A scalable 3D fracture and fragmentation algorithm based on a hybrid, discontinuous Galerkin, Cohesive Element Method

R. Radovitzky [a,*], A. Seagraves [a], M. Tupek [a], L. Noels [b]

[a] *Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139, U.S.A.*

[b] *University of Liège, Computational & Multiscale Mechanics of Materials, B-4000 Liège, Belgium*

Abstract

A scalable algorithm for modeling dynamic fracture and fragmentation of solids in three dimensions is presented. The method is based on a combination of a discontinuous Galerkin (DG) formulation of the continuum problem and Cohesive Zone Models (CZM) of fracture. Prior to fracture, the flux and stabilization terms arising from the DG formulation at interelement boundaries are enforced via interface elements, much like in the conventional intrinsic cohesive element approach, albeit in a way that guarantees consistency and stability. Upon the onset of fracture, the traction-separation law (TSL) governing the fracture process becomes operative without the need to insert a new cohesive element. Upon crack closure, the reinstatement of the DG terms guarantee the proper description of compressive waves across closed crack surfaces.

The main advantage of the method is that it avoids the need to propagate topological changes in the mesh as cracks and fragments develop, which enables the

indistinctive treatment of crack propagation across processor boundaries and, thus, the scalability in parallel computations. Another advantage of the method is that it preserves consistency and stability in the uncracked interfaces, thus avoiding issues with wave propagation typical of intrinsic cohesive element approaches.

A simple problem of wave propagation in a bar leading to spall at its center is used to show that the method does not affect wave characteristics and as a consequence properly captures the spall process. We also demonstrate the ability of the method to capture intricate patterns of radial and conical cracks arising in the impact of ceramic plates which propagate in the mesh impassive to the presence of processor boundaries.

Key words: Dynamic brittle fracture, Cohesive zone models, Discontinuous Galerkin methods, Parallel computing

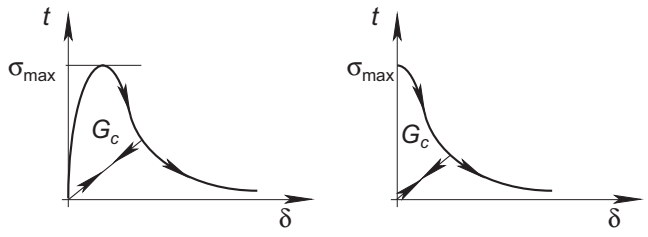
1 Introduction

The damage and failure of brittle materials subjected to intense loads is characterized by the development of intricate patterns of three-dimensional cracks, especially in the case of localized impact loading, e.g. (1; 2; 3). The computational characterization of these three-dimensional aspects of brittle damage processes is critical in many applications including armor materials (4) and orbital debris mitigation (5). One class of approaches which has shown promise for modeling brittle fracture processes is based on the so-called "discrete crack" model of fracture. In the discrete crack approach, the initiation and propagation of cracks is modeled explicitly by introducing surfaces of discontinuity

* Corresponding author. Tel: +1-617-252-1518, Fax: +1-617-253-0361, E-mail: rapa@mit.edu

within the material. The fracture processes at these surfaces of discontinuity can be described by cohesive zone models (CZM) of fracture (6; 7) via a phenomenological traction-separation law (TSL). The most popular implementation of this concept is the so-called "cohesive element" method (see (8) for a recent review) in which crack openings are represented as displacement jumps at the inter-element boundaries using "interface" or "cohesive" finite elements. Simulations using cohesive element methods suffer from a well-known mesh dependency as the possible crack nucleation sites and propagation paths are constrained by the finite element discretization (9; 10; 11; 12). This hinders the ability to describe complex crack patterns arising in three dimensional problems. A possible avenue for mitigating mesh dependency in cohesive element methods is to employ very fine meshes. As the mesh becomes finer, the available sites for crack nucleation and propagation increases, thus reducing crack path dependence on the mesh. Mesh refinement is also critical for resolving the size of the fracture process zone which is exceedingly small in brittle materials. The need for highly refined meshes demands parallel computational schemes that are scalable to large numbers of processors for problems of increasing size, especially in three dimensional problems. However, owing to fundamental issues in the formulation and implementation of conventional cohesive element methods, scalable three-dimensional algorithms for fracture and fragmentation are yet to be demonstrated.

In the original implementation of the cohesive element method for modeling brittle crack propagation in elastic materials (commonly known as the "intrinsic" approach, Figure 1a), cohesive surfaces are assumed to have an initially elastic response. This class of cohesive laws are implemented by creating the interface element data structures at all the interelement boundaries prior to



(a) Intrinsic approach (b) Extrinsic approach

Fig. 1. Surface traction t in terms of opening δ for the cohesive laws utilized in the a) the intrinsic approach, and b) the extrinsic approach. The cohesive law is typically characterized by the fracture energy G_c and the cohesive strength $\sigma_{\max}$.

the calculation (13; 14). Due to the fact that surfaces of discontinuity are initially present at all possible crack initiation sites, the intrinsic approach is potentially scalable. Indeed, in (10) Xu et al developed a 2D parallel implementation of the intrinsic approach and demonstrated parallel calculations on up to thirty six processors. A well-known problem with the intrinsic approach is that the initially elastic response of the intrinsic TSL necessarily alters the effective elastic response of the material prior to the onset of fracture, a phenomenon called "artificial compliance". This spurious behavior affects the propagation of elastic stress waves in the uncracked continuum (15) and has been shown to be mesh dependent (16). The only way to reduce the effect of artificial compliance is by increasing the initial elastic slope of the cohesive law, leading to severe stable time step restrictions which render the intrinsic approach unsuitable for explicit dynamic calculations (15) or to ill-conditioning of consistent Jacobians in static or implicit dynamics calculations, (17).

A second class of cohesive methods fall under the so-called "extrinsic" approach in which cohesive elements are inserted at interelement boundaries only when a specified fracture criterion is met, (18; 19; 20). In this case, the TSL effectively has an initially rigid response (Figure 1b), thus avoiding artifi-

cial compliance issues prior to fracture. It should be noted, however, that this issue also arises in the extrinsic approach if there is a compressive reloading of the cohesive elements, e.g. upon crack closure. In the extrinsic method, the insertion of cohesive elements is effected on-demand and incrementally through dynamic modification of data structures describing the topology of the mesh, a process which is algorithmically complex, especially in three dimensions.

The first sequential implementation of the adaptive insertion concept in three dimensions was presented by Pandolfi and Ortiz, (21; 22) and used data structures in which all of the relevant mesh entities and adjacency relationships are explicitly stored to facilitate the modification of the mesh topology. Ortiz's group, (23), recently showed that the computational complexity of this mesh topology modification algorithm is unnecessarily exorbitant ($O\left(N_I^{1.9}\right)$, where N_I is the number of interface elements), and proposed an alternative insertion scheme based on a graph representation of the finite element mesh which grows linearly in the number of interface elements. To address the same issues, Paulino *et al*, (24), proposed a different data structure and algorithmic approach also of linear complexity.

It bears emphasis that the totality of these data structures and algorithms are restricted in their application to the case of single processor calculation. The extension of these algorithms to the widely adopted scenario of parallel finite element computations using partitioned meshes and communications based on message passing offers a considerably higher challenge. The main problem in the extension to the parallel case lies in the inherent difficulty associated with propagating topological changes in the mesh across processor boundaries. Just like with other algorithms in computational geometry involving non-local topological mesh changes, propagating these changes across processor bound-

aries leads to granular communications of topological data of *a priori* unspecified size and location, and is thus inherently unscalable. A promising solution to this difficulty can be found in a new extrinsic approach proposed recently by (25) in which extrinsic cohesive elements are inserted at all interelement boundaries prior to the calculation and remain dormant until fracture occurs, at which point they are activated. Since the cohesive elements are initially present at all possible crack nucleation sites, as for the intrinsic approach, this method can be parallelized in principle in a scalable fashion. Indeed, (25) presented a 2D parallel implementation of the method and demonstrated its scalability for a 1.2 M element mesh on up to 512 processors. The possibility of extending this approach to three dimensions is palpable, although it is yet to be demonstrated.

A variety of other methods have been proposed to model fracture and other discontinuities discretely while enabling arbitrary crack paths in simulations and reduce mesh dependency. The basic idea of these approaches is to allow surfaces of discontinuity to propagate through the interior of volumetric elements (see e.g. the embedded localization line method (26), (27; 28), the extended finite element method (XFEM) (29; 30; 31), and the cohesive segments method (32; 33)). This class of methods have significant potential to reduce mesh dependency by enabling arbitrary crack paths with relatively coarse meshes. However, fine meshes are still necessary to resolve the size of the fracture process zone in brittle materials. Scalable algorithms for this family of methods have not been demonstrated and are expected to exhibit an even higher level of complexity than the parallelization of cohesive element methods.

The purpose of the present paper is to develop a scalable cohesive element

approach for modeling dynamic fracture of solids in three dimensions. Toward this end, we propose a new method based on a discontinuous Galerkin (DG) reformulation of the continuum problem following (34; 35). The idea of using DG formulations in the description of fracture was originally proposed by Mergheim et al (36) who applied it to the problem of interface delamination in two dimensions. A recent paper also utilizes a space-time DG formulation to describe dynamic fracture for constrained crack paths in two dimensions (37). The method was applied to simulate mode I crack growth under impact loading and particle matrix decohesion using the exponential cohesive law of Xu and Needleman (14). However, the potential of this approach for modeling arbitrary dynamic crack propagation in three dimensions has not yet been realized. In the method we propose in this paper, interface elements are inserted at interelement boundaries at the beginning of the simulation, which proceeds using a DG approach. Consistency and stability of the finite element solution in the pre-fracture regime are guaranteed by additional terms in the weak statement of the problem which emerge naturally from the DG formulation. By stark contrast, intrinsic cohesive approaches violate both consistency and stability, which leads to a variety of numerical problems including "artificial compliance" and wave propagation issues (15). When the specified fracture criterion is met at an interelement boundary, the computation of the DG interface flux terms gives its place to an extrinsic cohesive law which describes the irreversible traction-separation response eventually leading to complete decohesion and the formation of new crack surfaces.

The proposed approach also addresses another important issue with conventional CZM which has been neglected in previous work, including in those methods that have been effectively parallelized in two dimensions, (10; 25).

In previous approaches, the post-fracture response in compression upon crack closure is handled through the TSL with a penalty-like stiffness approach, which also induces artificial compliance (i.e. violates consistency) and, thus, affects wave propagation. Accurate modeling of compressive wave propagation is particularly important in simulating the multi-hit impact response of protective structures (e.g. armor plates), as well as in other structures exhibiting progressive damage.

A unique advantage of the proposed method is that its parallel implementation follows directly from the parallel DG method for the uncracked continuum (35), without any need for special treatment of cracks propagating across processor boundaries. Topological changes to the mesh and modification to interprocessor communication maps are completely avoided as the mesh is fragmented initially by requirement of the DG method. Crack nucleation and propagation is effected entirely at the constitutive level when evaluating interelement boundary integrands at the quadrature points of the interface elements. As a consequence, the method also inherits the scalability properties of the DG method demonstrated in (35).

In section 2, the hybrid DG/cohesive framework is presented. The implementation of the method for parallel computation is discussed in section 2.4. Two numerical examples are presented in section 3 to demonstrate the features of the method and to illustrate parallel computations involving cracks which cross processor boundaries. The first problem corresponds to the propagation of a uniaxial tensile stress wave in a bar leading to spall at its center. This example is used to show that the method does not affect wave characteristics and as a consequence properly captures spall at the center plane. The second example involves the simulation of the impact of a ceramic plate and is used to

demonstrate the ability of the method to capture intricate patterns of radial and conical cracks arising in the impact of ceramic plates which propagate in the mesh impassive to the presence of processor boundaries.

2 Discontinuous-Galerkin Cohesive-Zone Modeling Framework

The formulation of the DG framework for the continuum problem follows closely the presentation in (34; 35; 8). For completeness, the main steps in the formulation are summarized below.

2.1 DG/CZ weak formulation

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

$$\rho_0 \ddot{\boldsymbol{\varphi}} = \boldsymbol{\nabla}_0 \cdot \mathbf{P} + \rho_0 \mathbf{B} \quad \text{in } B_0 \tag{1}$$

$$\boldsymbol{\varphi} = \bar{\boldsymbol{\varphi}} \quad \text{on } \partial_D B_0 \tag{2}$$

$$\mathbf{P} \cdot \mathbf{N} = \bar{\mathbf{T}} \quad \text{on } \partial_N B_0 \tag{3}$$

In these relations ρ_0 is the initial density, $\mathbf{P}$ is the first Piola-Kirchhoff stress tensor, $\mathbf{N}$ is the unit surface normal in the reference configuration.

The discontinuous Galerkin (DG) weak form of Eqs. (1-3) arises by seeking an elementwise-continuous polynomial approximation $\boldsymbol{\varphi}_h$ of the deformation over the discretization $B_{0h} = \bigcup_{e=1}^{E} \bar{\Omega}_0^e$ of B_0, where $\bar{\Omega}_0^e$ is the union of the open do-

main Ω_0^e with its boundary $\partial\Omega_0^e$, i.e. $\boldsymbol{\varphi}_h \notin C^0(B_{0h})$, as in continuous Galerkin approximations, but $\boldsymbol{\varphi}_h \in C^0(\Omega_0^e)$. Consequently, for a DG formulation the trial functions $\delta\boldsymbol{\varphi}_h$ are also discontinuous across the element interfaces on the internal boundary of the body $\partial_I B_{0h} = \left[\bigcup_{e=1}^{E} \partial\Omega_0^e\right] \backslash \partial B_{0h}$.

The new weak formulation of the problem is obtained in a similar way as for the continuous Galerkin approximation. The strong form (1) of the linear momentum balance is enforced in a weighted-average sense by multiplying by a suitable test function $\delta\boldsymbol{\varphi}_h$ and integrating in the domain. However, since both test and trial function are discontinuous, the integration by parts is not performed over the whole domain, but on each element instead, leading to

$$\sum_e \int_{\Omega_0^e} \left(\rho_0 \ddot{\boldsymbol{\varphi}}_h \cdot \delta\boldsymbol{\varphi}_h + \mathbf{P}_h : \boldsymbol{\nabla}_0 \delta\boldsymbol{\varphi}_h\right) dV - \sum_e \int_{\partial\Omega_0^e \cap \partial_I B_{0h}} \delta\boldsymbol{\varphi}_h \cdot \mathbf{P}_h \cdot \mathbf{N} dS = \sum_e \int_{\Omega_0^e} \rho_0 \mathbf{B} \cdot \delta\varphi_h dV + \sum_e \int_{\partial\Omega_0^e \cap \partial_N B_{0h}} \delta\boldsymbol{\varphi}_h \cdot \bar{\mathbf{T}} dS \quad (4)$$

In this equation the discretized stress tensor $\mathbf{P}_h$ results from the discretized deformation state $\mathbf{F}_h = \boldsymbol{\nabla}_0 \boldsymbol{\varphi}_h$ through a constitutive material law. Equation (4) can be written as

$$\int_{B_{0h}} \left(\rho_0 \ddot{\boldsymbol{\varphi}}_h \cdot \delta\boldsymbol{\varphi}_h + \mathbf{P}_h : \boldsymbol{\nabla}_0 \delta\boldsymbol{\varphi}_h\right) dV + \int_{\partial_I B_{0h}} [\![\delta\boldsymbol{\varphi}_h \cdot \mathbf{P}_h]\!] \cdot \mathbf{N}^- dS = \int_{B_{0h}} \rho_0 \mathbf{B} \cdot \delta\varphi_h dV + \int_{\partial_N B_{0h}} \delta\boldsymbol{\varphi}_h \cdot \bar{\mathbf{T}} dS \quad (5)$$

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

$$[\![\bullet]\!] = \left[\bullet^+ - \bullet^-\right] \quad (6)$$

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

that

$$\int_{\partial_I B_{0h}} [\![\delta\boldsymbol{\varphi}_h \cdot \mathbf{P}_h]\!] \cdot \mathbf{N}^- dS \rightarrow \int_{\partial_I B_{0h}} [\![\delta\boldsymbol{\varphi}_h]\!] \cdot \mathbf{h}\left(\mathbf{P}^+, \mathbf{P}^-, \mathbf{N}^-\right) dS \tag{7}$$

where $\mathbf{N}^-$ is the outward unit surface normal for a given element. Rewriting the term in question we find

$$\int_{\partial_I B_{0h}} [\![\delta\boldsymbol{\varphi}_h \cdot \mathbf{P}_h]\!] \cdot \mathbf{N}^- dS = \int_{\partial_I B_{0h}} [\![\delta\boldsymbol{\varphi}_h]\!] \cdot \langle \mathbf{P}_h \rangle \cdot \mathbf{N}^- dS + \int_{\partial_I B_{0h}} \langle \delta\boldsymbol{\varphi}_h \rangle \cdot [\![\mathbf{P}_h]\!] \cdot \mathbf{N}^- dS \tag{8}$$

where the average operator defined by

$$\langle \bullet \rangle = \frac{1}{2}\left[\bullet^+ + \bullet^-\right] \tag{9}$$

has been used. The last term in equation (8) can be neglected because the jump in $\mathbf{P}_h$ does not require penalization to ensure consistency. Hence, $\mathbf{h}$ is chosen to be

$$\mathbf{h}\left(\mathbf{P}^+, \mathbf{P}^-, \mathbf{N}^-\right) = \langle \mathbf{P}_h \rangle \cdot \mathbf{N}^- \tag{10}$$

This form of the numerical flux was proposed by Bassi and Rebay (38) in the first DG formulation concerning elliptic equations. Other forms for the numerical flux are possible and can be found in the work of Arnold *et al.* (39) and Brezzi *et al.* (40). Using the choice of numerical flux from equation (10), the weak formulation reduces to

$$\int_{B_{0h}} \left(\rho_0 \ddot{\boldsymbol{\varphi}}_h \cdot \delta\boldsymbol{\varphi}_h + \mathbf{P}_h : \boldsymbol{\nabla}_0 \delta\boldsymbol{\varphi}_h\right) dV + \int_{\partial_I B_{0h}} [\![\delta\boldsymbol{\varphi}_h]\!] \cdot \langle \mathbf{P}_h \rangle \cdot \mathbf{N}^- dS = \int_{B_{0h}} \rho_0 \mathbf{B} \cdot \delta\varphi_h dV + \int_{\partial_N B_{0h}} \delta\boldsymbol{\varphi}_h \cdot \bar{\mathbf{T}} dS \tag{11}$$

Since the interelement displacement continuity is not enforced strongly in a DG formulation, it must be enforced weakly which, in turn, ensures stability of the numerical solution. To this end, the compatibility equation $\boldsymbol{\varphi}_h^- - \boldsymbol{\varphi}_h^+ = 0$ on $\partial_I B_{0h}$ is enforced through a (sufficiently large) quadratic stabilization term

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

$$\int_{B_{0h}} (\rho_0 \ddot{\boldsymbol{\varphi}}_h \cdot \delta\boldsymbol{\varphi}_h + \mathbf{P}_h : \boldsymbol{\nabla}_0 \delta\boldsymbol{\varphi}_h)\, dV + \int_{\partial_I B_{0h}} [\![\delta\boldsymbol{\varphi}_h]\!] \cdot \langle \mathbf{P}_h \rangle \cdot \mathbf{N}^- dS + \int_{\partial_I B_{0h}} \left\{ [\![\delta\boldsymbol{\varphi}_h]\!] \otimes \mathbf{N}^- : \langle \frac{\beta_s}{h_s} \mathbb{C} \rangle : [\![\boldsymbol{\varphi}_h]\!] \otimes \mathbf{N}^- \right\} dS = \int_{B_{0h}} \rho_0 \mathbf{B} \cdot \delta\varphi_h dV + \int_{\partial_N B_{0h}} \delta\boldsymbol{\varphi}_h \cdot \bar{\mathbf{T}} dS \quad (12)$$

where β_s plays the role of a penalty parameter. This formulation, known as the Interior Penalty Method, has been shown to be stable (for $\beta_s > 1$), consistent and to possess the optimal convergence rate in the energy norm (34).

The extension of this DG framework to explicit dynamics time integration including parallel implementation was presented in (35), where it was also shown that the stable time step is reduced by a factor of $\sqrt{\beta_s}$ as compared to a CG formulation, i.e

$$\Delta t < \Delta t_{\text{crit}} = \frac{h_s}{\sqrt{\beta_s} c} \quad (13)$$

where c is the sound speed of the material. More details concerning this approach, and in particular the numerical implementation based on interface elements can be found in (34; 35).

A dynamic simulation proceeds initially and prior to the nucleation of cracks according to the above DG framework. The onset of fracture is effected in

the same manner as in the extrinsic CZM approach, i.e. following a fracture stress criterion. Upon the nucleation of a crack at an interface element, the DG flux terms cease to operate and give place to the TSL governing the fracture process in the material. It should be noticed that this does not require any modifications of the mesh, but simply a change in the terms evaluated at the interface element integration points. Hence, if $\mathbf{T}$ is the surface traction resulting from the TSL in the reference configuration, equation (11) becomes

$$\begin{aligned}\int_{B_{0h}} \left(\rho_0 \ddot{\boldsymbol{\varphi}}_h \cdot \delta\boldsymbol{\varphi}_h + \mathbf{P}_h : \boldsymbol{\nabla}_0 \delta\boldsymbol{\varphi}_h\right) dV + \int_{\partial_I B_{0h}} \alpha \mathbf{T}\left(\llbracket \boldsymbol{\varphi}_h \rrbracket\right) \cdot \llbracket \delta\boldsymbol{\varphi}_h \rrbracket \, dS \\ + \int_{\partial_I B_{0h}} (1-\alpha) \llbracket \delta\boldsymbol{\varphi}_h \rrbracket \cdot \langle \mathbf{P}_h \rangle \cdot \mathbf{N}^- dS + \\ \int_{\partial_I B_{0h}} (1-\alpha) \llbracket \delta\boldsymbol{\varphi}_h \rrbracket \otimes \mathbf{N}^- : \langle \frac{\beta_s}{h_s} \mathbb{C} \rangle : \llbracket \boldsymbol{\varphi}_h \rrbracket \otimes \mathbf{N}^- dS \\ = \int_{B_{0h}} \rho_0 \mathbf{B} \cdot \delta\varphi_h dV + \int_{\partial_N B_{0h}} \delta\boldsymbol{\varphi}_h \cdot \bar{\mathbf{T}} dS \quad (14)\end{aligned}$$

In this equation α is a binary operator defined as $\alpha = 0$ before fracture and $\alpha = 1$ after the fracture stress criterion is met. It should be noted that this approach provides for the possibility of partially fractured interface elements as α may adopt different values at different quadrature point of each interface element.

Equation (14) can also help to understand some of the numerical problems in the intrinsic CZM. Consistency is a requirement for convergence which, in the DG formulation for the uncracked body ($\alpha = 0$), is satisfied by the DG flux term: $\llbracket \delta\boldsymbol{\varphi}_h \rrbracket \cdot \langle \mathbf{P} \rangle \cdot \mathbf{N}^-$, as shown in (34). By contrast, in the intrinsic CZM formulation the flux term is $\mathbf{T}\left(\llbracket \boldsymbol{\varphi}_h \rrbracket\right) \cdot \llbracket \delta\boldsymbol{\varphi}_h \rrbracket$, which violates consistency. In fact, as stated, this term corresponds to the stabilization term of the DG formulation, which only enforces consistency in the limit when the tangent in the reversible range of the TSL $\mathbf{T}(\llbracket \boldsymbol{\varphi}_h \rrbracket)$ tends to infinity (16). This creates

severe restrictions in the critical time step size in intrinsic CZM methods, a problem that is avoided in the discontinuous Galerkin method where the only penalty in the critical time step size is a factor of $\sqrt{\beta_s}$, as discussed above (13). In practice, $1 < \beta_s < 10$ is enough to ensure numerical stability (35), implying a minimal impact on the stable time step size.

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

2.2 Cohesive Law

The proposed approach is general in the sense that it does not rely on any particular assumption about the specific TSL employed in the description of fracture. It can thus be adapted to a wide class of brittle and ductile fracture

behavior. For definiteness, we here adopt the irreversible, linear softening cohesive law of (19). This particular TSL assumes that the cohesive behavior is isothermal, isotropic, and that the cohesive tractions depend on the local state of deformation at the crack tip only through the displacement jump. A simple formulation for modeling mixed-mode fracture, in accordance with these assumptions, is obtained by assuming that the cohesive free energy density depends on the surface opening vector $\boldsymbol{\Delta}$ only through an effective separation δ defined by

$$\delta = \sqrt{\gamma^2 \Delta_m^2 + \Delta_n^2} \tag{15}$$

In this expression, Δ_n is the positive normal separation along the unit local normal in the deformed configuration $\mathbf{n}$, Δ_m is the tangential separation along the unit local tangent in the deformed configuration $\mathbf{m}$, and γ is a parameter which assigns different weights to normal and tangential separations.

Under this set of assumptions, one can define an effective cohesive traction T per unit undeformed area which is given by

$$T = T\left(\delta, \mathbf{q}\right) \tag{16}$$

where $\mathbf{q}$ is a suitable set of internal variables describing the irreversible processes involved in decohesion. This law becomes operative at a quadrature point of an interface element when the fracture criterion

$$\sqrt{\left(\boldsymbol{\sigma}_h : [\mathbf{n} \otimes \mathbf{n}]\right)^2 + \gamma^{-2} \left(\boldsymbol{\sigma}_h : [\mathbf{n} \otimes \mathbf{m}]\right)^2} \geq \sigma_c \tag{17}$$

is satisfied. In this expression, $\boldsymbol{\sigma}_h$ is the Cauchy stress, and σ_c is the critical effective cohesive strength. Once the effective traction is determined, the components of the traction vector follow from

$$\mathbf{T} = \frac{T}{\delta} \left(\gamma^2 \Delta_m \mathbf{m} + \Delta_n \mathbf{n}\right) \tag{18}$$

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

$$T\left(\delta, \delta_{\max}\right) = \sigma_c \left(1 - \frac{\delta}{\delta_c}\right) \frac{ds}{dS} \quad \text{for } \dot{\delta} \geq 0,\ \delta = \delta_{\max} \tag{19}$$

where $\frac{ds}{dS}$ is the surface change between the deformed and reference configuration. Complete decohesion ($T \equiv 0$) occurs for $\delta \geq \delta_c$. In reference to Figure 2, the variable $\delta_{\max}$ is the maximum effective opening displacement and constitutes the internal variable describing irreversibility. For crack unloading, characterized by $\dot{\delta} < 0$ or $\delta < \delta_{max}$, the functional form of the linear irreversible softening law is assumed to follow a straight path to and from the origin yielding

$$T\left(\delta, \delta_{\max}\right) = \frac{T_{\max}}{\delta_{\max}} \delta \quad \text{for } \dot{\delta} < 0, \text{ or } \delta < \delta_{\max} \tag{20}$$

where $T_{\max}$ is the value of the effective traction at $\delta = \delta_{max}$. The work of separation (denoted by ϕ_{s}) for the linear softening law is simply

$$\phi_{\mathrm{s}} = \frac{1}{2} \sigma_c \delta_c \frac{ds}{dS} = G_c, \tag{21}$$

where G_c is the fracture energy, which can be obtained from experiments. Figure 2 depicts the $T - \delta$ relationship for the linear softening law, and assuming small deformation ($\frac{ds}{dS} = 1$). For crack closure, i.e. $\Delta_n = 0$ and $\dot{\Delta}_n < 0$, we assume that the normal response is governed by the continuum response, and therefore fall back to the DG form of the interface terms. This guarantees that compressive stress wave components can propagate across the closed crack surface as in the uncracked body. The tangential response is still governed by the TSL. Improvements on this model would incorporate a frictional component in the tangential response, but this has been neglected in the present work for simplicity and due to lack of experimental data supporting this need.

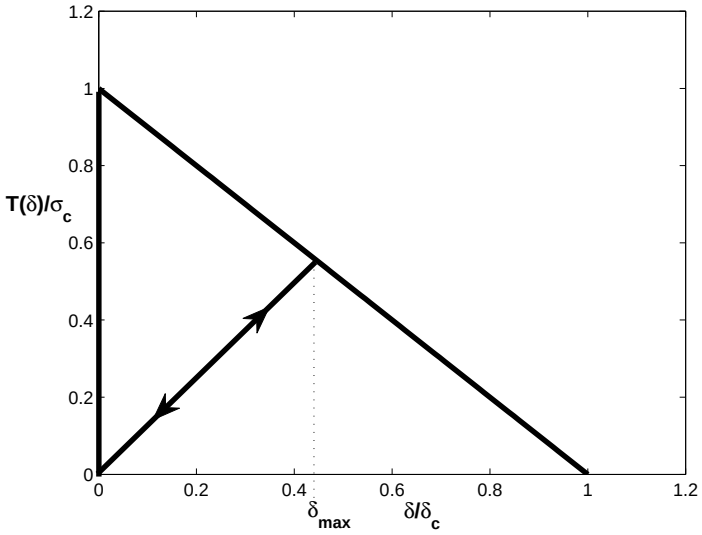


Fig. 2. $T - \delta$ relationship for the linear softening extrinsic law

2.3 Finite element discretization and explicit time integration

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

$$\boldsymbol{\varphi}_h(\mathbf{X}) = N_a(\mathbf{X})\,\mathbf{x}_a\ ,\ \delta\boldsymbol{\varphi}_h(\mathbf{X}) = N_a(\mathbf{X})\,\delta\mathbf{x}_a\ ,\ \ddot{\boldsymbol{\varphi}}_h(\mathbf{X}) = N_a(\mathbf{X})\,\ddot{\mathbf{x}}_a\ , \quad (22)$$

where N_a is the conventional shape function corresponding to node $a \in [1, N]$, $\mathbf{x}_a$ is the vector of current nodal positions, and N is the number of nodes. The weak form (14) therefore reduces to the set of ordinary differential equations:

$$M_{ab}\ddot{\mathbf{x}}_b + \boldsymbol{f}^{\mathrm{i}}_a(\mathbf{x}) + \boldsymbol{f}^{\mathrm{s}}_a(\mathbf{x}) = \boldsymbol{f}^{\mathrm{e}}_a,\ \ \forall t \in \mathrm{T}\ , \quad (23)$$

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

$$M_{ab}\ddot{\mathbf{x}}_b = \int_{B_{0h}} \rho_0 N_a N_b dV \;\; \ddot{\mathbf{x}}_b \;, \tag{24}$$

$$\boldsymbol{f}_a^{\mathrm{i}} = \int_{B_{0h}} \mathbf{P} : \boldsymbol{\nabla}_0 N_a dV \;, \tag{25}$$

$$\begin{aligned}\boldsymbol{f}_{a^\pm}^{\mathrm{s}} = &\pm \int_{\partial_I B_{0h}} (1-\alpha) \langle \mathbf{P} \rangle \cdot \mathbf{N}^- N_a^S dS \\ &\pm \int_{\partial_I B_{0h}} (1-\alpha) \left[\left\langle \frac{\beta_s}{h_s} \mathbb{C} \right\rangle : [\![\mathbf{x}_b]\!] \otimes \mathbf{N}^- \right] \cdot \mathbf{N}^- N_a^S N_b^S dS \\ &\pm \int_{\partial_I B_{0h}} \alpha \mathbf{T}\left([\![\mathbf{x}]\!]\right) N_a^S dS \;,\end{aligned} \tag{26}$$

$$\boldsymbol{f}_a^{\mathrm{e}} = \int_{B_{0h}} \rho_0 \mathbf{B} N_a dV + \int_{\partial_N B0h} \bar{\mathbf{T}} N_a^S dS \;, \tag{27}$$

where M_{ab} is the mass matrix, and $\pm$ refers to the boundaries of the two elements sharing the same interface.

The numerical implementation of the DG/cohesive framework is based on the use of interface elements introduced between the volumetric finite elements as depicted in Figure 3. The main advantage of using interface elements is the ability to integrate both the DG interface forces as well as the cohesive law (26). In this formulation, the conventional finite elements inside the volume of the domain can be used without modification. In this paper, 10-node quadratic tetrahedral elements are used, resulting in 12-node quadratic interface elements. Tetrahedral elements are integrated using a 4-point reduced quadrature rule, while the interface elements require full 6-point integration in order to prevent spurious penetration modes (35).

The interface elements are inserted between the two volume elements Ω_0^{e+} and Ω_0^{e-} by splitting the shared nodes, leading to independent problem unknowns. This new element encompasses the surface elements $\partial_I \Omega_0^{e+}$ and $\partial_I \Omega_0^{e-}$, which coincide in the reference configuration. In the reference configuration the interpolation of the position, the deformation mapping and its jumps are computed using the standard shape functions of the surface element $N_a^S(\boldsymbol{\xi}), a \in [1, n]$,

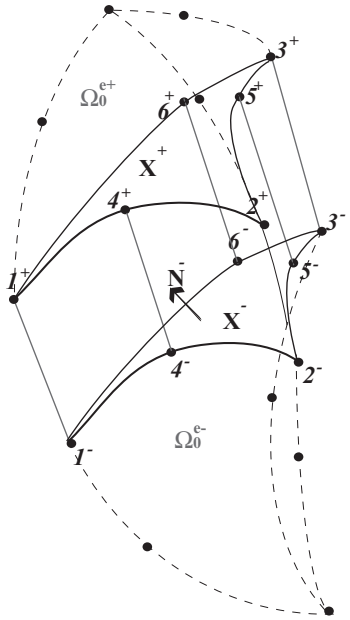


Fig. 3. Description of a 12-node interface element introduced between two 10-node quadratic tetrahedra Ω_0^{e+} and Ω_0^{e-}.

where $\boldsymbol{\xi} = (\xi_1, \xi_2)$ are the natural coordinates. For example, the reference configuration of the element is thus described by the expression:

$$\mathbf{X}^{\pm}(\boldsymbol{\xi}) = \sum_{a=1}^{n} N_a^S(\boldsymbol{\xi}) \mathbf{X}_{a^{\pm}} \,, \tag{28}$$

where $\mathbf{X}_{a^{\pm}}$, $a \in [1, 6]$ are the nodal coordinates of the surface elements in the reference configuration. The reference interelement outer surface normal $\mathbf{N}^-$ corresponding to element Ω_0^{e-} evaluated on the middle surface is obtained from the expression:

$$\mathbf{N}^-(\boldsymbol{\xi}) = \frac{\mathbf{G}_1(\boldsymbol{\xi}) \times \mathbf{G}_2(\boldsymbol{\xi})}{|\mathbf{G}_1(\boldsymbol{\xi}) \times \mathbf{G}_2(\boldsymbol{\xi})|} \,, \tag{29}$$

in which

$$\mathbf{G}_\alpha(\boldsymbol{\xi}) = \langle \mathbf{X}_{,\alpha} \rangle = \sum_{a=1}^{n} N_{a,\alpha}^S(\boldsymbol{\xi}) \, \langle \mathbf{X}_a \rangle \tag{30}$$

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

Time integration of the dynamics equations (23) is effected using a conventional second-order central-difference scheme (41) with mass lumping, which is suitable for the fast dynamic fracture processes of interest in this paper.

2.4 Parallel Implementation using MPI

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

(1) Partitioning of the original conforming finite element mesh into as many processors as form part of the calculation. For this step we utilize the popular METIS library. (42).

(2) Identification of the topology of the partitioned mesh to determine neighbors of each partition and generation of the nodal communication maps with neighbors. Although these are later discarded after the DG mesh is created, they provide very useful information to identify partition boundaries in the DG mesh.

(3) **Generation of the data structures of the new volumetric finite element mesh corresponding to the partition**. In the DG formulation, volumetric elements have their own nodes which are not shared with adjacent elements. This can be done in a straightforward manner from the continuous partition finite element mesh by iterating through all the elements, replicating all the original nodes shared with the adjacent elements and updating the connectivity table to point to the new node identifiers. It is then clear that a node in the original mesh gets replicated as many times as its number of adjacent volume elements. A map from the new to the old node ID is created as each new node is

generated for later convenience when generating the interface elements and the interprocessor communication maps.

(4) **Generation of interface elements in the interior of the partition mesh**. This can be efficiently accomplished in linear time on the total number of faces in the mesh by employing the WingedFace data structure (43) applied to the original partition mesh. For each face, this stores information about the adjacent volume elements and their face adjacency. Through the new volume element connectivity table one can then directly create the connectivity map for the interface elements. It should be noted that this step does not require the insertion of any extra nodes in the mesh.

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

(6) **Update of the interprocessor nodal communication maps**. In the framework proposed, the new nodal communication maps are simple to

obtain from the existing information of the interprocessor boundary faces. Remarkably, this step can be done without the need for any communications, barring an exceptional case as described below. The number of nodes involved in the pairwise map is equal to the number of faces on that interprocessor boundary multiplied by the number of nodes per face (six in the case of 10-node tetrahedral elements used here). A simple spatial sort procedure of the faces involved and their nodes is used to match up the nodal communication pairs while avoiding communications. The spatial sorting needs to be done in two levels (for all the nodes of each face first and then for face node groups), as there are spatially-coincident nodes in adjacent faces which can lead to incorrect pairwise node identification across the communication map due to machine round-off. The only exception which requires communication occurs when a partition boundary node is shared by more than one face of the same volumetric element. The face-based communication map then leads to a duplication of a node on the communication map. Fortunately, these nodes can be simply identified with minimal communication and either eliminated from the communication map or handled in the interprocessor assembly operation. We have chosen this latter approach for implementation efficiency. This requires a modification of the assembly operator. The modification of the assembly operator is described in the following.

(7) **Modified parallel assembly operator**. The usual approach of assembling global nodal quantities (e.g. masses, internal forces, etc.) by first adding the local processor contributions to all the processor nodes and then completing boundary-node values through an MPI reduce operation using the communication maps needs to be extended to contemplate the case of repeated nodes in the face-based communication maps. The ex-

tension involves simply pre-assembling the information in the identified repeated boundary nodes prior to MPI communication. This guarantees that the local nodes contain all the necessary information that needs to be exchanged with the neighbor.

It bears emphasis that the steps in the parallel implementation are independent of the specific finite and interface element type employed. The outlined procedure is then equally applicable to higher-order or hexahedral elements, etc. Additional aspects of the parallel DG implementation may be found in (35). However, it should be noted that the steps outlined above represent a significant improvement compared to the approach in this reference as it avoids the need of using and maintaining a global node ID, which, in turn, requires significant communication and spatial searches in the initialization stage of the overall algorithm.

Once the new communication maps and modified assembly operator are established, the calculation proceeds in the same fashion as the well-established and widely-adopted approach of MPI-based explicit finite element calculations (44; 45).

One of the unique aspects of the DG/CZM computational framework proposed is that the nucleation and propagation of cracks proceeds natural and impassively to the presence of processor boundaries without any extra operation, since the fracture modeling approach operates directly and exclusively at the constitutive level of the interface elements.

3 Numerical examples

3.1 Application: Longitudinal wave propagation and spall of an elastic bar

For a simple and exacting verification test of the hybrid DG/CZM method, we consider the spall of a bar which remains elastic up to a given fracture stress. Here, we use the term "spalling" to denote an internal fracture surface created by the interaction of two tensile stress waves to differentiate it from other types of fracture. Spall conditions at the center of the bar are induced by two longitudinal tensile waves originated at both bar ends by imposing a constant extensional velocity V_z. The waves travel toward the center at the longitudinal wave speed $c_d = \sqrt{\frac{\lambda+2\mu}{\rho_0}}$, where ρ_0 is the reference material density and λ and μ are the respectively the first and second Lamé constants, with a stress intensity $\sigma_w = \rho_0 c_d V_z$. The imposed velocity V_z is chosen such that the longitudinal stress is half the fracture strength $\sigma_w = 0.5\sigma_c$, which, in turn, causes the material to spall when the waves meet a the center of the bar. A schematic of the domain geometry is depicted in Figure 4.

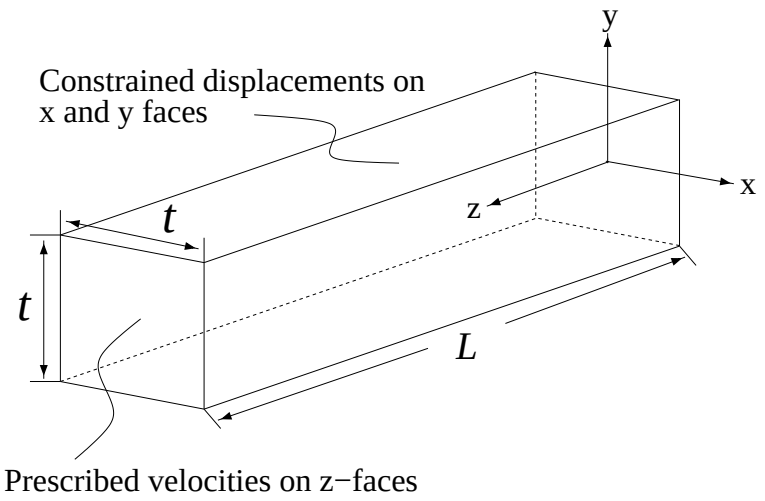


Fig. 4. Schematic of domain geometry and boundary conditions

The boundary and initial conditions for the z-faces correspond to a prescribed uniform velocity V_z:

$$-\dot{\boldsymbol{\varphi}}\left(x,y,0,t\right)=\dot{\boldsymbol{\varphi}}\left(x,y,L,t\right)=\left(0;\ 0;\ V_z\right)\quad\forall t\in[0,T] \tag{31}$$

Additionally, the x and y displacements are constrained on the lateral faces. In this wave propagation problem, the material response prior to fracture remains elastic and well into the infinitesimal strain limit. For convenience, an elastic neo-Hookean potential is used which reproduces the linear response in the limit of small deformations:

$$W=\left(\frac{\lambda}{2}\log J-\mu\right)\log J+\frac{\mu}{2}(I_1-3)\ , \tag{32}$$

where λ and μ are the Lame constants, $J=\det(\mathbf{F})$, and $I_1=\mathrm{tr}(\mathbf{C})$ for $\mathbf{F}=\nabla_0\boldsymbol{\varphi}$ and $\mathbf{C}=\mathbf{F}^{\mathrm{T}}\mathbf{F}$. The material properties of the uncracked continuum elements and the dimensions of the bar, as well as the applied velocity of the z-faces are given in Table 1.

3.2 Comparison of DG/CZM and CG/CZM Approaches

The problem is simulated both with the proposed DG/CZM approach and with the conventional intrinsic CG/CZM method for the purpose of comparison. The material properties chosen for the DG/cohesive interface elements are given in Table 2. The parameters are selected to be representative of alumina (critical stress and fracture energy, which determines the COD). However, the specific values are not really important as the example is used for verification purposes only and is not intended to reproduce any particular experiment.The solution is computed in parallel for the two approaches using the coarse mesh shown in Figure 5 which consists of 13683 tetrahedral elements or 130683

Properties	Values
Length	$L = 4.0$ mm
Thickness	$t = 0.4$ mm
Initial density	$\rho_0 = 3690$ kg·m^{-3}
Elastic Modulus	$E = 260$ GPa
Poisson's Ratio	$\nu = 0.21$
Longitudinal Wave Speed	$c_d = 8906$ m·s^{-1}
Velocity	$V_z = 6.086$ m·s^{-1}

Table 1

Simulation of spall of an elastic bar: bar dimensions, material properties and imposed velocity.

nodes.

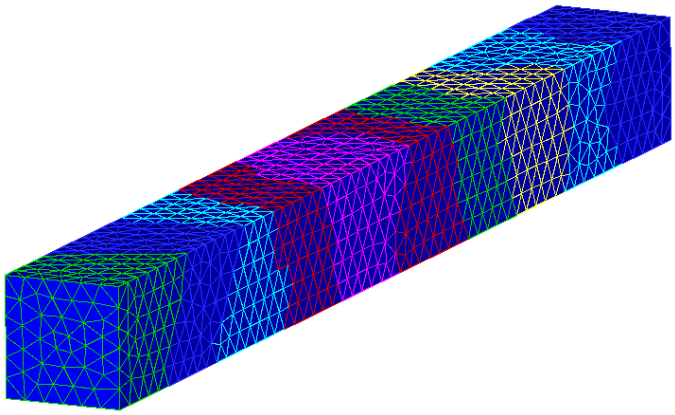


Fig. 5. Undeformed mesh consisting of 13,683 10-node tetrahedra showing the processor boundaries of the partitioned mesh for 10 processors

Figure 6 shows snapshots of axial stress in the bar at different times for the DG/CZM simulation. In Figure 6(a)-(b) the 200 MPa wave transmitted to the

Properties	Values
DG Stability Parameter	$\beta_s = 4.0$
Fracture stress	$\sigma_c = 400$ MPa
Fracture Energy	$\phi_{\rm s} = 34$ J·m^{-2}
Critical Opening Displacement	$\delta_c = 0.17$ μm
Weighting Parameter	$\gamma = 1.0$

Table 2

Simulation of spall of an elastic bar: properties for the DG elements and cohesive law.

bar by the external velocity applied to the bar ends can be seen propagating toward the center at times 0.08μs and 0.16μs, respectively. Figure 6(c) shows the stress distribution in the bar at time 0.32μs, slightly after the two incident waves reach the center at time 0.224μs. The zero stress at the center of the bar denotes the presence of a fracture plane which creates a stress relief wave that propagates outwardly toward the bar ends in both directions. The narrow higher-intensity precursor stress of about 300 MPa reflects the stress rise during the formation of the fracture surface. The later snapshot shown in Figure 6(d) shows the continuing unloading of the two bar halves after spall takes place.

In Figure 7 we compare the time history of stress σ_{zz} at the point $(x, y, z) = (0, 0, L/4 = 1.0mm)$ as predicted by the DG/cohesive approach against the theoretical result given by one-dimensional wave theory. The exact solution

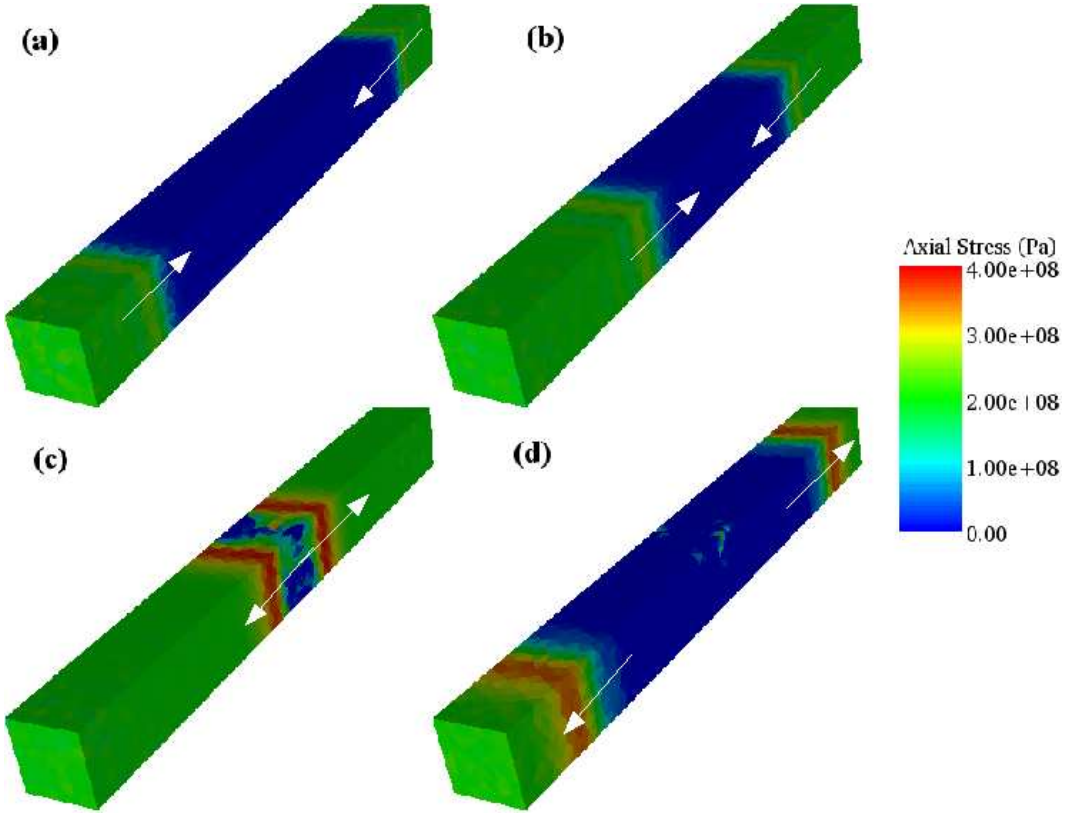


Fig. 6. Snapshots of wave propagation for DG/Cohesive formulation showing the incident waves at (a) 0.08 μs, (b) 0.16 μs and the reflection waves subsequent to the onset of spall fracture at (c) 0.32 μs, (d) 0.45 μs

corresponds to the idealized case of the wave propagating in a bar of half the length with a free surface at the end, i.e. it ignores the spall plane fracture process.

Examining Figure 7, we find that despite the coarseness of the mesh, the DG/CZM method accurately captures the speed and intensity of the incident stress wave. After the onset of fracture, the DG/cohesive formulation also captures the ensuing stress relief waves that reflect from the fracture surfaces. In Figure 8 we show: (a) an exterior close-up view of the fracture surface predicted by the DG/Cohesive scheme and (b) and internal view of the actual computed spall surface by plotting all the fully-fractured interface elements.

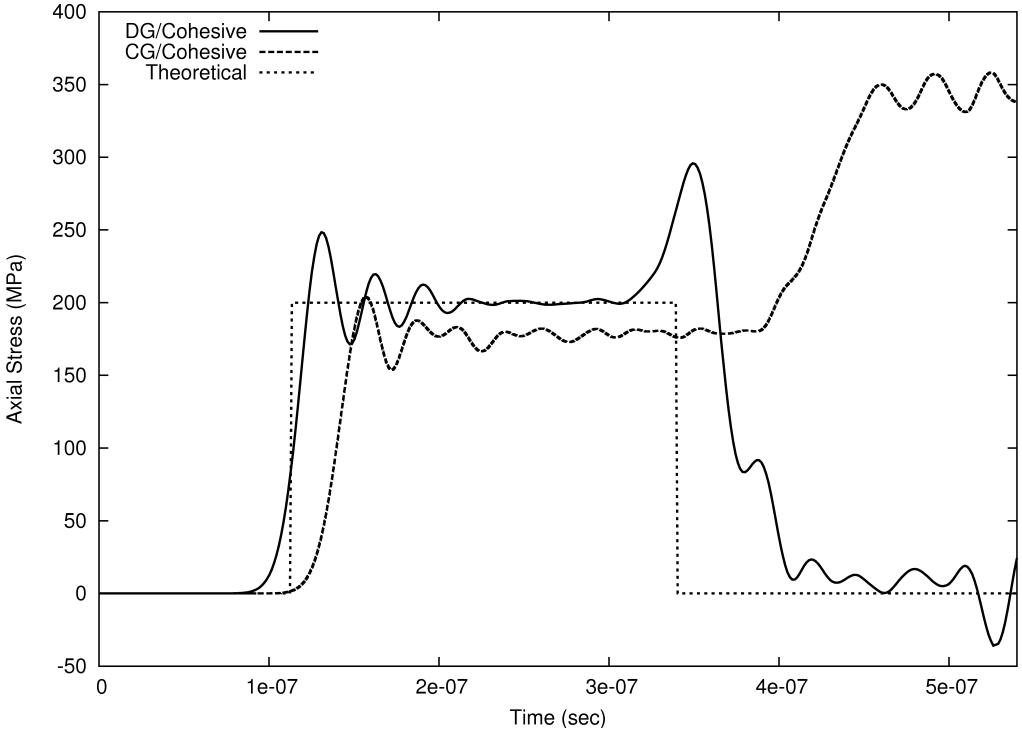
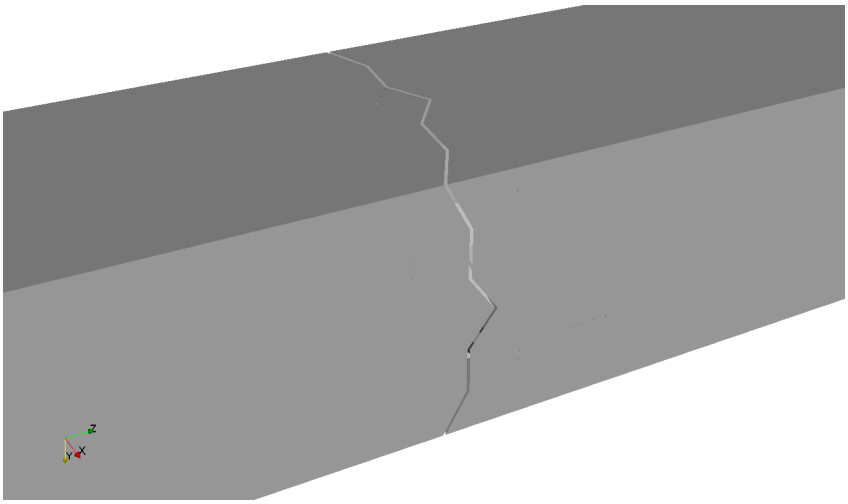
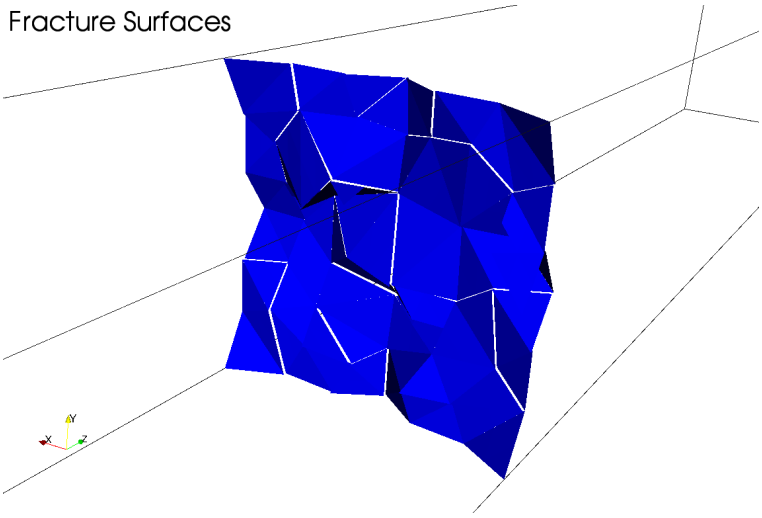


Fig. 7. σ_{zz} versus time for a point located at $(x, y, z) = (0, 0, 1.0mm)$

It can be observed in this figure that due to the unstructured character of the mesh, the exact location of the theoretical spall plane $z = L/2$ is not present as a possible locus of cracks and, as a result, the computed fracture plane exhibits a certain roughness. This underscores the problem of mesh dependency usually associated with cohesive element methods. However this problem can be alleviated by mesh refinement as shown in Figure 9, where the calculation has been repeated with a selectively refined mesh of 150742 tetrahedral elements or roughly 1.5 million nodes, Figure 9a. In this case, the spall surface much better approximates the flat spall plane at the bar center, Figure 9b. Further, the simulation was repeated with a mesh of 223,294 elements or 2.2M nodes, designed to contain the $z = L/2$-plane in its set of interface elements, Figure 10(a). In this case, the spall plane at the center was captured exactly, as expected, Figure 10b.



(a) External trace of computed spall surface

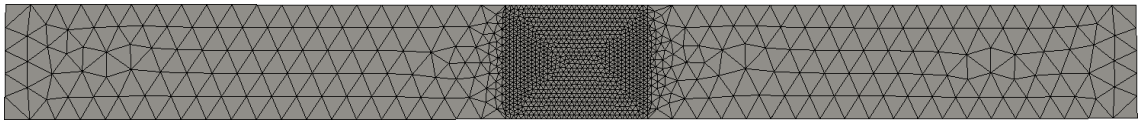


(b) Fully fractured interface elements defining the spall surface

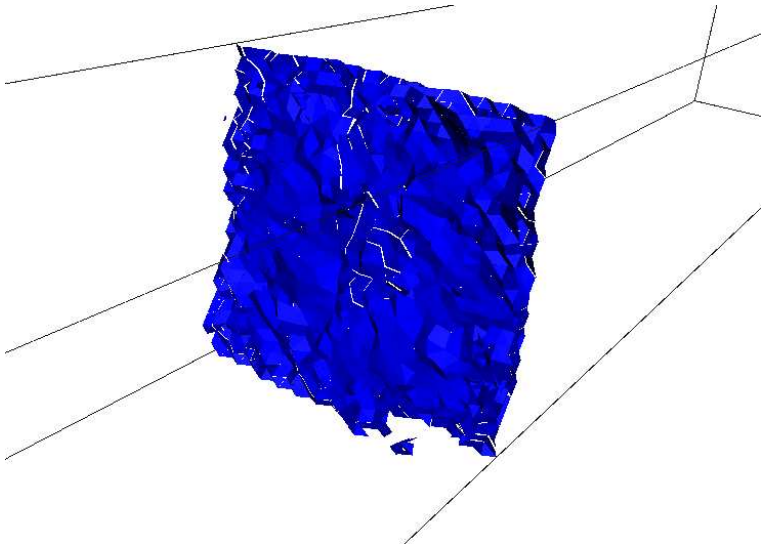
Fig. 8. External and internal views of the computed spall surface

We contrast the results obtained for this problem with the DG/CZM method with those in the CG intrinsic approach. In the latter, the Rose-Smith-Ferrante intrinsic cohesive law is utilized. For this law, the normal and tangential tractions are given respectively as,

$$T_N = \frac{\sigma_c}{\delta_0} \mathrm{e}^{(1-\delta/\delta_0)} \Delta_N \frac{ds}{dS} \tag{33}$$



(a) Computational mesh



(b) Fracture surface

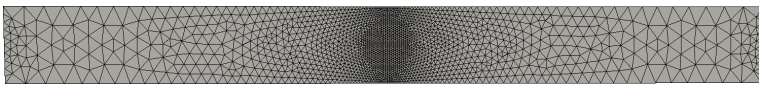
Fig. 9. Computed spall surface for a selectively-refined mesh showing a better resolution of the fracture plane

and

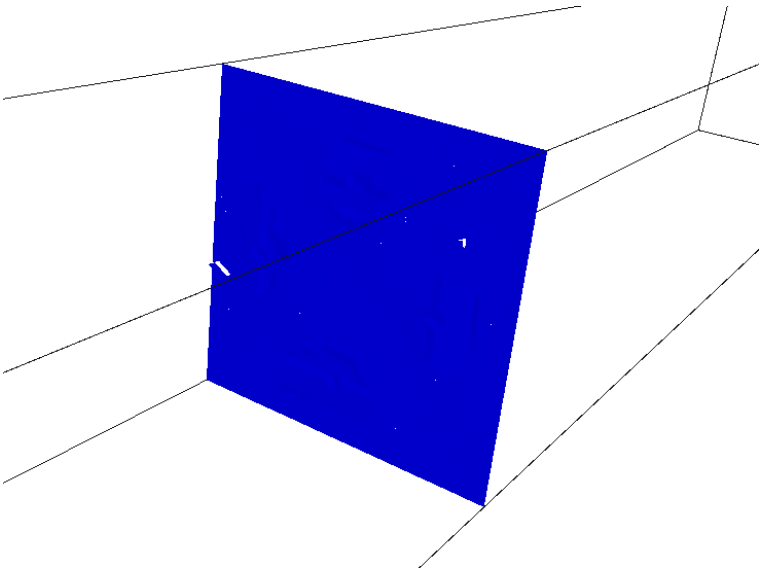
$$T_M = \gamma^2 \frac{\sigma_c}{\delta_0} \mathrm{e}^{(1-\delta/\delta_0)} \Delta_M \frac{ds}{dS} \tag{34}$$

where σ_c is the critical cohesive strength at the onset of fracture which occurs at $\delta = \delta_0$. For consistency with the DG/CZM approach, we choose the same values for the spall strength and the fracture energy. The cohesive law properties for the intrinsic TSL are given in Table 3. It bears emphasis that in this example the initial slope of the intrinsic law is determined by the physical parameters of the phenomenological TSL (i.e. the spall strength for σ_c, and the fracture energy for ϕ_s), and thus it cannot be used as a numerical knob to reduce the effect of artificial compliance.

Figure 11 shows the contours of longitudinal stress obtained with the CG/intrinsic



(a) Computational mesh



(b) Fracture surface

Fig. 10. Computed spall surface for a selectively-refined mesh with the plane $z = L/2$ contained among the mesh interelement boundaries which captures the fracture plane exactly

Properties	Values
Critical Cohesive Strength	$\sigma_c = 400$ MPa
Work of Separation	$\phi_s = 34$ J·m^{-2}
Critical Opening Displacement	$\delta_0 = 0.031$ μm
Weighting Parameter	$\gamma = 1.0$

Table 3

Cohesive law properties for the CG intrinsic formulation used in the simulation of uniaxial wave propagation and spall.

CZM method, corresponding to the same time snapshots as shown in Figure 6 for the DG approach. The first two snapshots in Figure 11(a)-(b) show no appreciable difference in this particular view. However, as Figure 7 attests, the speed and intensity of the stress wave is noticeably smaller than it should, as previously noted by (15). This is a manifestation of the problem of artificial compliance introduced by the intrinsic cohesive law, which affects wave propagation, as discussed above. The immediate consequence of this is that when the two incident tensile waves meet at the center, the spall stress is not reached, and the waves continue propagating from the center of the bar toward the ends with double intensity and no fracture, Figures 11(c)-(d) and Figure 7.

3.3 Scalability Tests

The scalability of the proposed DG/CZM method is assessed using simulations of the problem in Section 3.1 on three different platforms: our group cluster, and two supercomputers at the Department of Defense High Performance Computing Modernization Program (HPCMP). The platform details are:

- Our group cluster which consists of 40 compute nodes (320 compute cores). Each compute node contains two Intel 2.26 GHz Xeon E5520 64-bit quad-core processors with 24Gb of memory. The nodes are interconnected via a 4x DDR Infiniband network.
- The MJM system at the DoD Supercomputing Resource Center which consists of 1100 compute nodes (4400 cores). Each compute node contains two Intel 3.0 GHz Woodcrest 64-bit dual-core processors with 8 Gb of memory

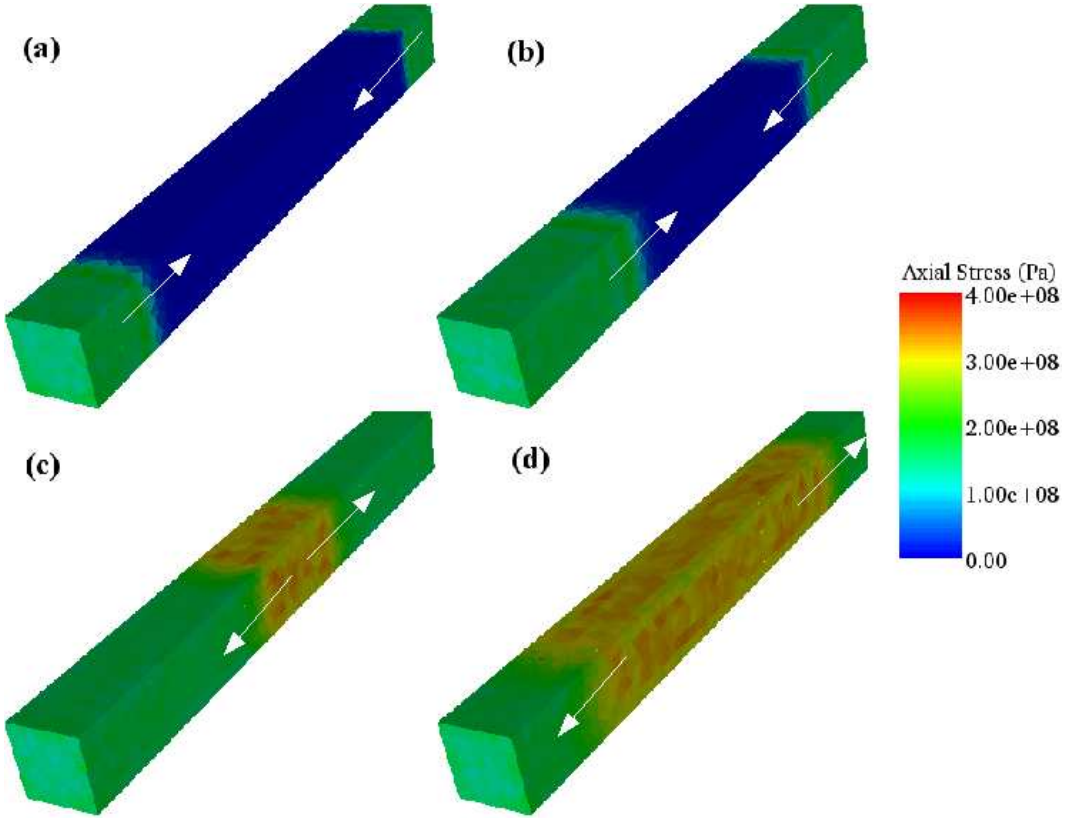


Fig. 11. Snapshots of wave propagation at (a) 0.08 μs, (b) 0.16 μs, (c) 0.32 μs, (d) 0.43 μs for the intrinsic CG/cohesive formulation showing a lack of spall due to incorrect wave propagation

each. The nodes are interconnected via a 4x DDR Infiniband network [1].

- The Diamond system which is and SGI Altix ICE consisting of 1,920 compute nodes (15,360 compute cores). Each compute node contains two 2.8GHz Intel Xeon 64bit quadcore Nehalem processors with 24 Gb of memory. The nodes are interconnected in a HyperCube topology DDR 4X InfiniBand network [2].

A finite element mesh comprising 50,732 volume elements (507,320 nodes) is employed iniitially on 2 processors. Then, both the number of processors and

[1] More information available at http://www.arl.hpc.mil/Systems/mjm.html
[2] More information available at http://www.erdc.hpc.mil/hardSoft/Hardware/ICE

the computational mesh are increased so as to maintain a fixed computational load per processor. The largest simulation was conducted on 4096 processors on the Diamond system and consisted of 103 million elements or 1.03 billion nodes.

The results are summarized in Figure 12, which shows a plot of the CPU time normalized with the number of elements and number of time steps as a function of the number of cores used for each platform considered. The plot shows that the DG/CZM method maintains excellent scalability at least up to 4096 processors.

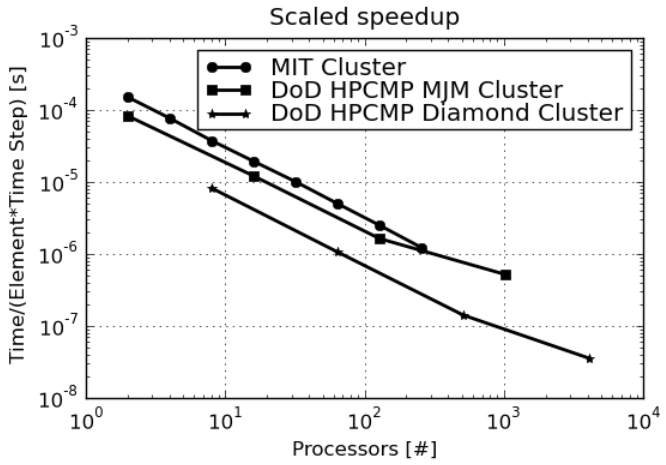


Fig. 12. Scalability of the DG/CZM method on up to 4096 processors and problems of upward of 3 billion degrees of freedom: Scaled speed-up results given as compute time per time step and per element as a function of number of processors for all three platforms tested

These results are consistent with the scalability of the continuum DG framework for explicit dynamic calculations of large deformation of solids presented in (35). However, the new results not only include the extended framework for dynamic fracture but are also for very large problems (up to 3 billion degrees of freedom) and processor counts (up to 4096). Thus, the scalability of the

proposed DG/CZM method for large parallel computations of fracture and fragmentation is demonstrated.

3.4 Application: Ceramic Plate Impact

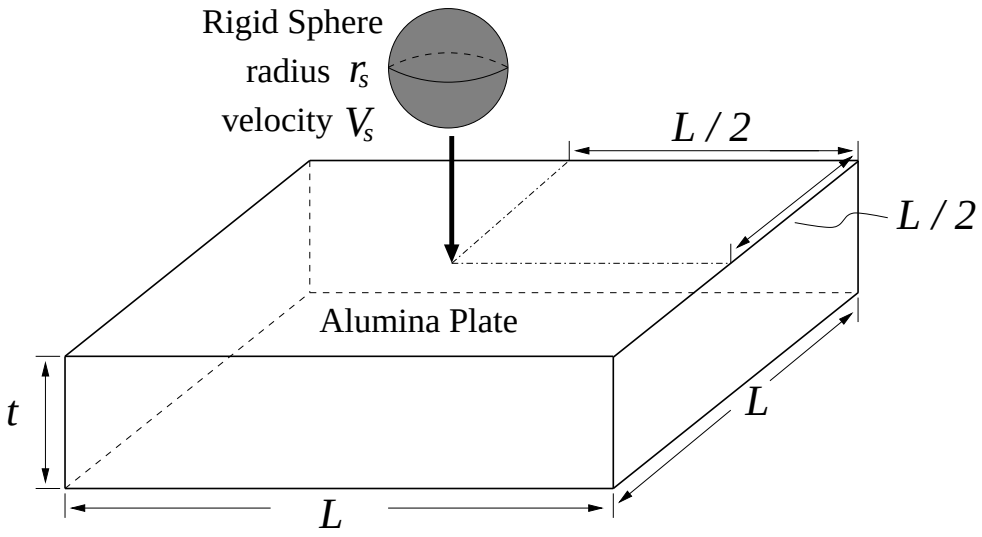


Fig. 13. A schematic of the simulation setup for the impact simulation.

As a key example demonstrating the capabilities of the proposed DG/CZM method in a problem of practical interest, we consider the high-velocity impact of ceramic plates with hard spherical projectiles. Ceramics are commonly employed in the design of protective armor plates in defense applications due to their high ballistic impact resistance. The pioneering paper of Shockey et al (46) showed that impact response of ceramic plates exhibits a complex combination of failure modes including radial, conical and lateral cracks, as well as lattice plasticity. The computational modeling of this complex problem using CZM was first attempted by Camacho and Ortiz (18). They showed that the extrinsic CZM was successful at capturing the conical crack patterns in ceramic plate impact providing finely resolved meshes were employed in the calculations. However, due to the axisymmetric assumption which enabled the

calculations to be conducted in a two-dimensional domain, the radial cracks could not be modeled explicitly via the CZM. Instead, they adopted a continuum damage approach only for this purpose. Here, we revisit this problem and attempt a full three-dimensional description of the ceramic plate impact problem using the DG/CZM proposed. We emphasize that three dimensional calculations with the resolution required to capture all the crack patterns in this problem is enabled by the scalability of the approach, which, in turn, relies on the possibility of seamlessly propagating cracks across processor boundaries.

We consider the model problem of impact of rigid sphere at the center of a square alumina plate depicted schematically in Figure 13. The plate and sphere dimensions are given in table 4. It is assumed that the plate behaves elastically until the onset of fracture, with the elastic response described by equation (32). The material properties for the bulk elements of the plate are listed in table 4. Prior to fracture, the interface elements at the interelement boundaries respond according to the continuum DG formulation as detailed in section 2. Upon the onset of fracture, the cohesive law in section 2.2 with the properties in table 5 becomes operative at the quadrature points of the interface elements following equation (14). The finite element mesh employed in the calculation consists of 183,673 volumetric finite elements or 1,836,730 nodes. In Figure 14 we show the initial mesh with 16 partitions represented with different intensities of gray.

The results of this simulation are shown in the sequence of Figures 15-20 and in Figure 21. Each Figure of the sequence 15-20 shows snapshots of the simulation at different times consisting of a montage of: (a) hydrostatic stress contours on the impacted surface of the plate, (b) instantaneous maximum principal stress contours on the back face of the plate, (c) contours of the

Properties	Values
Plate	
Length	$L = 5.08$ cm
Thickness	$t = 1.2$ cm
Initial density	$\rho_0 = 3690$ kg·m^{-3}
Elastic Modulus	$E = 260$ GPa
Poisson's Ratio	$\nu = 0.21$
Sphere	
Radius	$r_s = 7.66$ mm
Density	$\rho_s = 8000$ kg·m^{-3}
Velocity	$V_s = (0;\ 0;\ -300$ m·s$^{-1})$

Table 4

Plate and sphere dimensions and material properties used for the ceramic impact simulation.

instantaneous maximum principal stress in a through-thickness cross section, (d) the trace of the fracture surfaces on a through-thickness cross section and (e) a three-dimensional rendering of the fracture surfaces. Figure 15 corresponds to time $t = 0.52\mu$s soon after the time of impact. It shows the incipient stress waves, including a strong state of hydrostatic compression directly underneath the impact point, a Rayleigh wave on the surface and a compression wave propagating through the cross section of the plate hemispherically. Figure 16 corresponds to time $t = 2.09\mu$s, soon after the stress wave has reached

Properties	Values
DG Stability Parameter	$\beta_s = 4.0$
Critical Cohesive Strength	$\sigma_c = 1.4$ GPa
Work of Separation	$\phi_{\mathrm{s}} = 25.4$ J·m^{-2}
Critical Opening Displacement	$\delta_c = 0.17$ μm
Weighting Parameter	$\gamma = 1.0$

Table 5

Cohesive law parameters used in the ceramic impact simulation

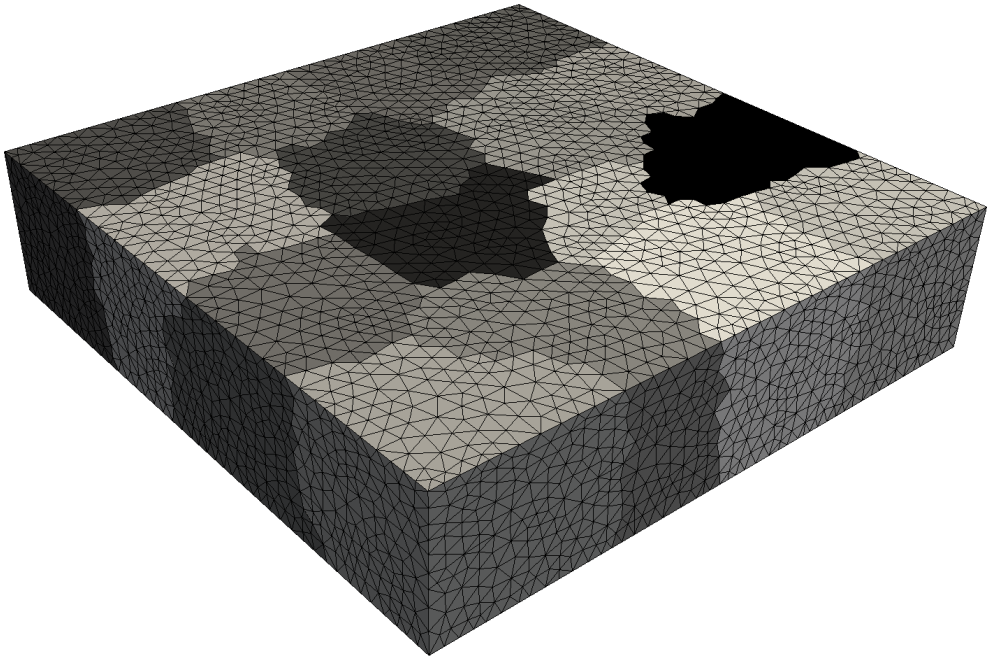
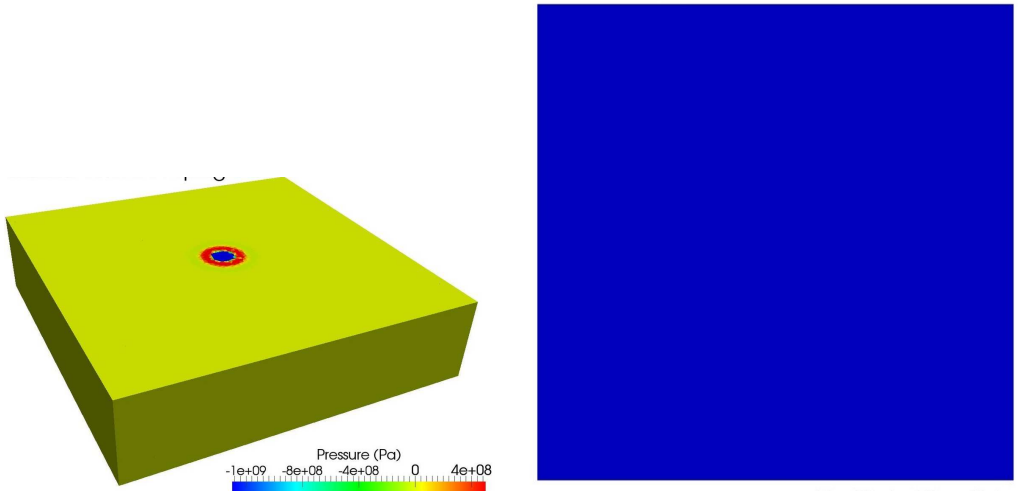


Fig. 14. Undeformed mesh for the ceramic plate impact problem consisting of 183,673 volumetric elements showing the processor boundaries for 16 processors

the back face. The maximum principal stress in the reflected wave is large enough to initiate the first cracks on the back surface, which is accompanied by a stress release on the newly-created free crack surfaces and behind the

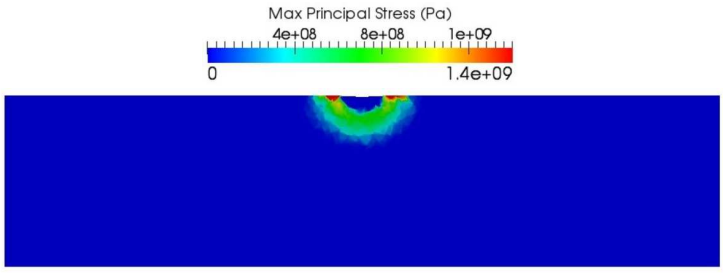
wave front propagating radially outwards. As Figure 16(e) shows, there is an ensuing fracture ring on the front face originated at the interface between the precursor surface wave propagating outwardly which and the hydrostatic-compression region underneath the impactor. At time $t = 3.4\mu$s, Figure 17, the development and propagation of radial cracks at the back face (b) and of conical cracks in the cross section (c),(d) can be clearly observed. The radial cracks are clearly driven by the high in-plane biaxial tensile stress wave propagating outwardly. The stress level drops significantly behind the radial crack tips. Similarly, the conical crack propagation is driven by the high principal stress localizing on the conical surface (c). At time $t = 5.46\mu$s, Figure 18, it becomes clear that the crack patterns are dominated by several radial cracks at the back face still propagating toward the edges and a principal conical crack stemming from underneath the impact zone. It can also be observed that the radial crack which initially propagates towards the bottom left corner in Figure 18(b) branches out into two cracks which propagate away from the corner and toward the adjacent edges. Later on in the simulation, Figure 19, the fracture pattern continues to develop and both the radial and conical cracks become clearly visible as crack openings grow to a macroscopic scale. Figure 20 shows the final computed snapshot, after which no further fragmentation is experienced by the plate. The trace of the conical and radial cracks on the face at the end of the simulation is shown in Figure 21.

From this simulation, it can be seen that the fracture and fragmentation process in ceramic plate impact is clearly determined by the propagation of stress waves in the material. Conversely, the propagation of stress waves is substantially affected by the ensuing cracks, as the sequence of pressure contours on the surface of the plate in Figures 15-20(a) clearly shows. More specifically,



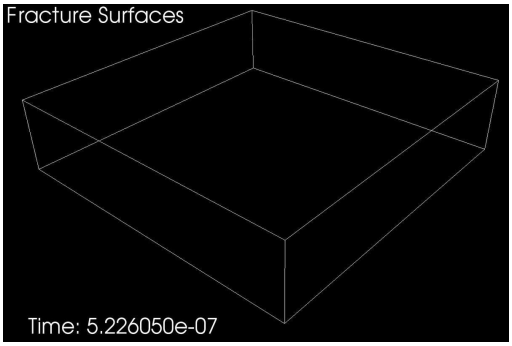
(a) Hydrostatic stress contours on external surface of the plate

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



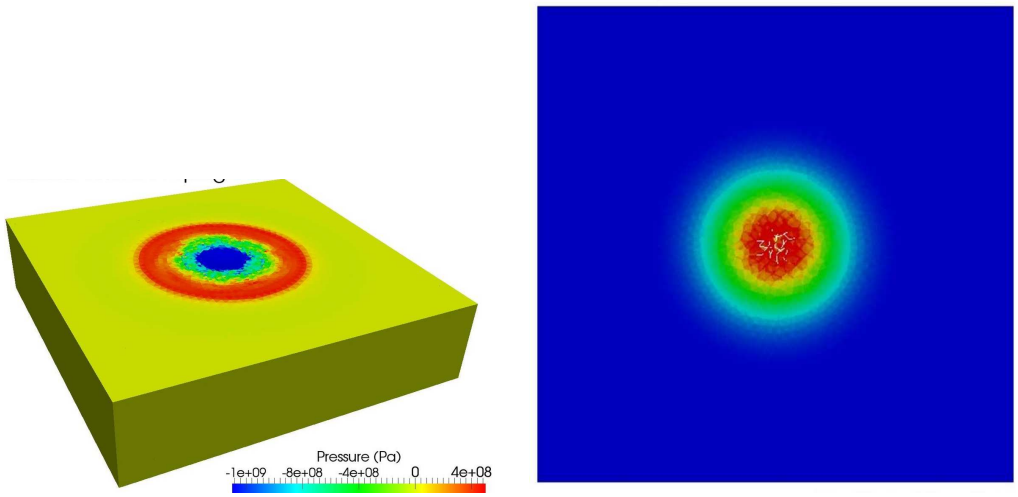
(c) Contours of maximum principal stress in cross section

(d) Trace of fracture surfaces on cross section



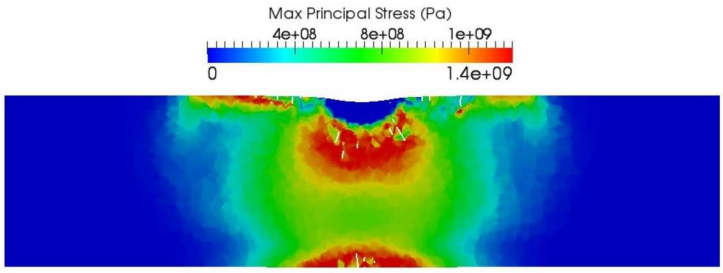
(e) 3D rendering of fracture surfaces

Fig. 15. Snapshot of simulation results at $t = 0.52 \times 10^{-6}$s

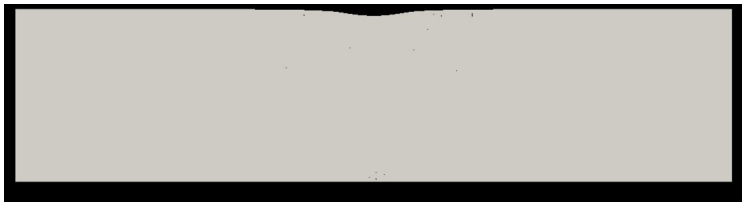


(a) Hydrostatic stress contours on external surface of the plate

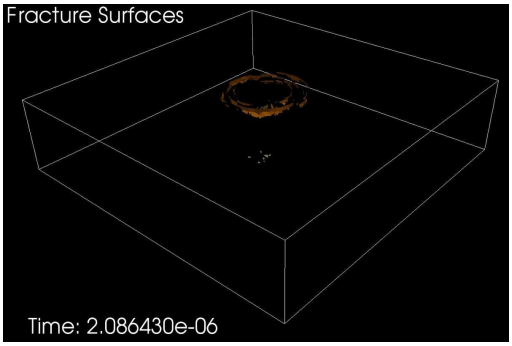
(b) Maximum principal stress contours on back face of the plate



(c) Contours of maximum principal stress in cross section

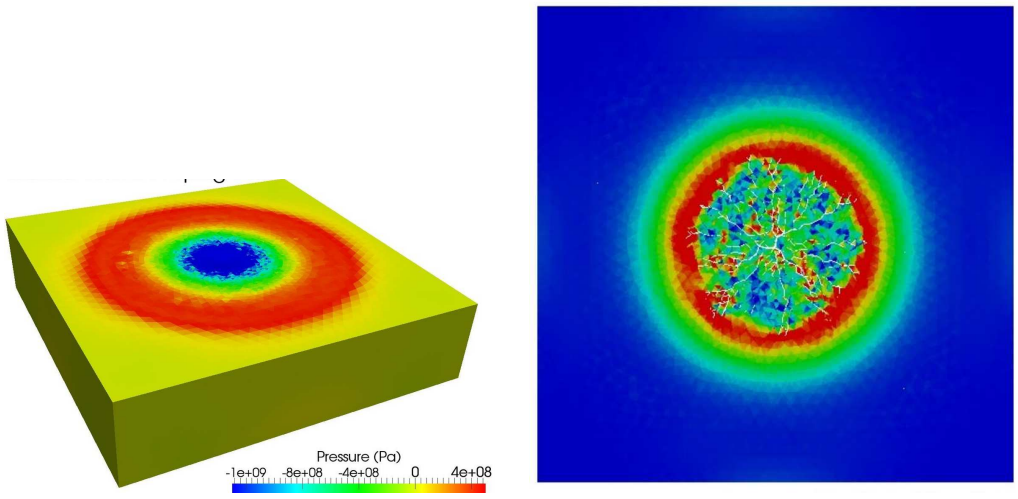


(d) Trace of fracture surfaces on cross section



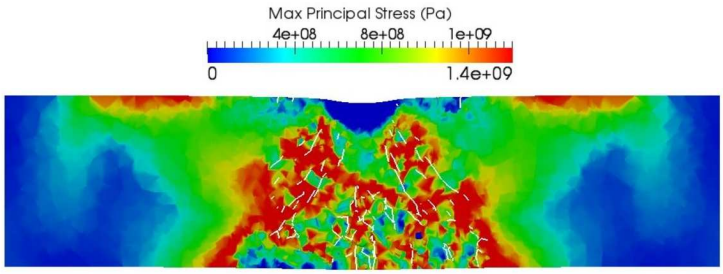
(e) 3D rendering of fracture surfaces

Fig. 16. Snapshot of simulation results at $t = 2.09 \times 10^{-6}$s

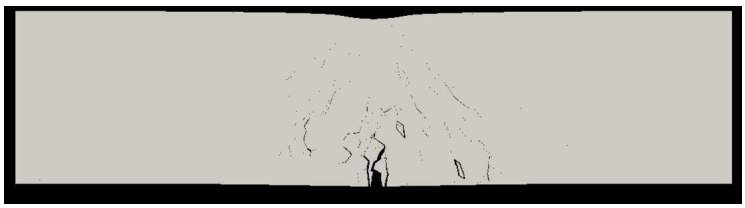


(a) Hydrostatic stress contours on external surface of the plate

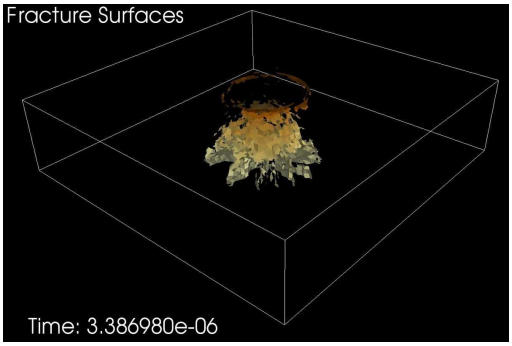
(b) Maximum principal stress contours on back face of the plate



(c) Contours of maximum principal stress in cross section

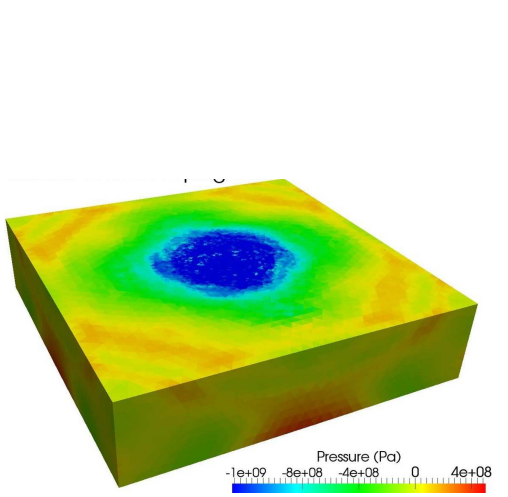


(d) Trace of fracture surfaces on cross section

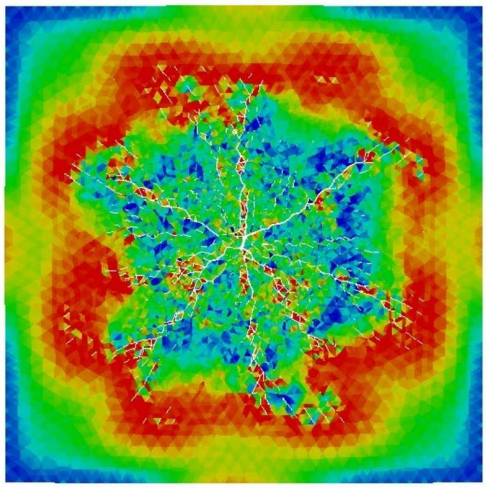


(e) 3D rendering of fracture surfaces

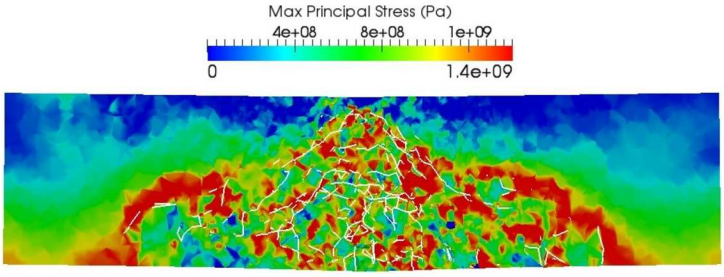
Fig. 17. Snapshot of simulation results at $t = 3.4 \times 10^{-6}$s



(a) Hydrostatic stress contours on external surface of the plate

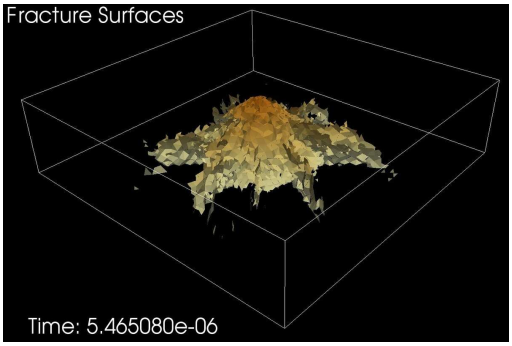


(b) Maximum principal stress contours on back face of the plate



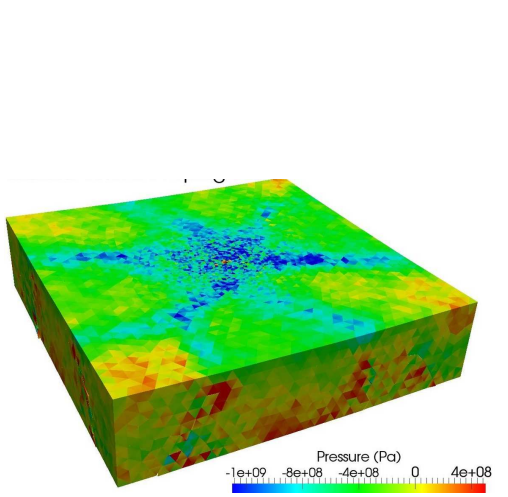
(c) Contours of maximum principal stress in cross section

(d) Trace of fracture surfaces on cross section

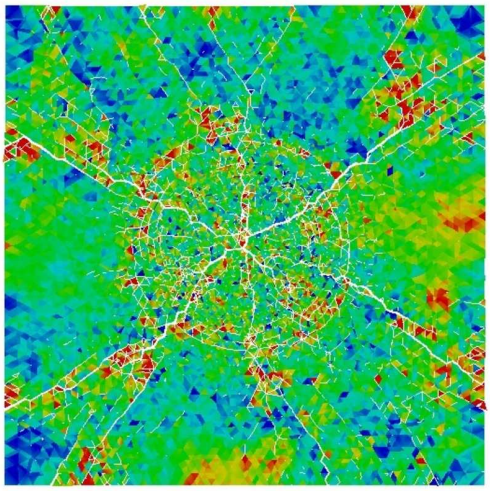


(e) 3D rendering of fracture surfaces

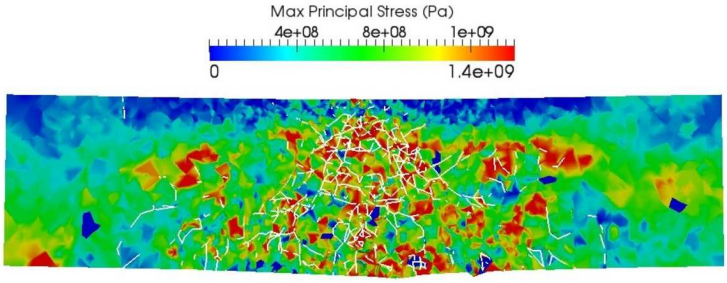
Fig. 18. Snapshot of simulation results at $t = 5.46 \times 10^{-6}$s



(a) Hydrostatic stress contours on external surface of the plate

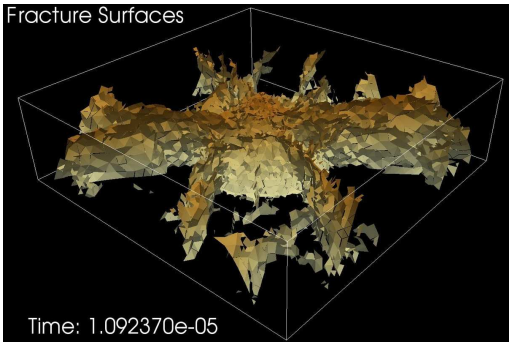


(b) Maximum principal stress contours on back face of the plate



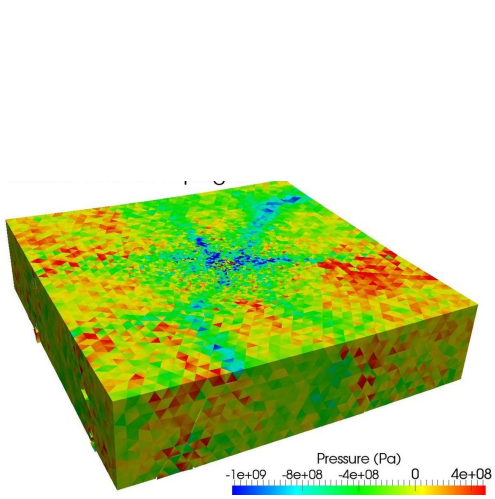
(c) Contours of maximum principal stress in cross section

(d) Trace of fracture surfaces on cross section

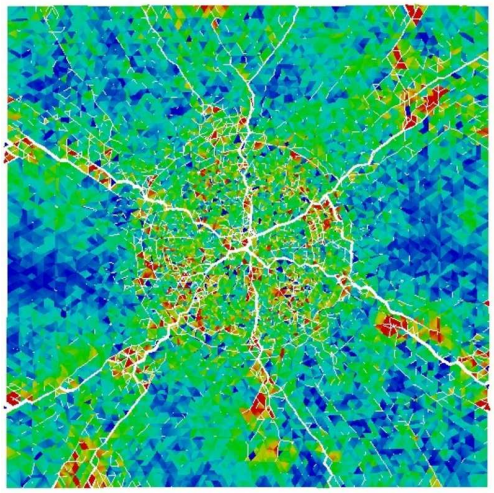


(e) 3D rendering of fracture surfaces

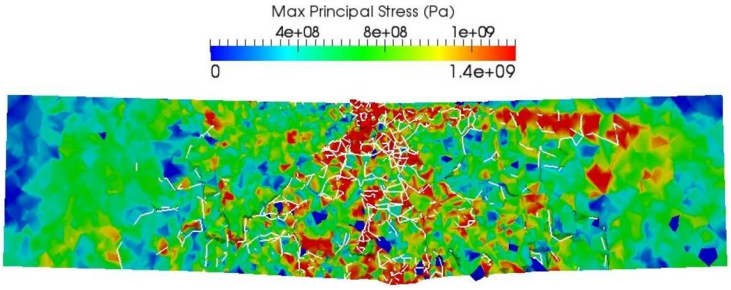
Fig. 19. Snapshot of simulation results at $t = 10.9 \times 10^{-6}$s



(a) Hydrostatic stress contours on external surface of the plate

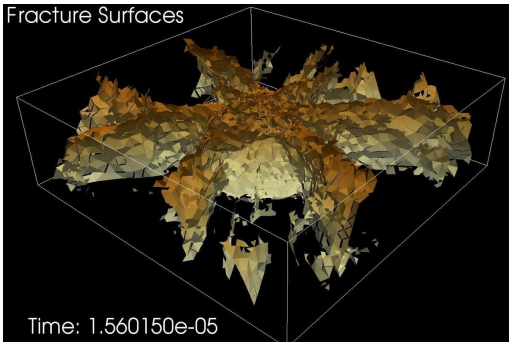


(b) Maximum principal stress contours on back face of the plate



(c) Contours of maximum principal stress in cross section

(d) Trace of fracture surfaces on cross section



(e) 3D rendering of fracture surfaces

Fig. 20. Snapshot of simulation results at $t = 15.6 \times 10^{-6}$s

stress concentrations at crack tips, stress release and posterior multiple reflections at newly-created crack surfaces lead to very heterogeneous stress distributions. It is thus clear the importance of properly describing the propagation of stress waves in the fracturing material. In particular, we highlight the potential importance of post-fracture transmission of compressive waves across closed or closing cracks, a challenge for the CZM approach which is clearly affected by the artificial compliance issue discussed throughout this paper. As explained in Section 2.2, this is properly handled in the proposed method by falling back to the normal component of the DG fluxes when cracks close and the interfaces go into compression, thus guaranteeing the transmission of longitudinal compressive waves as in the continuum uncracked problem.

Fig. 21. Radial cracks and trace of dominant conical crack on plate back face at the end of the simulation

We finally show in Figure 22 a view of the back face in the deformed con-

figuration at the end of the simulation with the different mesh partitions in different colors. The purpose of this figure is to demonstrate that in the parallel DG/CZM method proposed, cracks are able to propagate freely across the processor boundaries without the need of communicating any topological information between the processors.

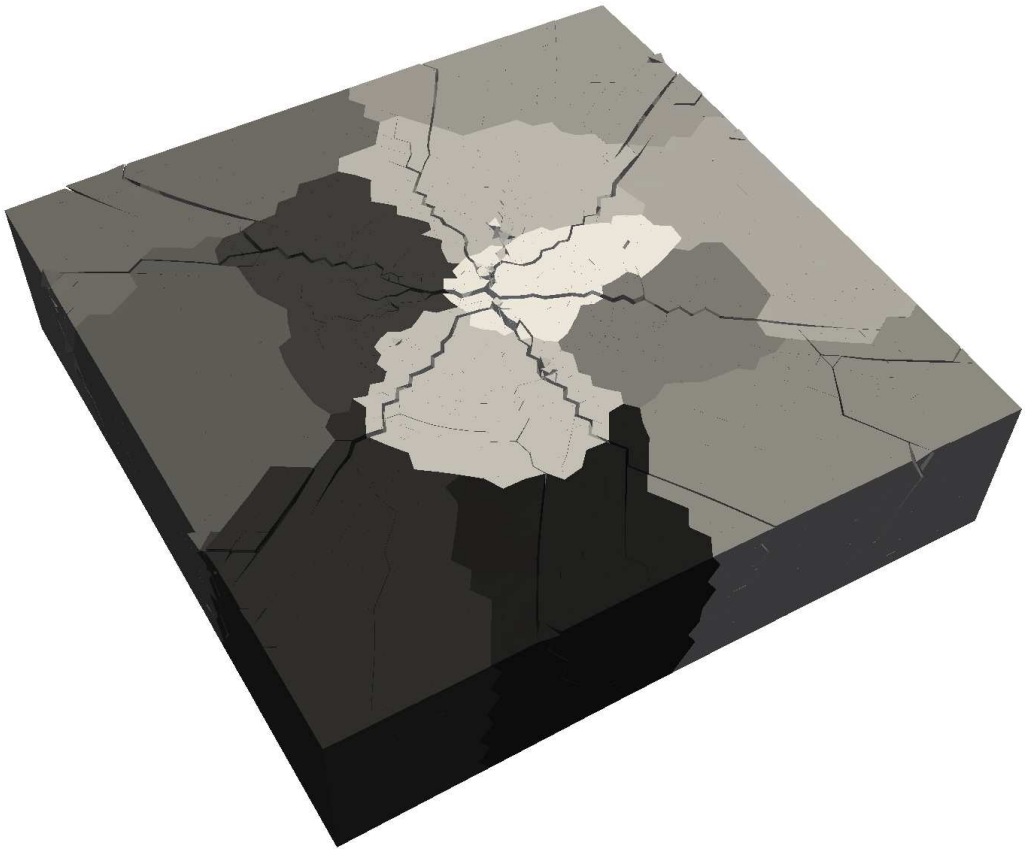


Fig. 22. Deformed back face view at the end of the simulation showing radial cracks propagating across mesh partitions

4 Conclusions

In this paper, a scalable 3D computational method was presented for modeling dynamic fracture and fragmentation of brittle solids. The method is based

on a combination of a discontinuous Galerkin formulation of the continuum problem and Cohesive Zone Models of fracture. The approach is general in the sense that it does not place any restriction on the TSL employed in the description of fracture.

By contrast to intrinsic cohesive zone models, the DG framework guarantees the consistency and stability of the numerical solution prior to fracture. This, in turn, avoids wave dispersion issues associated with "artificial compliance" of intrinsic cohesive element methods. This was demonstrated in the context of uniaxial wave propagation in a brittle elastic bar, where the proposed method captures both the spall and stress relief, whereas the intrinsic cohesive approach fails. Another important advantage of the DG/CZM formulation is that the cohesive law operates strictly at the quadrature point where the fracture criterion is met, whereas other quadrature points may remain unfractured, thus affording the possibility of sub-element crack resolution.

Perhaps the most appealing aspect of the DG/CZM formulation presented is its straightforward parallel implementation which does not require the communication of topological mesh changes as cracks propagate across processor boundaries. This enables large-scale simulations with highly refined meshes which is critical for describing complex crack patterns and fragmentation in three-dimensions. The excellent scalability properties of the parallel implementation were demonstrated on three different platforms on up to 4096 processors and problem sizes of up to three billion degrees of freedom.

The method was applied to the problem of impact of a ceramic plate and captured the main patterns of conical and radial cracks which propagate in the mesh impassive to the presence of processor boundaries.

5 Acknowledgments

The authors acknowledge the support of the Office of Naval Research under grant N00014-07-1-0764. Partial support from the U.S. Army through the Institute for Soldier Nanotechnologies, under Contract DAAD-19-02-D-0002 with the U.S. Army Research Office is also gratefully acknowledged.

References

[1] F. P. Bowden, J. E. Field. The brittle fracture of solids by liquid impact, by solid impact and by shock. Proceedings of the Royal Society of London, A 1964;282:331–352. 1

[2] A. G. Evans, T. R. Wilshaw. Dynamic solid particle damage in brittle materials, as appraisal. Journal of Material Science 1977;12:97–116. 1

[3] A. G. Evans, M. E. Gulden, M. Rosenblatt. Impact damage in brittle materials in the elastic-plastic response regime. Proceedings of the Royal Society of London, A 1978;361:343–365. 1

[4] L. Prokurat Franks, editor. Proceedings of the 32nd International Conference and Exposition on Advanced Ceramics & Composites. The American Ceramics Society. Wiley, Daytone Beach, FL, 2008. 1

[5] N. R. Council. Protecting the Space Station from Meteoroids and Orbital Debris. National Academy Press, Washington, DC, 1997. 1

[6] G. Barenblatt. The mathematical theory of equilibrium cracks in brittle fracture. Advances in Applied Mechanics 1962;7:55–129. 1

[7] D. Dugdale. Yielding of steel sheets containing clits. Journal of the Mechanics and Physics of Solids 1960;8:100–104. 1

[8] A. Seagraves, R. Radovitzky. Dynamic Failure of Materials and Struc-

tures. Springer, 2009. Ch. 12 Advances in Cohesive Zone Modeling of Dynamic Fracture. 1, 2

[9] X. Xu, A. Needleman. Numerical simulations of dynamic crack growth along an interface. International Journal of Fracture 1996;74:289–324. 1

[10] X. Xu, A. Needleman, F. Abraham. Effect of inhomogeneities on dynamic crack growth in an elastic solid. Modelling and Simulation in Materials Science and Engineering 1997;5:489–516. 1, 1

[11] G. Ruiz, A. Pandolfi, M. Ortiz. Three-dimensional cohesive modeling of dynamic mixed-mode fracture. International Journal for Numerical Methods in Engineering 2001;52(1-2):97–120. 1

[12] Z. Zhang, G. Paulino, W. Celes. Extrinsic cohesive modeling of dynamic fracture and microbranching instability in brittle materials. International Journal for Numerical Methods in Engineering 2007;72:893–923. 1

[13] M. Ortiz, S. Suresh. Statistical properties residual stresses and intergranular fracture in ceramic materials. Journal of Applied Mechanics 1993;60:77–84. 1

[14] X. Xu, A. Needleman. Numerical simulation of fast crack growth in brittle solids. Journal of the Mechanics and Physics of Solids 1994;42(9):1397–1434. 1, 1

[15] H. Espinosa, P. Zavattieri. A grain level model for the study of failure initiation and evolution in polycrystalline brittle materials. part i: Theory and numerical implementation. Mechanics of Materials 2003;35:333–364. 1, 1, 3.2

[16] P. Klein, J. Foulk, E. Chen, S. Wimmer, H. Gao. Physics-based modeling of brittle fracture, cohesive formulations and the applications of meshfree methods. Theoretical and Applied Fracture Mechanics 2001;37:99–166. 1, 2.1

[17] J. Foulk III. An examination of stability in cohesive zone modeling. Computer Methods in Applied Mechanics and Engineering 2010;199:465–470. 1

[18] G. Camacho, M. Ortiz. Computational modeling of impact damage in brittle materials. International Journal of Solids and Structures 1996;33(20–22):2899–2983. 1, 3.4

[19] M. Ortiz, A. Pandolfi. Finite-deformation irreversible cohesive elements for three-dimensional crack-propagation analysis. International Journal for Numerical Methods in Engineering 1999;44:1267–1282. 1, 2.2

[20] O. Nguyen, E. Repetto, M. Ortiz, R. Radovitzky. A cohesive model of fatigue crack growth. International Journal of Fracture 2001;110:351–369. 1

[21] A. Pandolfi, M. Ortiz. Solid modeling aspects of three-dimensional fragmentation. Engineering with Computers 1998;14:287–308. 1

[22] A. Pandolfi, M. Ortiz. An efficient adaptive procedure for three-dimensional fragmentation simulations. Engineering with Computers 2002;18:148–159. 1

[23] A. Mota, J. Knap, M. Ortiz. Fracture and fragmentation of simplicial finite element meshes using graphs. International Journal for Numerical Methods in Engineering 2008;73:1547–1570. 1

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

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

[26] M. Ortiz, Y. Leroy, A. Needleman. A finite element method for localized

failure analysis. Computer Methods in Applied Mechanics and Engineering 1987;61:189–214. 1

[27] E. Dvorkin, A. Cuitino, G. Gioia. Finite elements with displacement interpolated embedded localization lines insensitive to mesh size and distortions. International Journal for Numerical Methods in Engineering 1990;30:541–564. 1

[28] F. Armero, C. Linder. Numerical simulation of dynamic fracture using finite elements with embedded discontinuities. International Journal of Fracture 2009;160:119–141. 1

[29] N. Moes, J. Dolbow, T. Belytschko. A finite element method for crack growth without rememshing. International Journal for Numerical Methods in Engineering 1999;46:131–150. 1

[30] J. Dolbow, N. Moes, T. Belytschko. An extended finite element method for modeling crack growth with frictional contact. Computer Methods in Applied Mechanics and Engineering 2001;190:6825–6846. 1

[31] P. Areias, T. Belytschko. Analysis of three-dimensional crack intitiation and propagation using the extended finite element method. International Journal for Numerical Methods in Engineering 2005;63:760–788. 1

[32] J. Remmers, R. de Borst, A. Needleman. A cohesive segments method for the simulation of crack growth. Computational Mechanics 2003;31:69–77. 1

[33] R. de Borst, J. Remmers, A. Needleman. Mesh-independent discrete numerical representations of cohesive-zone models. Engineering Fracture Mechanics 2006;73:160–177. 1

[34] L. Noels, R. Radovitzky. A general discontinuous Galerkin method for finite hyperelasticity. Formulation and numerical applications. International Journal for Numerical Methods in Engineering 2006;68(1):64–97.

1, 2, 2.1, 2.1, 2.1

[35] L. Noels, R. Radovitzky. An explicit discontinuous Galerkin method for non-linear solid dynamics. Formulation, parallel implementation and scalability properties.. International Journal for Numerical Methods in Engineering 2007;74:1393–1420. 1, 2, 2.1, 2.1, 2.1, 2.3, 2.4, 3.3

[36] J. Mergheim, E. Kuhl, P. Steinmann. A hybrid discontinuous Galerkin/interface method for the computational modelling of failure. Communication in Numerical Methods in Engineering 2004;20:511–519. 1

[37] R. Abedi, M. Hawker, R. Haber, K. Matous. An adaptive spacetime discontinuous galerkin method for cohesive models of elastodynamic fracture. International Journal for Numerical Methods in Engineering 2009;81:1207–1241. 1

[38] F. Bassi, S. Rebay. A high-order accurate discontinuous finite element method for the numerical solution of the compressible navier-stokes equations. Journal of Computational Physics 1997;131:267–279. 2.1

[39] D. Arnold, F. Brezzi, B. Cockburn, L. Marini. Unified analysis of discontinuous galerkin methods for elliptic problems. SIAM Journal for Numerical Analysis 2002;39(5):1749–1779. 2.1

[40] F. Brezzi, M. Manzini, D. Marini, P. Pietra, A. Russo. Discontinuous galerkin approximations for elliptic problems. Numerical Methods for Partial Differential Equations 2000;16:47–58. 2.1

[41] T. Belytschko. Computational Methods for Transient Analysis. Elsevier Science, North-Holland, 1983. 2.3

[42] G. Karypis, V. Kumar. Analysis of multilevel graph partitioning. in: A. for Computing Machinery, editor, Supercomputing. San Diego, 1995. 1

[43] R. Radovitzky, M. Ortiz. Tetrahedral mesh generation based on node insertion in crystal lattice arrangements and advancing-front-delaunay triangulation. Computer Methods in Applied Mechanics and Engineering 2000;187:543–569. 4

[44] K. Danielson, N. R.R.. Nonlinear dynamic finite element analysis on parallel computers using FORTRAN 90 and MPI. Advances in Engineering Software 1998;29:179–186. 2.4

[45] J. Cummings, M. Aivazis, R. Samtaney, R. Radovitzky, S. Mauch, D. Meiron. A virtual test facility for the simulation of dynamic response in materials. Journal of Supercomputing 2002;23:39–50. 2.4

[46] D. A. Shockey, A. Marchand, S. Skaggs, G. Cort, M. Burkett, R. Parker. Failure phenomenology of confined ceramic targets and impacting rods. International Journal of Impact Engineering 1990;9(3):263–275. 3.4