exclusively of causal manifolds (see Section 2.2.5). Inter-element boundaries within the patch can be noncausal; hence, the elements within the patch must be solved simultaneously. However, the dependency graph between patches defines a partial ordering wherein the solution on each patch depends only on boundary data and solutions on earlier patches in the partial ordering. Thus, patches can be solved locally, without approximation, using only boundary data and outflow data from previously solved patches (see [41] for details). Furthermore, elements at the same level

DOI: 10.1002/nme

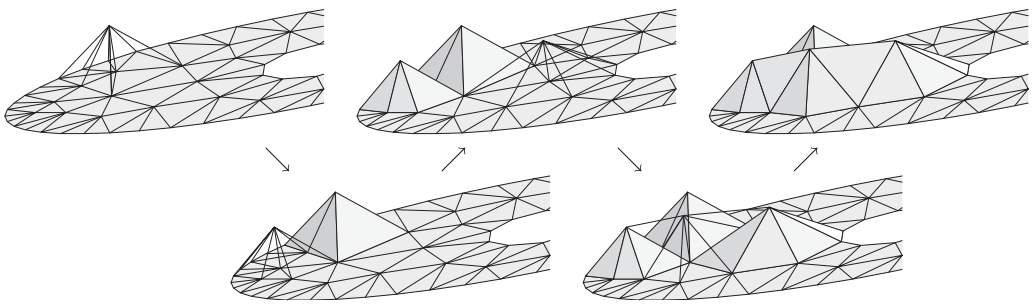

Figure 2. Pitching tents in spacetime. Wire-frame-rendered clusters of tetrahedra indicate newly constructed, unsolved patches; the shaded, triangulated surface depicts the current front in each image.

in the partial ordering are mutually independent and can be solved in parallel. This causality-based partial ordering is the basis of our advancing-front solution procedure.

We use the *Tent Pitcher* meshing procedure [43] to construct causal spacetime meshes in 2D$\times$ time, as illustrated in Figure 2. Tent Pitcher begins with a constant-time triangulation of the spatial domain at the initial analysis time; this is the initial *front*. A front is a triangulated spacetime surface, generally with nonuniform time coordinates. Tent Pitcher constructs a new front by advancing in time a local-minimum vertex of the current front. A *local-minimum vertex* is a vertex whose time coordinate is less than or equal to the time coordinates of all its neighboring vertices. The line segment between the old and new positions of the advancing vertex is called a *tent pole*. Each time the front advances, Tent Pitcher fills the volume between the old and new fronts with a patch (or *tent*) of tetrahedral elements that share the tent pole as a common edge. We refer to the process of erecting a new tent pole and generating the corresponding patch of elements as *pitching a tent*.

In order to guarantee that every patch is causal, we require every triangular facet in each new front to be a causal manifold. Tent Pitcher limits the height of each new tent pole to satisfy this *causality constraint*. Unfortunately, advancing the front sequentially, subject only to the causality constraint, can lead to a locking condition in which it is impossible to advance the front further. Tent Pitcher enforces a second constraint on the tent-pole height, called the *progress constraint*, to ensure that this locking condition does not occur (see [43] for details).

Each new causal patch can be solved independently of all subsequent patches, using data only from the spacetime domain boundary and from adjacent, previously solved elements. Tent Pitcher passes the geometric description of each new patch, along with all required material and face flags to the SDG finite element routines for solution. The finite element routines compute the SDG solution on the new patch and return it to Tent Pitcher. Tent Pitcher stores the new patch solution, updates the current front to the new front, and locates a new local-minimum vertex in preparation for pitching the next tent. Thus, the processes of spacetime mesh generation and finite element solution are interleaved at the patch level. The tent-pitching cycle repeats until the entire spacetime analysis domain is covered.

This patchwise solution method has several advantages. First, it delivers linear computational complexity in the number of patches (if the maximum degree of any vertex in the spatial triangulation is bounded). There is no need to assemble and store a global system of equations; the finite element routines are written to solve a single patch at a time. The patchwise solution

DOI: 10.1002/nme

procedure localizes the nonlinearity of the cohesive model to only those patches that contain cohesive faces. Assuming linear response in the bulk material (as we do here), we can solve most patches using linear analysis methods. The local structure also facilitates adaptive analysis, as described in Section 3.3. If the error in the current patch is unacceptable, the mesh can be refined locally; only the solution on that particular patch need to be discarded. Thus, the overall cost of the adaptive solution is reduced relative to adaptive methods that require recalculation of the solution over a global time step. Finally, the patch-by-patch solution technique lends itself to parallel computation. Multiple tents can be pitched and solved simultaneously on separate processors, subject only to the partial ordering constraint for patches. The local characteristics of the algorithm minimize interprocessor communication.

3.2. Implementing a cohesive model within the SDG framework

Our implementation of the cohesive model involves three relatively simple extensions to the basic SDG algorithm. These are: identification of spacetime element faces associated with the trajectory Γ^{C} of the cohesive interface, use of the TSL to compute the target momentum flux $\boldsymbol{M}^*$ on Γ^{C} (see Section 2.4), and solution of the resulting non-linear equations. We begin by aligning edges in the initial spatial triangulation with the cohesive interfaces and by setting a boundary-type flag for every triangle edge to indicate whether the edge is associated with $\partial D^{\boldsymbol{M}}$, $\partial D^{\boldsymbol{v}}$, an interior surface where Riemann values are required, or a cohesive interface. Then, as Tent Pitcher constructs the spacetime mesh, vertical element faces are assigned the boundary-type flag of the triangle edge they are built over. Since cohesive interfaces are material surfaces, their trajectories in spacetime are always vertical. Thus, Tent Pitcher automatically identifies all spacetime element faces on Γ^{C} as cohesive faces.

Whenever the finite element solution algorithm encounters a spacetime element face that Tent Pitcher has flagged as cohesive, it uses (23) and (24) to implement the traction–separation relation as it evaluates $\boldsymbol{M}^{\mathrm{C}}$ in (21). Our current implementation uses the Xu and Needleman potential (25) and the simplified component relations (26). However, the implementation is completely modular such that any traction–separation relation could be easily substituted for (26). In particular, our method is extensible to large-deformation, irreversible and history-dependent traction–separation relations.

3.3. Adaptive analysis

The distinct length and time scales associated with macroscopic response, shock fronts[‖] and fracture process zones make elastodynamic crack propagation an inherently multi-scale phenomenon. As such, its analysis calls for numerical meshes that are strongly graded in both space and time. Furthermore, the motions of shocks and propagating cracks suggest that the mesh grading should evolve to follow the trajectories of these features. Since very short time increments are required to capture these local effects, a numerical simulation should avoid imposing a global, spatially uniform time step.

Our use of unstructured spacetime meshes is responsive to these requirements. The SDG model facilitates the construction of strongly graded meshes because nonconforming meshes are fully admissible. However, the complex trajectories of shocks and crack tips are not easily

[‖]Here we use 'shocks' to refer to weak shocks in the derivatives of the displacement field.

DOI: 10.1002/nme

predicted *a priori*, hence, adaptive analysis methods are generally needed to capture these dynamic phenomena. This section describes adaptive extensions of the SDG method that ensure an accurate solution, especially when shocks and moving crack-tip fields are present. Geometric aspects of adaptive mesh refinement and coarsening are described in Section 3.3.1. Adaptive methods for controlling numerical dissipation and the error associated with fracture energy are described in Sections 3.3.2 and 3.3.3.

3.3.1. Adaptive mesh refinement and coarsening. We use an adaptive extension of the Tent Pitcher algorithm, as described in [44], to implement adaptive refinement and coarsening within our patch-by-patch, advancing-front solution algorithm. Rather than adapting patches of spacetime elements directly, Tent Pitcher implements adaptive refinement by managing the triangulation of the current front. Each time a patch is solved, local error indicators (see below) are computed for each element in the patch and tested against a user-specified, per-element target error.

If any error indicator is too far above its target value in a given element, then that element is marked for *refinement* and the solver *rejects* the patch when it is returned to Tent Pitcher. Tent Pitcher, in turn, discards the rejected patch and, using a newest vertex bisection algorithm (see [44] for details), refines the corresponding facets of the current front. This procedure effectively refines the spacetime mesh in both space and time when tent pitching is resumed, because the causality constraint dictates shorter tent-pole heights (local time steps) at vertices associated with refined facets. Note that Tent Pitcher discards the solution only on the rejected patch. The solutions on all previously solved patches are unaffected due to the patchwise causal structure of the spacetime grid, hence, the amount of redundant calculation is minimized when the front is refined.

If all the error indicators in a given element are too far below their target values, then that element is marked as *coarsenable* when it is returned to Tent Pitcher. Otherwise, if an element is neither coarsenable nor marked for refinement, the element is marked as *acceptable*. Tent Pitcher accepts the solution on the current patch if all elements in the patch are either acceptable or coarsenable. In this case, Tent Pitcher stores the patch solution, advances the front and copies the status (acceptable or coarsenable) from the patch elements to the corresponding facets of the new front. When all the elements surrounding a vertex in the front are marked as coarsenable, Tent Pitcher implements a special spacetime operation, as described below, to delete the vertex from the front.

Coarsening operations in conventional adaptive algorithms involve remeshing followed by a projection of the solution from the old mesh onto the new mesh. The projection introduces error that limits convergence rates to $\mathcal{O}(h)$, even for higher-order elements in regions where the solution is smooth. Tent Pitcher implements coarsening in spacetime with special *coarsening patches* whose causal inflow facets match the causal outflow facets of previously solved patches, and whose causal outflow facets generate a new front in which one of the vertices in the old front has been deleted. This approach eliminates the projection operation in conventional remeshing procedures. Hence, it preserves the full convergence rates of higher-order elements and preserves elementwise balance of linear and angular momentum. See Figure 1 in [50] for illustrations of the geometries of spacetime patches that perform coarsening and other adaptive meshing operations without incurring projection error.

3.3.2. Adaptive control of numerical dissipation. Since our underlying SDG method is dissipative, we limit numerical dissipation to ensure an accurate solution. Further, to achieve an efficient

DOI: 10.1002/nme

solution, we seek to distribute the numerical dissipation evenly over the spacetime mesh. Let $\epsilon_{\mathrm{D}}^{\mathrm{Q}}$ denote the numerical dissipation error for element Q, expressed as the total energy dissipation over Q less the physical dissipation associated with cohesive crack propagation. Recalling (10) and (14) and enforcing the target flux values, we find that

$$\epsilon_{\mathrm{D}}^{\mathrm{Q}} = \frac{1}{2}\int_{\partial Q} (\dot{\boldsymbol{u}}^* \wedge \boldsymbol{M}^* + \boldsymbol{\varepsilon}^* \wedge \mathbf{i}\boldsymbol{M}^*) + \int_Q \dot{\mathbf{u}} \wedge \rho \boldsymbol{b} \tag{27}$$

Let $\tilde{\epsilon}_{\mathrm{D}}$ be the target dissipation per element. The dissipation on element Q is considered acceptable when $\underline{\epsilon}_{\mathrm{D}} \leqslant \epsilon_{\mathrm{D}}^{\mathrm{Q}} \leqslant \overline{\epsilon}_{\mathrm{D}}$, where $\underline{\epsilon}_{\mathrm{D}} = (1-\eta)\tilde{\epsilon}_{\mathrm{D}}$ and $\overline{\epsilon}_{\mathrm{D}} = (1+\eta)\tilde{\epsilon}_{\mathrm{D}}$, in which $\eta > 0$ is a user-specified parameter. Refinement is required when $\epsilon_{\mathrm{D}}^{\mathrm{Q}} > \overline{\epsilon}_{\mathrm{D}}$, and element Q might be coarsenable (depending on whether it is also coarsenable with respect to the cohesive traction error defined in Section 3.3.3) when $\epsilon_{\mathrm{D}}^{\mathrm{Q}} < \underline{\epsilon}_{\mathrm{D}}$. The parameter η should be chosen sufficiently large to avoid undesirable cycling between coarsening and refinement. We use $\eta = 0.2$ in our current implementation.

Although it is not used directly in our adaptive algorithms, the *total energy dissipation over D with respect to the target flux on ∂D*, denoted by ϵ_{D}, provides a useful measure of solution accuracy. It is given by

$$\epsilon_{\mathrm{D}} := \frac{1}{2}\int_{\partial \mathrm{D}} (\dot{\boldsymbol{u}}^* \wedge \boldsymbol{M}^* + \boldsymbol{\varepsilon}^* \wedge \mathbf{i}\boldsymbol{M}^*) + \int_{\mathrm{D}} \dot{\mathbf{u}} \wedge \rho \boldsymbol{b}. = \sum_{Q \in \mathscr{P}_h} \epsilon_{\mathrm{D}}^{\mathrm{Q}} \tag{28}$$

3.3.3. Adaptive enforcement of the cohesive traction–separation relation. Cohesive models can frustrate numerical stability when too few elements are included in the active cohesive process zone [51]. A common response to this problem is to use quasi-static estimates of process-zone size to determine a maximum diameter for cohesive elements that applies throughout an elastodynamic simulation. However, this approach is not conservative, since the process-zone size decreases from the quasi-static value with increasing crack velocity. Ideally, adequate resolution within active process zones should be enforced adaptively during the course of a simulation in response to an evolving solution. Beyond meeting this basic stability requirement, the discrete solution should track the traction–separation relation with sufficient precision to ensure that the work of separation and the history of crack-tip motion are modeled accurately. Adaptive control of the numerical dissipation is not always sufficient to meet these requirements, so we introduce a second adaptive refinement criterion that directly limits the error in the work of separation, a fundamental quantity in fracture mechanics that governs crack kinetics. Our numerical experience indicates that this refinement criterion can play an important role in capturing the correct kinetics of fast-moving cohesive process zones.

We define a *cohesive energy error*, ϵ_{C}, to measure the error in the fracture energy induced by the mismatch between the tractions $\mathbf{t}$ generated by the trace of the finite element stress solution and the target tractions $\mathbf{t}_{\mathrm{c}}$ generated by the TSL on $\Gamma^{\mathbf{C}}$:

$$\epsilon_{\mathrm{C}} = \|(\mathbf{t} - \mathbf{t}_{\mathrm{c}}) \cdot \mathbf{v}\|_{L_1(\Gamma^{\mathrm{C}})} \tag{29}$$

In differential forms notation, the traction $\mathbf{t}$ on the cohesive surface is simply the restriction to $\Gamma^{\mathbf{C}}$ of the trace from the element interiors of the d-form $\boldsymbol{\sigma}$. This is the analogue of computing $\mathbf{t} = \boldsymbol{\sigma} \cdot \mathbf{n}$

DOI: 10.1002/nme

in tensor notation. The cohesive energy error on element Q is given by

$$\epsilon_{\mathrm{C}}^{\mathrm{Q}} = \|(\mathbf{t}-\mathbf{t}_{\mathrm{c}})\cdot\mathbf{v}\|_{L_1(\partial Q_{\mathrm{c}})} \tag{30}$$

Let $\tilde{\epsilon}_{\mathrm{C}}$ be the target value of the elementwise cohesive energy error. The cohesive energy error on element Q is acceptable when $\underline{\epsilon}_{\mathrm{C}} \leqslant \epsilon_{\mathrm{C}}^{\mathrm{Q}} \leqslant \overline{\epsilon}_{\mathrm{C}}$, where $\underline{\epsilon}_{\mathrm{C}} = (1-\eta)\tilde{\epsilon}_{\mathrm{C}}$ and $\overline{\epsilon}_{\mathrm{C}} = (1+\eta)\tilde{\epsilon}_{\mathrm{C}}$. Refinement is required when $\epsilon_{\mathrm{C}}^{\mathrm{Q}} > \overline{\epsilon}_{\mathrm{C}}$, and element Q might be coarsenable (depending on whether it is also coarsenable with respect to the dissipation error) when $\epsilon_{\mathrm{C}}^{\mathrm{Q}} < \underline{\epsilon}_{\mathrm{C}}$.

3.4. Postprocessing

We require special procedures to postprocess data computed on unstructured spacetime grids because these data do not naturally align with constant-time planes. We also need to extract trajectories of certain 'landmark' positions along the cohesive interface to interpret the fracture history implied by our SDG solutions. These special locations include nominal positions for the crack tip, the leading and trailing edges of the process zone, and the core of singular fields that might develop in the vicinity of the crack tip. This section describes special conventions and procedures that address these requirements, as well as per-pixel-accurate rendering methods used to visualize and animate the data.

Cohesive models replace a mathematically sharp crack tip with a cohesive process zone that has a finite size, but in general, no clear boundary. Thus, no crack-tip position exists in a conventional sense, and we need to introduce conventions to define a nominal crack-tip position as well as the extent of the process zone. Here, we define the *nominal crack-tip position* as the location $\mathbf{x}_{\mathrm{c}}$ where the critical separation δ_{c}, corresponding to the peak cohesive stress in the TSL, is attained. Neither the leading nor the trailing edge of the cohesive process zone is sharply defined in the exponential TSL of Xu and Needleman. Therefore, we also introduce conventions to identify nominal positions for these features. The trailing edge of the process zone is associated with the position where complete separation first occurs across the cohesive interface. Since the traction only vanishes asymptotically for large separations, we identify the trailing edge with the position $\mathbf{x}_{\mathrm{T}}$ where the separation takes the value $\delta_{\mathrm{T}} > \delta_{\mathrm{C}}$ and the traction reduces to 1% of the peak cohesive stress. The leading edge of the process zone is identified with the separation on the loading branch where the TSL exhibits a significant deviation from linearity. In this study, we identify the leading edge of the cohesive process zone as the position, $\mathbf{x}_{\mathrm{L}}$, where the separation attains the value $\delta_{\mathrm{L}} = 0.80\ \delta_{\mathrm{c}}$. The cohesive process zone is defined as the region between $\mathbf{x}_{\mathrm{T}}$ and $\mathbf{x}_{\mathrm{L}}$, and its size is given by $\Lambda = |\mathbf{x}_{\mathrm{L}} - \mathbf{x}_{\mathrm{T}}|$.

The crack-tip velocity is distinct from the material velocity at the crack tip, since the tip of a propagating crack is not a material location. Common practice for computing crack-tip velocity in semi-discrete models involves recording the crack-tip position at discrete-time intervals and then applying a finite difference operator. Alternatively, one can differentiate a curve fit locally to the discrete data [19, 52]. All of these approaches filter data over discrete-time increments and tend to filter out meaningful high-frequency response information along with numerical noise. In this work, we compute crack velocities analytically by applying the implicit function theorem to the definition of the nominal crack-tip position $\mathbf{x}_{\mathrm{c}}$.

Let the separation (normal, tangential or mixed, depending on the fracture mode) on a spacetime cohesive manifold Γ be given by $\Delta = f(\mathbf{x}, t)$. Then the trajectory of the nominal crack tip is the

DOI: 10.1002/nme

set, $\mathscr{C}=\{(\mathbf{x},t)\in\Gamma: f(\mathbf{x},t)=\delta_c\}$. By the implicit function theorem, the analytical crack-tip velocity $\dot{a}$ at any position $(\mathbf{x}_c,t)\in\mathscr{C}$ where f is differentiable is given by

$$\dot{a}(\mathbf{x}_c,t)=-\left.\frac{\dfrac{\partial f}{\partial t}}{\dfrac{\partial f}{\partial \mathbf{x}}}\right|_{(\mathbf{x}_c,t)} \tag{31}$$

The structure of SDG basis functions implies that f is differentiable almost everywhere along the crack-tip trajectory $\mathscr{C}$. Using Equation (31), we can compute and plot crack-tip velocity histories to any desired resolution. Rather than resort to filtering, we rely on convergence under intense adaptive mesh refinement to distinguish high-frequency physical response from numerical noise in the examples reported in Section 4.

We use special per-pixel rendering software developed by Zhou *et al.* [53] to visualize our spacetime data sets. This software evaluates high-order finite element solutions and their derivatives directly at every pixel. The resulting high-resolution images reflect the raw solution data, without smoothing or interpolation in the graphics rendering pipeline. Numerical artifacts are readily apparent, hence, we are better able to judge the quality of our solutions. A series of constant-time images can be viewed sequentially to animate the motions of elastic waves and running cracks.

4. NUMERICAL RESULTS

This section presents numerical results that demonstrate the effectiveness of the SDG implementation of cohesive models. We study a middle-crack tension specimen to explore the model's convergence properties and stability. In particular, we compare crack-tip velocity histories predicted by the SDG model with those reported in [19] and investigate in some detail the development of the cohesive process zone and associated quasi-singular response in the material velocity field. All of the studies in this section are based on tetrahedral spacetime elements with complete cubic polynomial bases.

4.1. Analysis model for middle-crack tension specimen

We analyze cohesive crack growth in a middle-crack tension specimen subject to plane-strain conditions and loaded by a uniform, prescribed velocity along two opposite edges. We apply symmetry conditions to model a single quadrant with dimensions, $L=10\,\text{mm}$, $W=3\,\text{mm}$ and initial crack length, $a_0=4.25\,\text{mm}$, as shown in Figure 3. The cohesive interface extends to the right of the initial crack-tip position along the bottom edge of the quadrant. We use a material model that approximates the elastic properties of polymethyl methacrylate: Young's modulus, $E=3.24\,\text{GPa}$; the Poisson ratio, $\nu=0.35$; and mass density, $\rho=1190\,\text{kg/m}^3$. The dilatational, shear and Rayleigh wave speeds are given by [19, 26].

$$c_d=\sqrt{\frac{E(1-\nu)}{\rho(1+\nu)(1-2\nu)}},\quad c_s=\sqrt{\frac{E}{2\rho(1+\nu)}},\quad c_R\approx c_s\frac{0.862+1.14\nu}{1+\nu} \tag{32}$$

For the given values of the material parameters, these speeds are $c_d=2090\,\text{m/s}$, $c_s=1004\,\text{m/s}$, and $c_R=938\,\text{m/s}$. The cohesive properties for the TSL are based on those used in [19]. We use

DOI: 10.1002/nme

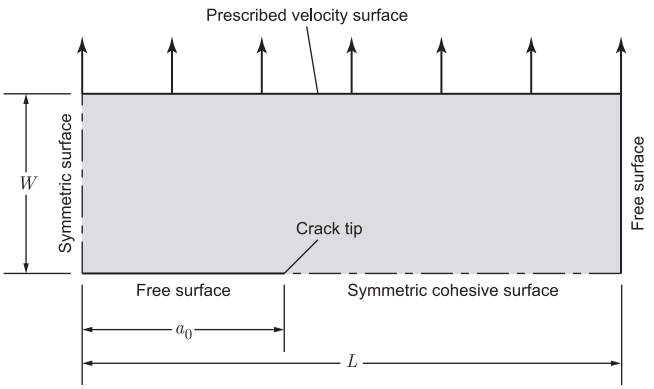

Figure 3. Model for middle-crack tension specimen.

two values of the cohesive strength in the examples below: either $\sigma_C = 0.1E = 324\,\text{MPa}$, identical to that in [19], or $\sigma_C = 0.01E = 32.4\,\text{MPa}$. The critical separation distance for both the normal and tangential directions is $\delta = 4.0 \times 10^{-4}\,\text{mm}$.

A uniform, prescribed velocity, ramping linearly from zero to a sustained value $\bar{v}$, is applied along the top edge of the model, as depicted in Figure 3. The duration of the prescribed-velocity ramp is $t_0 = 0.1\,\mu\text{s}$. For purposes of studying the effects of varying the cohesive strength σ_C, we seek loads that vary in proportion to the strength. The boundary traction induced by the prescribed velocity $\bar{v}$ is, roughly, proportional to $E\bar{v}/c_d$. Therefore, we fix the non-dimensional group, $(E\bar{v})/(\sigma_C c_d)$, to maintain an invariant relation between the induced boundary traction and the cohesive strength. Thus, holding E and c_d constant, we use $\bar{v} = 15$ and $1.5\,\text{m/s}$ for $\sigma_C = 324$ and $32.4\,\text{MPa}$, respectively.

4.2. Convergence of the model and performance of two adaptive criteria

Cohesive models for quasi-static fracture analysis require adequate resolution of the cohesive process zone to maintain numerical stability [5, 54]. It is also essential to resolve the process zone in dynamic fracture studies, but robust minimum requirements for maintaining stability have not yet been established for this class of problems. Further, the minimum requirements for stability generally do not suffice to ensure an accurate simulation of cohesive material failure. We use two error indicators in our h-adaptive implementation, as described in Section 3.3, to guarantee both stability and accuracy. The dissipation error indicator addresses the accuracy of our model for the dynamic response in the bulk material, while the cohesive energy error indicator ensures the accuracy and stability of our numerical rendering of the TSL. The studies in this section demonstrate the effectiveness and convergence properties of the adaptive error indicators, both for the dissipation indicator alone and for the two indicators in combination. We use $\sigma_C = 0.01E = 32.4\,\text{MPa}$ and $\bar{v} = 1.5\,\text{m/s}$, and we compute the crack-tip trajectory using the position $\mathbf{x}_c$, cf. Section 3.4.

4.2.1. Convergence study using only the dissipation error indicator. We study the convergence of the crack-tip trajectory with varying levels of adaptive mesh refinement, as controlled by the target per-element dissipation defined in Section 3.3.2. We deactivate the cohesive error indicator in this study, so that the target dissipation is the only control on adaptive refinement.

DOI: 10.1002/nme

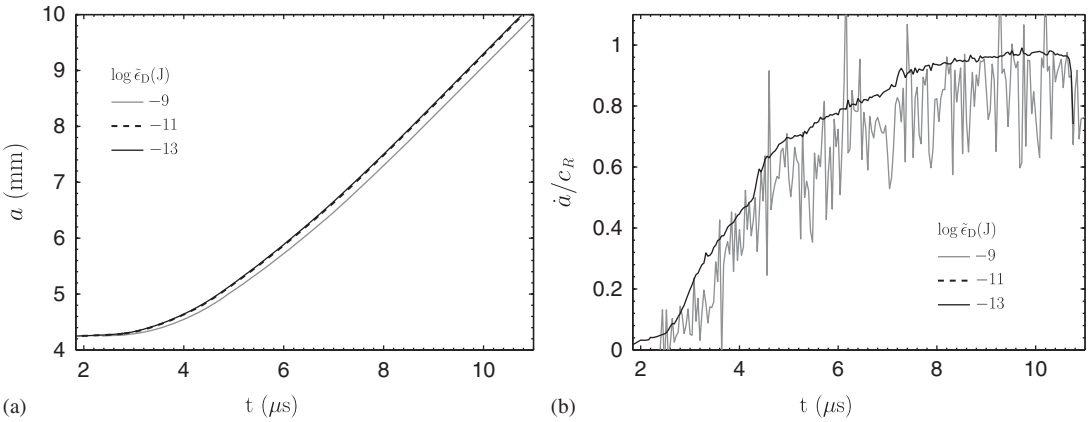

Figure 4. Convergence of crack-tip trajectory and velocity with decreasing target dissipation per element, $\tilde{\epsilon}_D$. (a) Crack length and (b) Crack-tip velocity.

Figure 4 shows the convergence of the crack-tip trajectory (or, equivalently, the crack-length history $a(t)$) as the target per-element dissipation, $\tilde{\epsilon}_D$, decreases; that is, with increasing adaptive refinement. The model becomes unstable, due to inadequate resolution of the active process zone, when $\tilde{\epsilon}_D$ is larger than the values displayed in the figure. The crack-tip trajectory is virtually identical to the results displayed for $\tilde{\epsilon}_D = 10^{-11}$ J when we specify smaller values for $\tilde{\epsilon}_D$.

Figure 4(b) shows the convergence of the crack-tip velocity, normalized by the Rayleigh wave speed c_R, as $\tilde{\epsilon}_D$ decreases. These time-derivative data sets naturally exhibit more numerical noise than the crack-tip trajectory data and, therefore, require more adaptive refinement for convergence. We emphasize that here we are plotting velocity as pointwise analytical derivatives obtained directly from the cubic polynomial basis functions, cf. (31). This provides a far more stringent test than the methods commonly used with semi-discrete data sets. Nonetheless, the noise is nearly eliminated for sufficiently small values of $\tilde{\epsilon}_D$. On the other hand, the results for $\tilde{\epsilon}_D = 10^{-9}$ J, the largest value consistent with numerical stability, show considerable scatter and a delayed onset of rapid crack extension relative to the converged solutions.

4.2.2. Convergence with error indicators for both bulk dissipation and cohesive fracture energy. Reducing $\tilde{\epsilon}_D$ serves to increase the overall level of mesh refinement, but it does not directly ensure sufficient resolution of the TSL in the process zone. Since the process-zone size decreases as the crack velocity increases [55], models designed to adequately resolve the quasi-static process zone might eventually lose accuracy, or even stability, in dynamic analysis. Furthermore, adaptive refinement based solely on dissipation is a relatively expensive means to ensure accurate resolution of the TSL, because the increased level of refinement applies to the entire spacetime analysis domain, not just the neighborhood of the cohesive process zone. Even for very small values of $\tilde{\epsilon}_D$, the crack-tip-velocity history contains some numerical noise, as shown in Figure 4(b). Thus, it is useful to introduce a second adaptive error indicator ϵ_C^Q, cf. Section 3.3.3, to control directly the error in the fracture energy and thereby ensure a stable and accurate resolution of the process zone.

DOI: 10.1002/nme

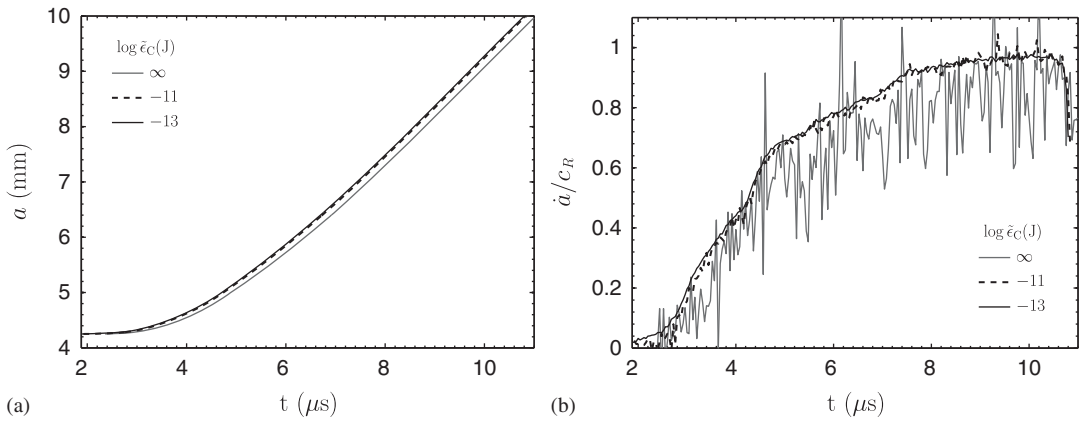

Figure 5. Convergence of crack-tip trajectory and velocity with decreasing target cohesive energy error per element, $\tilde{\epsilon}_C$ for $\tilde{\epsilon}_D = 10^{-9}$ J. (a) Crack length and (b) Crack-tip velocity.

The two error indicators play complementary roles. The dissipation indicator governs the overall accuracy of the spacetime solution, especially the resolution of sharp wave fronts in the bulk material, while the cohesive error indicator ensures an accurate rendering of the TSL along the cohesive interface. This approach improves computational efficiency because the cohesive error indicator provides a means to boost mesh refinement locally in the vicinity of the cohesive process zone without affecting the global level of refinement.

We choose $\tilde{\epsilon}_D = 10^{-9}$ J, the largest value of $\tilde{\epsilon}_D$ that ensures stability in Section 4.2.1, and activate the cohesive error indicator to study its effectiveness. The crack-tip trajectory corresponding to $\tilde{\epsilon}_D = 10^{-9}$ J differs significantly from the better-resolved solutions (Figure 4), hence, it serves as a good reference for testing the cohesive error indicator. Figure 5(a) shows the convergence of the crack-tip trajectory as the target per-element cohesive energy error $\tilde{\epsilon}_C$ decreases. The gray curve for $\tilde{\epsilon}_C = \infty$ corresponds to the solution with $\tilde{\epsilon}_D = 10^{-9}$ J and without the cohesive error criterion. The black curves show results for adaptive runs with both the diffusive and cohesive error criteria active. The crack-tip trajectories for smaller values of $\tilde{\epsilon}_C$ (not shown) are virtually identical to the one for $\tilde{\epsilon}_C = 10^{-11}$ J.

Figure 5(b) shows the convergence of the crack-tip velocity history with respect to $\tilde{\epsilon}_C$. The numerical noise decreases as $\tilde{\epsilon}_C$ decreases, and the lag in the onset of crack extension vanishes. There is a period of near-zero tip velocity that marks the transition from a stationary crack to cohesive crack propagation. This feature is less clearly defined for larger values of $\tilde{\epsilon}_C \geqslant 10^{-11}$ J. Energy flowing into the cohesive process zone during this initiation stage is largely devoted to the formation of a quasi-singular velocity field, as described in the next section. Once the singular field becomes well established, the crack tip accelerates as the energy flow is redirected to the formation of new crack surface.

Figure 6 shows how enhancing dissipation-based adaptive analysis with adaptive control of the cohesive energy error improves computational efficiency. The plot shows cost, measured by the number of spacetime elements N, as a function of cohesive energy error ϵ_C for different combinations of $\tilde{\epsilon}_D$ and $\tilde{\epsilon}_C$ at time $t = 6\,\mu$s. The black curve shows results for dissipation error control alone, with no direct control over the cohesive error. Each of the patterned curves shows

DOI: 10.1002/nme

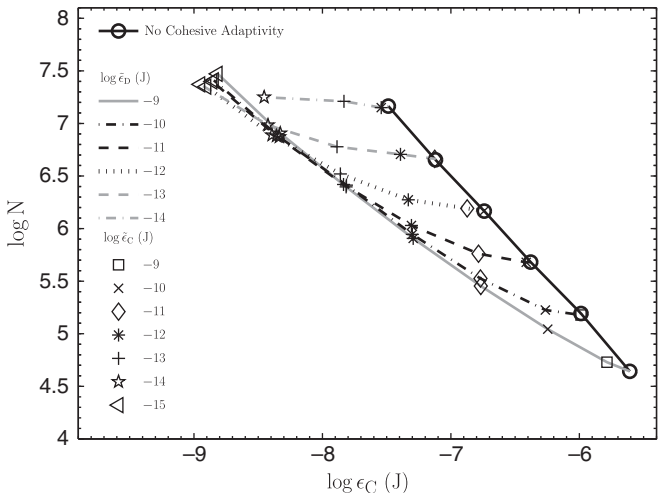

Figure 6. Relative efficiency of adaptive strategies, using single and paired error indicators, in reducing cohesive energy error ϵ_C. Number of spacetime elements N measures computational expense. Solutions sharing a common per-element target cohesive error $\tilde{\epsilon}_C$ have similar total cohesive errors ϵ_C and, therefore, tend to align vertically in this plot.

the effects of varying the cohesive error tolerance with $\tilde{\epsilon}_D$ fixed. For a given cost N, the combined approach always yields a smaller error, by as much as on order of magnitude or more. For each choice of $\tilde{\epsilon}_D$, the incremental cost of adding cohesive error control is generally modest up to the point where $\tilde{\epsilon}_C \approx \tilde{\epsilon}_D$. Thereafter, for $\tilde{\epsilon}_C < \tilde{\epsilon}_D$, the analysis cost tends to increase more rapidly with decreasing $\tilde{\epsilon}_C$. There is little advantage to requiring either of the per-element energy errors, dissipative or cohesive, to be much smaller than the other. For example, stringent control of the cohesive energy error cannot compensate for dissipation due to inadequate resolution of sharp wavefronts in the bulk solution. Therefore, we recommend using the same value for $\tilde{\epsilon}_C$ and $\tilde{\epsilon}_D$. Since this choice adds only minor expense relative to dissipation control alone, adaptive control of the cohesive error is an attractive option, particularly when high-resolution solutions are required.

Figure 7 shows how alternative methods for computing crack-tip velocities can produce dramatically different results. In particular, we compare the analytical method based on the implicit function theorem, cf. (31), with the parabolic fit method described in [19]. The reference velocity history in Figure 7 was extracted from a highly resolved solution, computed with $\tilde{\epsilon}_D = 10^{-9}$ J and $\tilde{\epsilon}_C = 10^{-13}$ J, using the implicit equation (31). The two other histories were extracted from a lower-resolution solution, computed with $\tilde{\epsilon}_D = 10^{-9}$ J and $\tilde{\epsilon}_C = 10^{-9}$ J, that is not fully converged. The result labeled 'implicit' is computed with the analytical expression (31). Considerable numerical noise is evident, an indication that the underlying solution is not converged. When we apply the numerical differentiation method described in [19] to the same under-resolved solution, we obtain the velocity history labeled 'parabolic fit'. This approach filters out most of the numerical noise, but still exhibits significant deviations from the converged reference solution. The relatively noise-free result might give a false impression of convergence, and the method's filtering effect might eliminate meaningful high-frequency response along with the noise. While numerical

DOI: 10.1002/nme

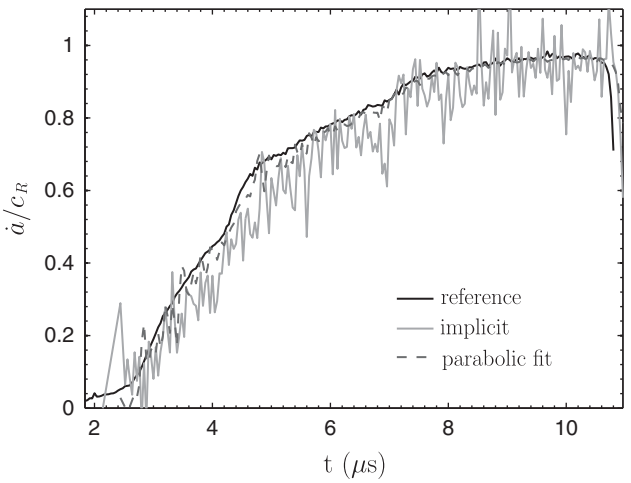

Figure 7. Comparison of methods for computing crack speed $\dot{a}$.

differentiation is a suitable choice for many applications, analytical extraction of crack-tip velocities might be preferred for studies that demand high-precision and reliability.

4.3. Dynamic fracture under shock loading conditions

This section presents numerical results for a model of dynamic fracture with high cohesive strength. We describe the details of crack-growth initiation and the development of quasi-singular velocity response which, to our knowledge, has not previously been reported in the literature on cohesive fracture. For purposes of comparison, we investigate the same problem studied by Xu and Needleman in [19]. The model parameters are given in Section 4.1; the only differences from the parameters used in the previous section are that the cohesive strength and the applied boundary velocity are 10 times larger: $\sigma_C = 0.1E = 324\,\text{MPa}$ and $\bar{v} = 15\,\text{m/s}$. The terminal time of the simulation is $t = 18.0\,\mu\text{s}$.

The capabilities of the adaptive SDG method to resolve shocks accurately and to control the fidelity of the numerical approximation of the TSL are key to the integrity of this study. The adaptive analysis parameters, $\tilde{\epsilon}_D = 10^{-11}\,\text{J}$ and $\tilde{\epsilon}_C = 10^{-10}\,\text{J}$, generate a spacetime mesh at the end of the simulation with a total of 117 million tetrahedra arranged in 20 million patches. A detail of the spacetime mesh at an intermediate stage of the analysis appears in Figure 8. This discretization yields a total numerical energy dissipation of 4.5% of the net energy inflow from the prescribed-velocity loading, while the linear and angular momentum balance to within machine precision globally, as well as locally on each element. The total cohesive energy error is $2.1 \times 10^{-4}\,\text{J}$, or 10.4% of the energy needed to completely debond the plates.

Figure 9 presents a series of still images from an animation of the SDG solution generated by the per-pixel-accurate rendering procedure described in Section 3.4. Only the upper-right quadrant of the center-cracked plate is shown. The color field** depicts the log of the strain-energy density,

**On-line version only. Color is mapped to gray scale in the printed version for Figures 9, 10 and 13.

DOI: 10.1002/nme

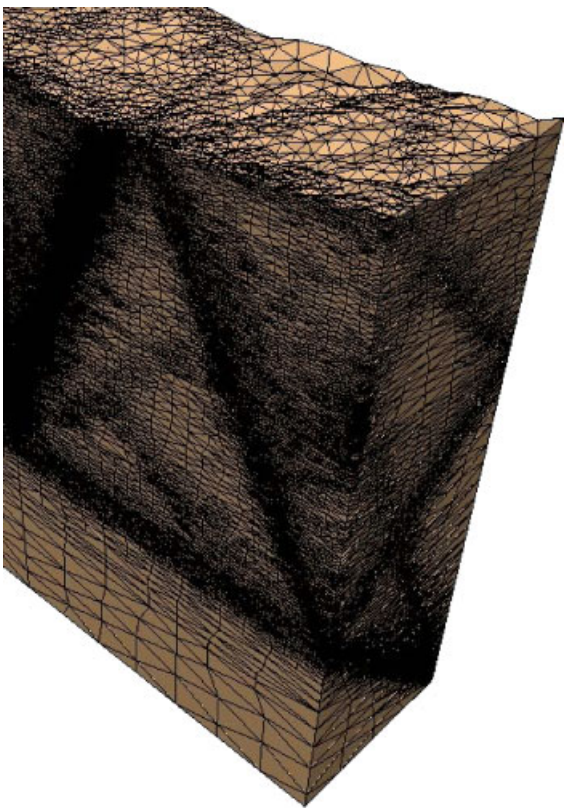

Figure 8. Detail of adaptively refined spacetime mesh at $t = 4.15\,\mu s$ with the time axis in the vertical direction. The left face of the spacetime volume corresponds to the cohesive surface, where the densely meshed region at the left follows the trajectory of the moving process zone. Other dense regions capture various wavefronts as they move through the bulk material.

where blue indicates zero energy density and violet indicates peak values. The height field depicts the modulus of the material velocity.

Figure 9(a) shows the progress through the plate of the primary wavefront generated by the applied loads at an early time, $t = 1.00\,\mu s$, as well as secondary waves generated along the free surface and Rayleigh waves traveling along the top and right surfaces. Figure 9(b) shows a time, $t = 1.85\,\mu s$, shortly after the primary wave reaches the crack plane. Wave scattering effects and a sharp gradient in the strain-energy density field are visible. A sharp spike in the velocity field starts to form, indicating the initiation stage of crack propagation. This spike develops into the quasi-singular structure described below. Figure 10 shows a detail view of the crack-tip region in which the solution is reflected about the crack plane to clarify the overall response in the plate. Figure 9(c) shows a later stage of crack propagation, where the velocity spike is more fully developed. Distinct wavefronts corresponding to scattered dilatational and shear waves are clearly visible at this stage.

The crack-tip velocity increases as propagation continues, until the velocity asymptotically approaches the Rayleigh wave speed, c_R. This acceleration tends to reduce the size of the active

DOI: 10.1002/nme

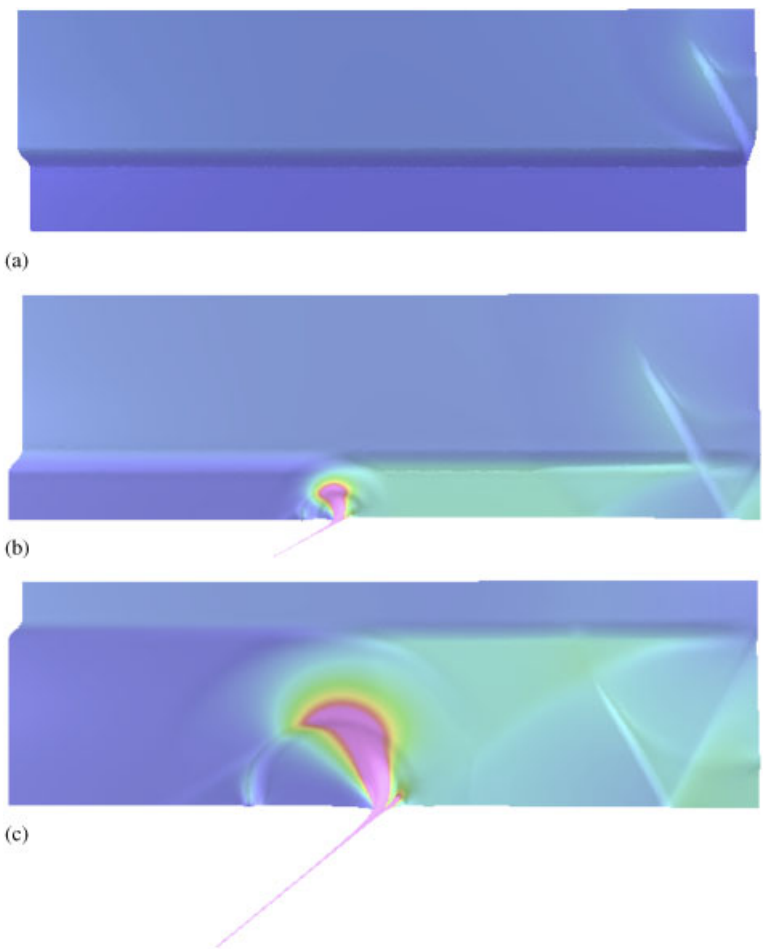

Figure 9. Visualization of crack propagation over time; upper-right quadrant of center-cracked plate. Color depicts log of strain-energy density; height field is modulus of material velocity. Initial crack-tip position is slightly to the left of midpoint of bottom edge. (a) $t = 1.00\,\mu s$; (b) $t = 1.85\,\mu s$; (c) $t = 2.60\,\mu s$; (d) $t = 3.75\,\mu s$; (e) $t = 6.25\,\mu s$; and (f) $t = 9.00\,\mu s$.

process zone. At the same time, propagation reduces the net section ahead of the crack tip, leaving less material to resist the tensile loading. Both effects strengthen the quasi-singular velocity response over time, as shown in Figures 9(d) and (e). There is a sharp release of kinetic energy from the quasi-singular structure in the velocity field when the crack breaks through the free boundary. This event sends sharp wavefronts through the plate interior and generates strong Rayleigh waves that travel in two directions along the broken plate's free boundaries, as depicted in Figure 9(f).

Asymptotic dynamic solutions for a running, mathematically sharp crack in a linear elastic medium exhibit an $r^{-1/2}$ singular form in the material velocity field [26], where r is the radial distance from the core of the singularity at the crack tip. Spikes in the velocity field surrounding

DOI: 10.1002/nme

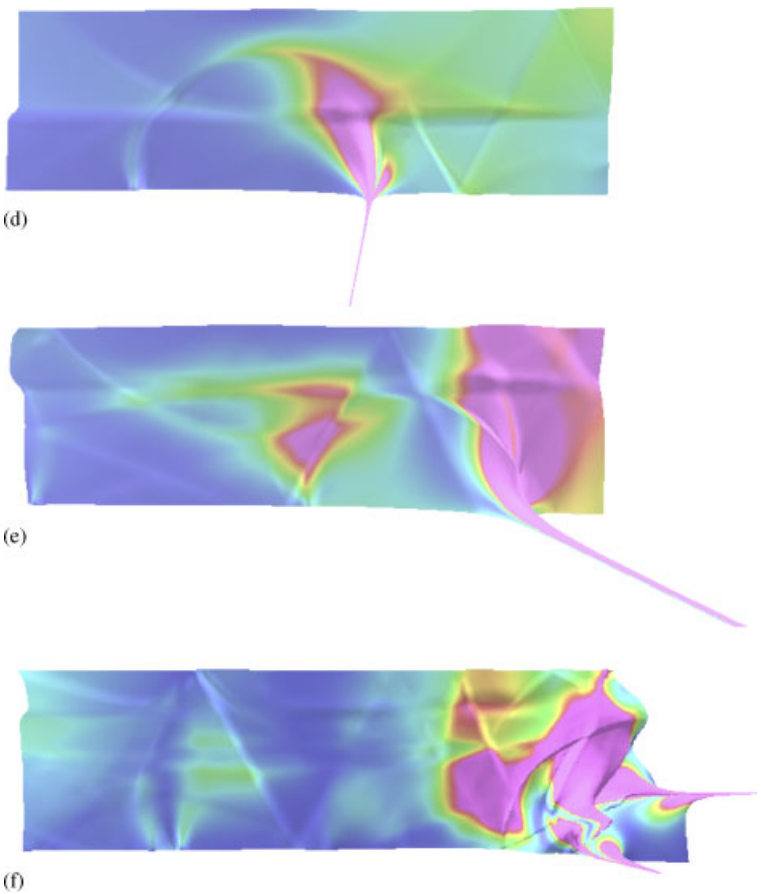

Figure 9. *Continued.*

the running crack tip, evident in Figure 9, suggest a similar structure in our simulation results. This would be an unexpected finding, if true. Cohesive models eliminate the associated singularities in the stress and strain fields by construction, hence, it is generally assumed that the velocity singularity is similarly annihilated.

Figure 11 presents a detailed analysis of the material velocity field in the neighborhood of a propagating crack tip. The figure shows a log–log plot of velocity magnitude versus radial distance r at $t = 5.0\,\mu s$, a time when the spike in the velocity field is well formed. The distance r is measured relative to the 'core' of the apparent singularity at $\mathbf{x}_0$, the position where the exponents of r in the expansions of the velocity field ahead of and behind the crack tip match. The velocity data in this plot are generated by directly evaluating the trace of the raw finite element solution for $\dot{\boldsymbol{u}}$ along the intersections of the constant-time plane with cohesive faces of individual spacetime elements. We sample multiple points within each intersected face to obtain a detailed plot of the velocity distribution. After identifying regions ahead of and behind the crack tip where the slope of the log–log plot is nearly constant (indicated by the shaded bands in Figure 11), we identify the core

DOI: 10.1002/nme

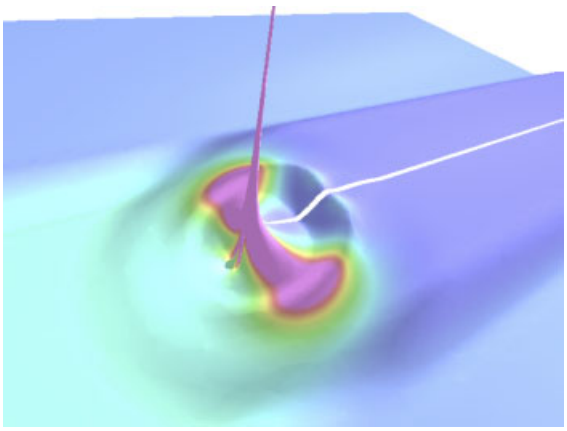

Figure 10. Detail view of developing quasi-singular velocity field at $t = 1.85\,\mu s$. Solution data are reflected about the crack plane to cover half the plate. The viewpoint position, relative to the layout in Figure 3, is above and to the right of the crack tip.

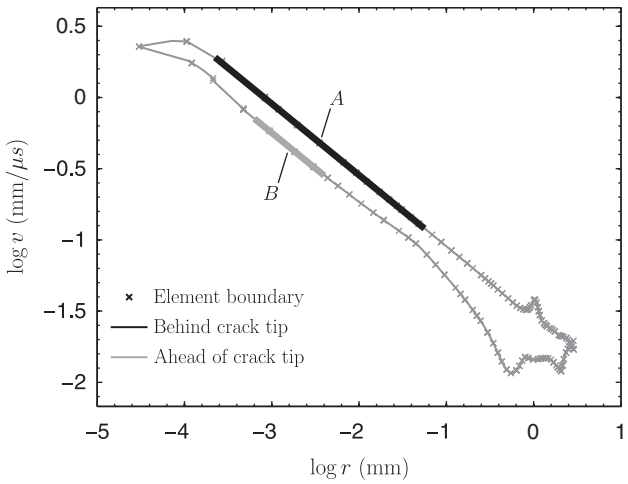

Figure 11. Analysis of quasi-singular velocity field, behind and ahead of moving crack tip, at $t = 5.0\,\mu s$. The modulus of the material velocity is $v := |\mathbf{v}|$, and shaded bands indicate ranges used to compute linear expansions:(A) $\log v = -0.500 \log r - 1.5456$; (B) $\log v = -0.500 \log r - 1.749$. Adaptive error tolerances are $\tilde{\epsilon}_D = 10^{-11}\,J$ and $\tilde{\epsilon}_C = 10^{-10}\,J$.

location by adjusting $\mathbf{x}_0$ until the slopes in the two regions match. We take the matching value of the slope as the exponent of r in the singular form.

For the data plotted in Figure 11, we found $x_0 = 7.13363\,mm$. The resulting exponent matches closely the theoretical value for a mathematically sharp crack, $-\frac{1}{2}$, for two decades of r behind $\mathbf{x}_0$ and one decade ahead. However, in contrast to the singular solution for a sharp, propagating crack, our highly resolved numerical solutions deviate from the singular structure for small radii

DOI: 10.1002/nme

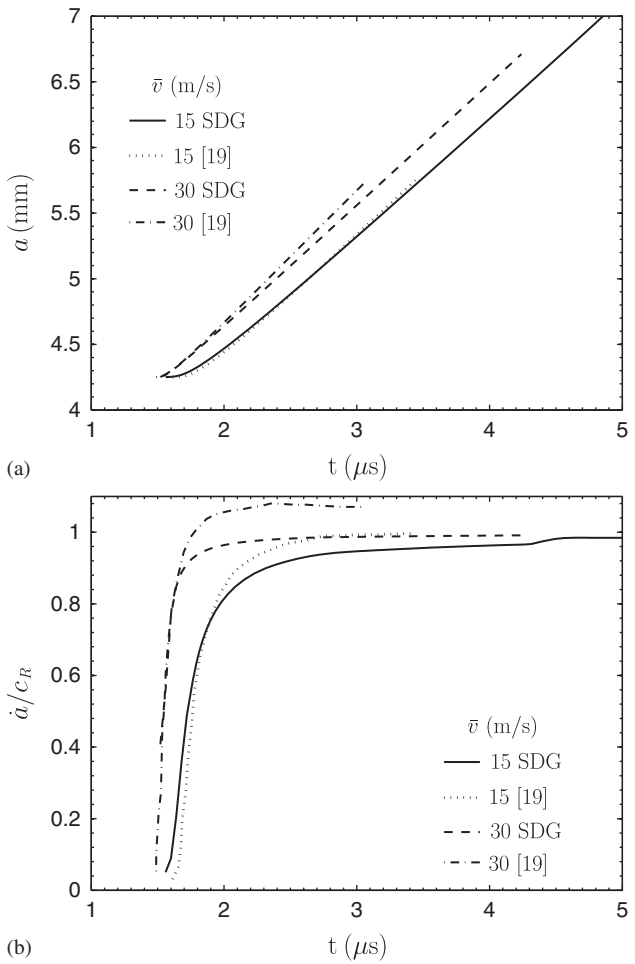

Figure 12. Trajectory and velocity of nominal crack tip; SDG solution compared with results from [19]. (a) Crack-tip position and (b) Crack-tip velocity.

$r < 10^{-3.3}$ mm, indicating a non-singular core. As it happens, the size of the non-singular core and the size of the process zone at the given time, $\Lambda = 10^{-3.1}$ mm (cf. Section 3.4), are of similar orders of magnitude, indicating that the cohesive process zone does indeed prevent a true singularity from forming. We use the term *quasi-singular* to describe this structure comprised of a non-singular core embedded within a zone that follows a singular form down to a small, but finite, radius. Figure 12 compares the crack-tip trajectory and velocity history predicted by the SDG cohesive model with corresponding results reported in [19] for two prescribed velocities, $\bar{v} = 15\,\mathrm{m/s}$ and $\bar{v} = 30\,\mathrm{m/s}$. Following the method in [19], we define the crack-tip position $\mathbf{x}_c$ in terms of the largest value of x along the cohesive interface for which $\Delta_n \geqslant \delta_n$.

Figure 12(a) compares crack-tip positions from the SDG solution with those reported in [19]. Xu and Needleman do not present results for the crack-tip trajectory directly, so we integrate their

DOI: 10.1002/nme

velocity data to plot the crack-tip position over time. Differences in the crack-tip trajectory are visible for both values of the prescribed velocity, but these are more pronounced for the faster loading, $\bar{v} = 30\,\mathrm{m/sec}$.

Figure 12(b) compares crack-tip velocity histories predicted by the SDG method with those reported in [19]. To facilitate comparison, we employ the same numerical differentiation scheme used in [19] to extract crack-tip velocities from the SDG solutions, rather than the analytical method used elsewhere in this work. This involves passing a quadratic polynomial through three successive values of $\mathbf{x}_c$ and recording the time derivative of the polynomial at the middle point as the crack-tip velocity. Comparison of the velocity solutions reveals larger differences than those obtained for crack-tip position. There is a modest lag in crack initiation in the results from [19] that is more pronounced in the data for $\bar{v} = 15\,\mathrm{m/s}$. This difference is likely due to the more accurate resolution of the formation of the quasi-singular velocity field in the SDG solution, since this phenomenon tends to delay initiation. The acceleration of the crack tip to the Rayleigh wave speed is noticeably more gradual in the SDG solution, which reaches the asymptotic value at a much later time for $\bar{v} = 15\,\mathrm{m/s}$. The relatively abrupt increase in crack-tip velocity at $t = 4.31\,\mu\mathrm{s}$ in the SDG solution is associated with the arrival of a reflected wave at the crack tip.

In contrast to the results from [19], the SDG crack-tip velocity for $\bar{v} = 30\,\mathrm{m/s}$ does not overshoot the Rayleigh speed. This difference can be attributed to the local balance properties of the SDG solution, its ability to adaptively track the dramatic decrease in the process-zone size as the velocity approaches the Rayleigh speed (cf. [55]), and the use of higher-order elements. Our results indicate that the overshoot is not an artifact of the chosen definition for $\mathbf{x}_c$, nor is it due to a lifting-off mode of separation, as suggested in [19]. Since theoretical results forbid crack-tip velocities between the Rayleigh and dilatational wave speeds for mode-I propagating cracks [56], the overshoot reported in [19] is likely due to discretization error associated with the non-adaptive solution's inability to track the shrinking process zone.

Whether a Mode-I crack is able to attain velocities approaching the theoretical limiting speed, c_R, depends on the specific material system, the loading conditions under consideration and the time scale at which velocity is measured. Experimental observations in homogeneous materials with amorphous microstructures subjected to far-field loading show that time-averaged crack-tip velocities rarely exceed roughly $0.5c_R$ due to micro-cracking and branching [37, 57]. These studies are sometimes cited to support the contention that crack-tip velocities never approach the theoretical limit in brittle materials. However, these observations apply only to macroscopic, time-averaged velocities. Fine-scale measurements by Sharon and Fineberg [58] demonstrate that instantaneous crack-tip velocities do, in fact, approach the theoretical limit between intermittent micro-cracking events. They found that microcracking causes only a temporary drop in the primary crack-tip velocity as energy flowing into the tip is briefly redirected elsewhere.

The lower limit on the macroscopic, time-averaged crack velocity is not universal for all brittle material systems and for all loading conditions. Lee and Knauss [59, 60] report macroscopic crack velocities that approach c_R for mode-I cracks growing along weak joints between identical brittle solids as the ratio of the strength of the bonded interface to the strength of the surrounding material approaches zero. This limiting situation pertains to the present example, as we have made no provision for material failure in the bulk. Weak interfaces are not rare in practice. For example, they are commonly created in manufacturing processes simply by scoring the surface of an otherwise uniform plate. The form and intensity of the loading and the material microstucture are also important. Winkler *et al.* [61] observed supersonic crack growth along weak crystallographic planes when the crack tip is directly loaded by a laser-induced, expanding plasma.

DOI: 10.1002/nme

Computational studies of brittle fracture processes, including microcracking and branching, call for numerical models that capture reductions in process-zone size and quasi-singular response as crack-tip velocities approach c_R. The adaptive SDG cohesive model's ability to predict detailed response under these conditions is indicative of the method's robustness and reliability and suggests that it is a suitable tool for multi-scale scientific studies of dynamic fracture in brittle materials.

4.4. Non-planar crack growth: dewetting of a circular inclusion

This final example demonstrates the SDG cohesive model's ability to simulate mixed-mode fracture and non-planar crack growth. The analysis domain is a $4\,\text{mm} \times 4\text{mm}$ square comprised of a stiff, circular inclusion of radius 1 mm centered in a matrix of more flexible material. The matrix material properties are Young's modulus, $E = 3.24\,\text{GPa}$; The Poisson ratio, $\nu = 0.35$; and mass density, $\rho = 1190\,\text{kg/m}^3$. The inclusion has the same properties, except the elastic modulus is 32.4 GPa. This produces wave speeds that are faster than the matrix by a factor of $\sqrt{10}$. A cohesive interface bonds the matrix to the inclusion. We model the response of the cohesive interface with the Xu–Needleman description [19], with cohesive strength, $\sigma_C = 32.4\,\text{MPa}$, and critical separation distance for both the normal and tangential directions, $\delta = 4.0 \times 10^{-4}\,\text{mm}$. The cohesive interface extends all the way around the inclusion, except for an interval from 17° to 22°, measured clockwise from the top of the inclusion, where we impose traction-free conditions to model an initial flaw in the cohesive bond. We enforce symmetry boundary conditions along the left and right edges of the square domain to simulate an infinite strip of similar inclusions. We enforce plane-strain conditions throughout the model.

Figure 13 shows results for a prescribed velocity, applied uniformly along the top and bottom edges of the domain, whose magnitude ramps linearly over an interval of 0.1 μs from zero to a sustained value, $\bar{v} = 7\,\text{m/s}$. Figure 13(a) depicts a time slightly after the wavefronts reach the top and bottom edges of the inclusion. As is evident from the response along the bottom edge, the action of the primary wave alone does not fracture the undamaged cohesive interface. However, the upper primary wave does initiate a fracture as it interacts with the flaw near the top of the inclusion. Part of the energy of the primary wavefronts is reflected by the cohesive bi-material interface, and part is transmitted to the interior of the inclusion. The greater wave speeds within the inclusion cause the interior waves to advance ahead of the wavefronts in the matrix, as shown in Figure 13(a). The interior waves combine with the lower primary wave to nucleate a second crack when they reach the bottom portion of the cohesive interface. The upper and lower cracks each propagate dynamically in two directions along the cohesive interface until the inclusion is eventually completely debonded. Figure 13(b) shows the configuration of the upper and lower cracks at a later time, just prior to complete debonding. At this stage, some of the crack-tip speeds are sufficient to generate quasi-singular velocity fields. Neither crack path is symmetric due to the asymmetric placement of the initial flaw.

5. CONCLUSIONS

We have presented a new framework for implementing cohesive fracture models within a spacetime discontinuous Galerkin finite element method that provides a robust environment for modeling dynamic, cohesive fracture. The elementwise balance properties and adaptive analysis capabilities of the SDG method enable it to resolve crack-tip fields and sharp wave fronts, whether they be

DOI: 10.1002/nme

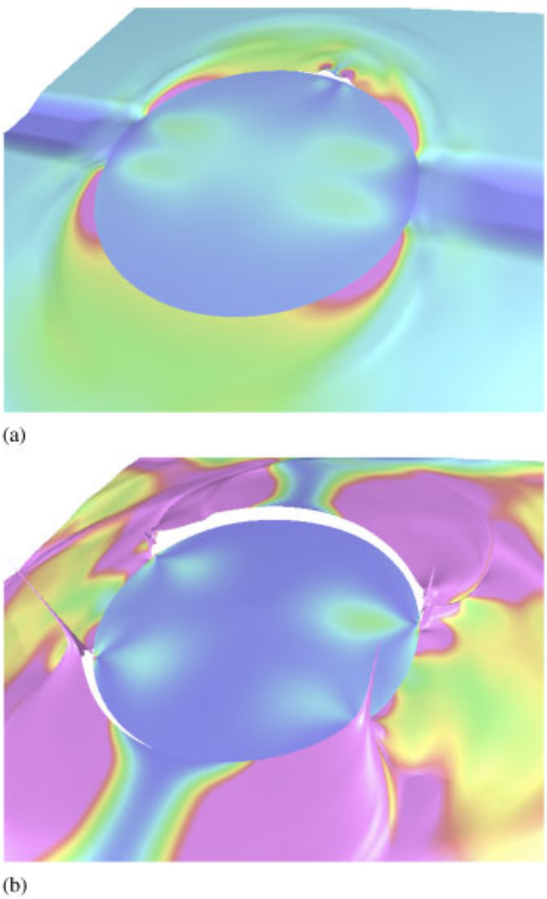

Figure 13. Non-planar crack growth: dewetting of a circular inclusion. Color depicts log of strain-energy density; height field is modulus of material velocity. (a) Fracture initiation at debonded defect; $t = 0.90\,\mu s$ and (b) late stage of dewetting with quasi-singular velocity response; $t = 1.90\,\mu s$.

due to shock loading, crack initiation and arrest events, or to waves emitted by a running crack. The use of discontinuous basis functions and the inclusion of target traction values in the SDG formulation simplify the implementation of cohesive TSLs; no special cohesive elements or other algorithmic devices are required.

An adaptive analysis capability is required to efficiently capture transient features, such as wavefronts and moving crack tips. The use of unstructured spacetime grids allows the SDG method to control both the spatial and temporal resolutions. The patch-by-patch solution scheme supports local error indicators and local adaptive meshing operations while maintaining linear computational complexity. Dual refinement indicators control numerical dissipation in the bulk material as well as the fracture-energy error on cohesive interfaces. This adaptive strategy automatically resolves the cohesive process zone to deliver stable numerical response for running cracks, even when the size of the process zone changes over time. Numerical studies demonstrate the convergence of

DOI: 10.1002/nme

various aspects of the fracture response as the adaptive tolerances are tightened. Using the two adaptive error indicators in combination substantially improves computational efficiency.

The SDG solution structure enables an analytical technique for computing crack-tip velocity from an implicit definition of the crack-tip position. This method neither masks numerical noise nor filters out meaningful high-frequency response. Thus, it promotes more objective evaluations of solution quality and convergence and reveals the full detail of the simulated response.

Our numerical solutions reveal new detail and unexpected qualitative features of cohesive, elastodynamic fracture. The presence of a quasi-singular velocity field and the fact that the crack-tip velocity does not exceed the Rayleigh wave speed, even under large impact velocities, are the most notable. Comparisons with previous solutions indicate that careful control of the discretization error is necessary to simulate accurately and reliably the dynamics of moving crack tips. For example, the adaptive SDG model's ability to capture the energetics of the quasi-singular velocity field surrounding running crack tips probably explain the differences in crack kinetics between our results and previous solutions. Our new findings are due entirely to the improved accuracy and resolution made possible by the adaptive SDG implementation since the underlying continuum model is the same as in previous studies. Further investigations of the structure and effects of the quasi-singular velocity field will be reported in a separate paper.

Our numerical experience suggests various modifications of our SDG implementation to improve efficiency. First, our h-adaptive method could be promoted to a hp-adaptive strategy to exploit the SDG formulation's ability to accommodate distinct polynomial orders in adjacent elements. Quadrature-free methods could reduce the cost of assembling the patch equations [62], and parallel solution methods could be applied to our patchwise meshing and solution processes. A parallel implementation, based on the ParFUM and CHARM++ frameworks [63, 64], is under development.

The implementation of cohesive models described in this paper is restricted to predefined cohesive surfaces. In continuing work, we are extending the adaptive SDG method to simulate crack nucleation and crack propagation along arbitrary, solution-dependent paths. Rather than introduce discontinuity surfaces on element interiors, as in [22, 21, 23], we use adaptive spacetime meshing operations to generate new cohesive surfaces in the SDG discretization that align with evolving crack trajectories.

Various extensions to the SDG mechanics formulation and to the cohesive model are needed to broaden our simulation capabilities. Provisions for finite-deformation response, in both the bulk material and cohesive models, are necessary for many studies. Rate-dependent and thermomechanical cohesive models [35, 36] provide more accurate descriptions of certain fracture processes where high strain rates accompanied by significant temperature increases have been observed [38, 65]. Various alternatives to the Xu–Needleman cohesive model are better suited to tracking solution-dependent crack paths, such as those with initially rigid behavior and inelastic unloading [28, 31]. Another possibility is to use damage models to evolve the cohesive properties, as in [66]. The modular structure of our implementation promotes the development and investigation of these alternative models.

Three-dimensional geometry is an important consideration in many fracture applications where two-dimensional idealizations are inadequate. Thus, we are interested in extending our implementation to $3D \times$time. The mechanics theory and finite element formulation presented here are fully extensible to three spatial dimensions and present no special difficulties. The primary challenge involves spacetime meshing. For non-adaptive meshing, where the spatial discretization is fixed, the mathematical theory that underlies Tent Pitcher is provably extensible to any

DOI: 10.1002/nme

dimension. However, the extension of certain aspects of the h-adaptive meshing algorithm remain open for future research. Although the task of building a fully three-dimensional code should not be underestimated, the problem is not as daunting as it might seem. Tent Pitcher's primary concern is with modeling the current front, which consists of a conventional three-dimensional space mesh augmented with a time coordinate at each vertex. True spacetime elements are only generated when a vertex advances in time and are not part of the meshing algorithm's fundamental data structure. Thus, there is some reason for optimism that this goal can be realized. Visualization of spacetime data need not be significantly more difficult than in other three-dimensional models. By passing a constant-time plane through the spacetime data set, we obtain a conventional three-dimensional model for visualization. If successfully extended, the powerful adaptive capabilities and intrinsic scalability of the SDG approach should prove beneficial in attaining the level of accuracy and efficiency required for reliable dynamic fracture simulations in three dimensions.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the contributions of Shuo-Heng Chung, Scott Miller, Jeff Erickson, Yong Fan, Michael Garland, Jayandran Palaniappan, Laxmikant Kale, Shripad Thite, Aaron Becker and Yuan Zhou to this work. Support from the Center for Process Simulation and Design (CPSD) and the Center for Simulation of Advanced Rockets (CSAR) at the University of Illinois is gratefully acknowledged. The U.S. National Science Foundation supports research in CPSD via grant NSF DMR 01-21695. The CSAR research program is supported by the U.S. Department of Energy through the University of California under subcontract B341494.

REFERENCES

1. Wilkins SJ, Schultz RA. 3d cohesive end-zone model for source scaling of strike-slip interplate earthquakes. *Bulletin of the Seismological Society of America* 2005; **95**(6):2232–2258.
2. Kiyono SS, Yonehara K, Graf RS, Howell WJ. Consideration of mechanical chip crack on fbga packages. *2001 Proceedings. Fifty-first Electronic Components and Technology Conference* (*Cat. No.01CH37220*), San Diego, CA, U.S.A., 2001; 193–197.
3. Pandolfi A, Ortiz M. An efficient adaptive procedure for three-dimensional fragmentation simulations. *Engineering with Computers* 2002; **18**:148–159.
4. Denoual C, Hild F. A damage model for the dynamic fragmentation of brittle solids. *Computer Methods in Applied Mechanics and Engineering* 2000; **183**:247–258.
5. Geubelle PH, Baylor JS. Impact-induced delamination of composites: a 2d simulation. *Composites Part B* 1998; **29B**:589–602.
6. Gasser TC, Holzapfel GA. A numerical framework to model 3-d fracture in bone tissue with application to failiure of the proximal femur. *Proceedings of the IUTAM Symposium*, *Symposium on Discretization Methods for Evolving Discontinuities*, Lyon, France, 2006; 199–212.
7. Johnson E. Process region changes for rapidly propagating cracks. *International Journal of Fracture* 1992; **55**:47–63.
8. Oliver J, Huespe AE, Samaniego E. A study on finite elements for capturing strong discontinuities. *International Journal for Numerical Methods in Engineering* 2003; **56**:2135–2161.
9. Abraham FF, Brodbeck D, Rafey RA, Rudge WE. Instability dynamics of fracture: a computer simulation investigation. *Physical Review Letters* 1994; **73**:272–276.
10. Nakano A, Kalia RK, Vashishta P. Dynamics and morphology of brittle cracks: a moleculardynamics study of silicon nitride. *Physical Review Letters* 1995; **75**:3138–3141.
11. Buehler MJ, Alexander H, Gao H, Duchaineau M, Abraham FF, Brodbeck D, Ridge WE, Xu X. Atomic plasticity: description and analysis of a one-billion atom simulation of ductile materials failure. *Computer Methods in Applied Mechanics and Engineering* 2004; **193**:5257–5282.

DOI: 10.1002/nme

12. Buehler MJ, Hartmaier A, Duchaineau MA, Abraham FF, Gao H. The dynamical complexity of work-hardening: a large-scale molecular dynamics simulation. *Acta Mechanica Sinica* (*English Series*) 2005; **21**(2):103–111.
13. Ravi-Chandar K. Dynamic fracture of nominally brittle materials. *International Journal of Fracture* 1998; **90**:83–102.
14. Marder M, Gross SP. Origin of crack tip instabilities. *Journal of the Mechanics and Physics of Solids* 1995; **43**:1–48.
15. Anderson H. A finite element representation of stable crack growth. *Journal of the Mechanics and Physics of Solids* 1973; **21**:337–356.
16. Yatomi M, Bettinson AD, O'Dowd NP, Nikbin KM. Modelling of damage development and failure in notched-bar multiaxial creep tests. *Fatigue and Fracture of Engineering Material and Structures* 2004; **27**(4):283–295.
17. Dugdale DS. Yielding of steel sheets containing slits. *Journal of the Mechanics and Physics of Solids* 1960; **8**:100–104.
18. Barenblatt GI. The mathematical theory of equilibrium of cracks in brittle fracture. *Advanced Applied Mechanics* 1962; **7**:55–129.
19. Xu XP, Needleman A. Numerical simulations of fast crack growth in brittle solids. *Journal of the Mechanics and Physics of Solids* 1994; **42**:1397–1434.
20. Belytschko T, Black T. Elastic crack growth in finite elements with minimal remeshing. *International Journal for Numerical Methods in Engineering* 1999; **45**:601–620.
21. Moes N, Dolbow J, Belytschko T. A finite element method for crack growth without remeshing. *International Journal for Numerical Methods in Engineering* 1999; **46**:131–150.
22. Dolbow J, Moes N, Belytschko T. Discontinuous enrichment in finite elements with a partition of unity method. *International Journal for Numerical Methods in Engineering* 2000; **36**(3):235–260.
23. Remmers JJC, de Borst R, Needleman A. A cohesive segments method for the simulation of crack growth. *Computational Mechanics* 2003; **31**:69–77.
24. Strouboulis T, Babuška I, Copps K. The design and analysis of the generalized finite element method. *Computer Methods in Applied Mechanics and Engineering* 2000; **181**:43–69.
25. Strouboulis T, Copps K, Babuška I. The generalized finite element method. *Computer Methods in Applied Mechanics and Engineering* 2001; **190**:4081–4193.
26. Freund LB. *Dynamic Fracture Mechanics*. Cambridge University Press: Cambridge, U.K., 1990.
27. Tvergaard V, Hutchinson JW. The relation between crack growth resistance and fracture process parameters in elastic–plastic solids. *Journal of the Mechanics and Physics of Solids* 1992; **40**:1377–1397.
28. Camacho GT, Ortiz M. Computational modelling of impact damage in brittle materials. *International Journal of Solids and Structures* 1996; **33**:2899–2938.
29. Needleman A. A continuum model for void nucleation by inclusion debonding. *Journal of Applied Mechanics* 1987; **54**:525–532.
30. Xu XP, Needleman A. Void nucleation by inclusion debonding in a crystal matrix. *Modelling and Simulation in Materials Science and Engineering* 1993; **1**:111–132.
31. Ortiz M, Pandolfi A. Finite-deformation irreversible cohesive elements for three-dimensional crack-propagation analysis. *International Journal for Numerical Methods in Engineering* 1999; **44**:1267–1282.
32. Matouš K, Kulkarni M, Geubelle P. Multiscale cohesive failure modeling of heterogeneous adhesives. *Journal of the Mechanics and Physics of Solids* 2008; **56**:1511–1533.
33. Li H, Chandra N. Analysis of crack growth and crack-tip plasticity in ductile materials using cohesive models. *International Journal of Plasticity* 2003; **19**:849–882.
34. Pandolfi A, Krysl P, Ortiz M. Finite element simulations of ring expansion and fragmentation: the capturing of length and time scales through cohesive models of fracture. *International Journal of Fracture* 1999; **95**:279–297.
35. Costanzo F, Walton JR. Steady growth of a crack with a rate and temperature sensitive cohesive zone. *Journal of the Mechanics and Physics of Solids* 2002; **50**:1649–1679.
36. Zhou F, Molinari JF, Shioya T. A rate-dependent cohesive model for simulating dynamic crack propagation in brittle materials. *Engineering Fracture Mechanics* 2005; **72**:1383–1410.
37. Fineberg J, Marder M. Instability in dynamic fracture. *Physics Reports* 1999; **313**:1–108.
38. Zehnder AT, Rosakis AJ. On the temperature distribution at the vicinity of dynamically propagating cracks. *Journal of the Mechanics and Physics of Solids* 1991; **39**:385–415.
39. Fagerström M, Larsson R. Theory and numerics for finite deformation fracture modeling using strong discontinuities. *International Journal for Numerical Methods in Engineering* 2006; **66**:911–948.

DOI: 10.1002/nme

40. Petracovici B. Analysis of a spacetime discontinuous Galerkin method for elastodynamics. *Ph.D. Thesis*, Department of Mathematics, University of Illinois at Urbana-Champaign, 2004.
41. Abedi R, Haber RB, Petracovici B. A spacetime discontinuous Galerkin method for elastodynamics with element-level balance of linear momentum. *Computer Methods in Applied Mechanics and Engineering* 2006; **195**:3247–3273.
42. Abedi R, Haber RB, Thite S, Erickson J. An *h*-adaptive spacetime-discontinuous Galerkin method for linearized elastodynamics. *Revue Européenne de Mécanique Numérique* (*European Journal of Computational Mechanics*) 2006; **15**(6):619–642.
43. Erickson J, Guoy D, Ungor A, Sullivan J. Building spacetime meshes over arbitrary spatial domains. *Proceedings of the 11th International Meshing Roundtable*, Ithaca, NY, U.S.A., 2002; 391–402.
44. Abedi R, Chung SH, Erickson J, Fan Y, Garland M, Guoy D, Haber RB, Sullivan J, White S, Zhou Y. Space–time meshing with adaptive refinement and coarsening. *Proceedings of the 20th Annual ACM Symposium on Computational Geometry*, Association for Computing Machinery, 2004; 300–309.
45. Adouani H, Tie B, Berdin C, Aubry D. Adaptive numerical modeling of dynamic crack propagation. *Journal de Physique IV* (*Proceedings*) 2006; **134**:501–506.
46. Arnold VI. *Mathematical Methods of Classical Mechanics* (2nd edn). Springer: New York, 1989.
47. Spivak M. *Calculus on Manifolds*. W. A. Benjamin: New York, 1965.
48. Bishop RL, Goldberg SI. *Tensor Analysis on Manifolds*. Prentice-Hall: Englewood Cliffs, NJ, 1980. Reprinted By Dover (1980).
49. Fleming WH. *Functions of Several Variables*. Addison-Wesley: Reading, MA, 1964.
50. Palaniappan J, Miller ST, Haber RB. Sub-cell shock capturing and spacetime discontinuity tracking for nonlinear conservation laws. *International Journal for Numerical Methods in Fluids* 2008; **57**:1115–1135.
51. Morrissey JW, Rice JR. Crack front waves. *Journal of the Mechanics and Physics of Solids* 1998; **46**:467–487.
52. Needleman A, Rosakis AJ. The efect of bond strength and loading rate on the conditions governing the attainment of intersonic crack growth along interfaces. *Journal of the Mechanics and Physics of Solids* 1999; **47**:2411–2449.
53. Zhou Y, Garland M, Haber RB. Pixel-exact rendering of spacetime finite element solutions. *Proceedings of IEEE Visualization 2004*. Institute of Electrical and Electronics Engineers, Inc., 2004; 425–432.
54. Matous K, Geubelle PH. Finite element formulation for modeling particle debonding in reinforced elastomers subjected to finite deformations. *Computer Methods in Applied Mechanics and Engineering* 2006; **196**(1–3): 620–633.
55. Yu H, Suo Z. Intersonic crack growth on an interface. *Proceedings of the Royal Society* 2000; **456**(1993):223–246.
56. Broberg KB. The near-tip field at high crack velocities. *International Journal of Fracture* 1989; **39**:1–13.
57. Sharon E, Gross S, Fineberg J. Energy dissipation in dynamic fracture. *Physical Review Letters* 1996; **76**:2117.
58. Sharon E, Fineberg J. Confirming the continuum theory of dynamic brittle fracture for fast cracks. *Nature* 1999; **397**(6717):333–335.
59. Lee O, Knauss W. Dynamic crack propagation along a weakly bonded plane in a polymer. *Experimental Mechanics* 1989; **29**:342–345.
60. Lee O, Knauss W. A reconciliation of dynamic crack velocity and Rayleigh wave speed in isotropic brittle solids. *International Journal of Fracture* 1994; **65**:97–114.
61. Winkler S, Shockey D, Curran D. Crack propagation at supersonic velocities—I. *International Journal of Fracture Mechanics* 1970; **6**:151–158.
62. Atkins HL, Shu CW. Quadrature-free implementation of discontinuous Galerkin method for hyperbolic equations. *American Institute of Aeronautics and Astronautics Journal* 1998; **36**:775–782.
63. Lawlor OS, Chakravorty S, Wilmarth TL, Choudhury N, Dooley I, Zheng G, Kale LV. ParFUM: a parallel framework for unstructured meshes for scalable dynamic physics applications. *Engineering with Computers* 2006; **22**:215–235.
64. Zheng G. Achieving high performance on extremely large parallel machines: performance prediction and load balancing. *Ph.D. Thesis*, Department of Computer Science, University of Illinois at Urbana-Champaign, 2005.
65. Tzou DY. Experimental evidence for the temperature waves around a rapidly propagating crack tip. *Journal of Heat Transfer* 1992; **114**:1042–1051.
66. Allix O, Corigliano A. Modeling and simulation of crack propagation in mixed modes interlaminar fracture. *International Journal of Fracture* 1996; **77**:111–140.

DOI: 10.1002/nme