

Peridynamics-Based Fracture Animation for Elastoplastic Solids

W. Chen, F. Zhu, J. Zhao, S. Li, G. Wang

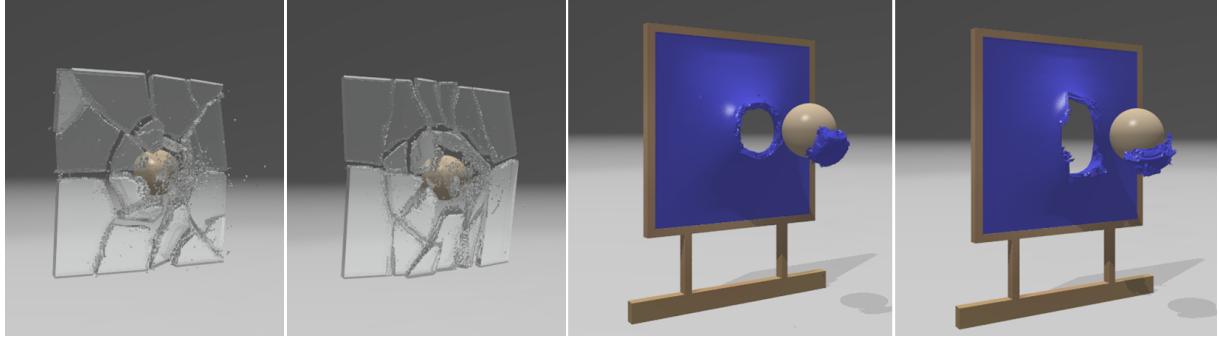


Figure 1: A sphere shoots through walls made of different materials, causing varied fracture behaviors. From left to right: isotropic brittle fracture, anisotropic brittle fracture, isotropic ductile fracture, anisotropic ductile fracture.

Abstract

In this paper, we exploit the use of peridynamics theory for graphical animation of material deformation and fracture. We present a new meshless framework for elastoplastic constitutive modeling that contrasts with previous approaches in graphics. Our peridynamics-based elastoplasticity model represents deformation behaviors of materials with high realism. We validate the model by varying the material properties and performing comparisons with FEM simulations. The integral-based nature of peridynamics makes it trivial to model material discontinuities, which outweighs differential-based methods in both accuracy and ease of implementation. We propose a simple strategy to model fracture in the setting of peridynamics discretization. We demonstrate that the fracture criterion combined with our elastoplasticity model could realistically produce ductile fracture as well as brittle fracture. Our work is the first application of peridynamics in graphics that could create a wide range of material phenomena including elasticity, plasticity, and fracture. The complete framework provides an attractive alternative to existing methods for producing modern visual effects.

Keywords: peridynamics, fracture, elastoplasticity

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation

1. Introduction

The simulation of deformable materials has been an important research topic in computer graphics for decades, since the early work by Terzopoulos and colleagues [TPBF87]. One of the strongest driving forces behind the active research is the persistently growing need for higher realism from the visual effects industry. Materials in the real world exhibit

complex behaviors, such as coupled elastoplastic deformations, fracture, etc. The complicated material behaviors are difficult to replicate by any single method despite the numerous ones that have been developed thus far. Existing approaches generally excel at some phenomena but would stumble (if not fail) at others. For instance, mesh-based methods [MG04, ITF04, TSIF05, SB12] are a good choice

to simulate elastic deformations whereas not preferred for phenomena that involve topological changes. Particle-based methods [MCG03, PKA^{*}05, SSC^{*}13] are considered suitable for modeling topological changes, however the inherent loss of connectivity information would cause undesirable numerical fracture [LZW11, ZZL^{*}16] while simulating large deformations.

We build on recent developments of peridynamics theory in the computational physics community [Sil00, SEW^{*}07, Mit11, ELP13, MO14] and propose a novel framework for graphical animation of varied deformation behaviors and fracture. Our aim is to enrich available options of simulation techniques for easier and better animation production. Peridynamics was first adopted to animation applications by Levine et al. [LBC^{*}14] where they described a simple spring-mass system to handle brittle fracture of solids. In contrast, we handle elastoplasticity, brittle fracture, and ductile fracture in a single framework. To this end, we propose several novel contributions in this work. We first present an elastoplastic constitutive model in the peridynamics-based framework with simple extension to anisotropy, and the model is validated against results produced by FEM. Furthermore, we show that both brittle and ductile fracture phenomena can be naturally represented with nearly no effort by integrating a simple fracture criterion into this material model. This is due to the integral-based formulation of peridynamics, in which forces at a material point are computed by gathering contributions from all material points in its interaction range through integration. On the other hand, methods based on classical continuum mechanics formulate force computations with partial differential equations that fail to be applicable directly on singularities such as a crack. This feature makes our peridynamics-based framework more attractive over existing approaches for producing animations that involve fracture. Lastly, our method is simple to implement and trivially parallelizable, providing a useful alternative to previous methods for animation production.

2. Related Work

A large body of literature has been devoted to physical simulation of natural phenomena as a result of active research. A complete literature review is beyond the scope of this paper. In the following we comment only on the representative works most related to ours.

Elastoplasticity Animation The modeling of deformable plasticity in graphics dates back to the pioneering work by Terzopoulos and Fleischer [TF88]. O'Brien and colleagues [OBH02] incorporated a similar additive plasticity model into a finite element simulation to animate ductile fracture. The strain measure was decomposed into two components, where one is due to elastic deformation and the other due to plastic deformation. Müller et al. [MKN^{*}04] applied this model in their point-based animation framework and simulated plastic behaviors of objects. Irving et al. [ITF04] presented a

multiplicative formulation of plasticity and pointed out that their model was better handling finite plastic deformation than the additive one. In contrast to the additive model, they decomposed the deformation gradient into two components through multiplication. The multiplicative model was extensively used by later works to animate phenomena that involve plasticity. Bargteil et al. simulated large viscoplastic flow [BWHT07], Gerszewski and his colleagues animated elastoplastic solids [GBB09], and Stomakhin et al. modeled plasticity of snow [SSC^{*}13], just to name a few. Unfortunately, neither of the above plasticity models applies in the peridynamics framework because there is no concept of strain nor deformation gradient in the integral-based formulation. As a result, we present a new constitutive model for peridynamics in this work to animate elastoplastic solids.

Fracture Animation Numerous methods have been proposed on fracture animation [MBP14, WWD15] because the stunning phenomenon of fracture and failure is an indispensable visual element in animated movies and video games. Early approaches use simple schemes to model fracture, such as the finite difference method [TF88], the mass-spring system [NTB^{*}91], and the mass-point constraint system [SWB01]. O'Brien and colleagues [OH99] adopted techniques from continuum mechanics and presented a FEM-based method to simulate brittle fracture of solids. They later extended their method to ductile fracture by incorporating a plasticity model [OBH02]. Müller et al. [MMDJ01] employed a quasi-static finite element analysis to animate brittle fracture of stiff materials undergoing collisions. Parker et al. [PO09] presented some useful techniques for real-time simulation of fracture in game environment. One major issue in FEM-based methods is the generation of fracture patterns on meshes, which could alter the underlying mesh topology. Early methods typically made use of simple separation along mesh element boundaries [NTB^{*}91, MMA99, SWB01, MMDJ01] or even element deletion [FDA02]. Mesh subdivision prior to splitting could somewhat increase the available geometric details [MK00, BG00], whereas this tended to introduce elements with poor aspect ratios. Allowing failure along more arbitrary paths could generate more geometrically rich fracture patterns [NF99, OH99, OBH02], albeit at the expense of complicated re-meshing. Molino et al. [MBF04] proposed a virtual node algorithm to avoid the complexity of re-meshing, where elements were duplicated into partially filled counterparts with virtual nodes. The virtual node algorithm was frequently used by subsequent works on fracture animation [BHTF07] and mesh cutting [SDF07, WJST14] due to its simplicity compared to re-meshing methods. Other representative mesh-based methods resorted to modal analysis [GMD13] and pure geometric mesh decompositions [MCK13, SO14] for real-time brittle fracture. Most recently, several works explored the boundary element method for rigid body fracture [ZBG15, HW15] where only

surface meshes were employed for both representation and computation.

In contrast to mesh-based approaches, meshless methods are generally considered as a better solution for animating topological changes. Based on the moving least square (MLS) meshless framework by Müller et al. [MKN^{*}04], Pauly and colleagues [PKA^{*}05] developed a novel meshless method for fracture animation of elastoplastic solids. Steinemann et al. [SOG09] employed surface mesh representation in meshless framework and presented a novel surface tracking technique to efficiently split the meshless deforming objects. Inspired by the rigid body assumption for simulating brittle fracture, Liu et al. [LHLW11] employed quasi-static analysis in a meshless local Petrov-Galerkin framework. Stomakhin et al. modeled the fracture of snow using a meshless material point method [SSC^{*}13]. Hegemann et al. [HJST13] combined a level set based mesh embedding technique with the material point method to animate dynamic ductile fracture.

Peridynamics The peridynamics theory was first proposed by Silling [Sil00] as a nonlocal reformulation of classical solid mechanics. It contrasts with classical (local) theory in that the state of a material point is influenced by not necessarily the material points located in its immediate vicinity, but also those over long distances. The governing equations of the peridynamics theory are spatial integral equations instead of partial differential equations. The theory was further developed by subsequent works [SEW^{*}07, ELP13], and its applications to the engineering field such as multi-scale material modeling [ABL^{*}08, SC14] and fracture modeling [AXS06, SWAB10, SA14] were studied. A comprehensive review of the research literature in the computational physics community is beyond our scope, we refer the readers to the book by Madenci and Oterkus [MO14]. Levine et al. [LBC^{*}14] first introduced peridynamics to graphics for fracture animation. Their method was limited to brittle fracture of isotropic elastic materials with a single Poisson ratio of 0.25. Our work, on the other hand, is a complete framework that models elastoplasticity and anisotropy under various parameter settings, representing brittle and ductile fracture with high realism.

3. Background

The peridynamics theory models a continuum with particles, where any particle \mathbf{x} interacts with other particles within a distance δ . The distance δ is called the **horizon** of \mathbf{x} , and the particles within the horizon are referred as its **family**, $H_{\mathbf{x}}$. See Figure 2 for an illustration of the peridynamics discretization. It seems analogous to other meshless methods based on classical theory [MCG03, MKN^{*}04], and the difference lies in the scale of interaction radius δ . In the case of the classical (local) continuum model, the state of a particle is influenced by only particles in its immediate vicinity. For case of the peridynamics theory, however, the

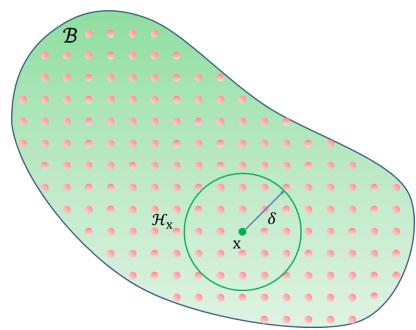


Figure 2: Illustration of peridynamics discretization. A continuum is represented as particles (pink dots), and any particle (green dot) interacts with the particles within its horizon (green circle).

state of a particle is influenced by particles within a region of finite radius. The peridynamics theory is thus referred as a nonlocal theory. As the radius becomes infinitely large, the peridynamics theory becomes the continuous version of the molecular dynamics model. As the radius becomes smaller, it becomes the continuum mechanics model. Therefore, the peridynamics model establishes a connection between the continuum mechanics and molecular dynamics models.

Our motivation for choosing peridynamics is that it is more favorable to handle material discontinuities, such as cracks. This benefit inherently from a integral force formulation of is governing equation, which stands in contrast to the partial differential equations used in the classical formulations. As we know, spatial derivatives are not well defined at discontinuities. Therefore, special treatment is generally required for fracture modeling in existing methods that are based on classical continuum mechanics. For instance, the mesh-based methods [OH99, OH02] employed re-meshing operations and the meshless method by Pauly et al. [PKA^{*}05] altered the particle weight functions. The peridynamics governing equations remain valid at discontinuities, and material damage is represented as part of the peridynamics constitutive model. These attributes permit fracture initiation and propagation to be modeled with arbitrary paths in the peridynamics framework.

In peridynamics, the governing equation at any point \mathbf{x} is formulated as below:

$$\rho \ddot{\mathbf{u}}(\mathbf{x}) = \int_{H_{\mathbf{x}}} [\mathbf{T} < \mathbf{x}' - \mathbf{x} > - \mathbf{T} < \mathbf{x} - \mathbf{x}' >] dV_{\mathbf{x}'} + \mathbf{b}(\mathbf{x}), \quad (1)$$

where ρ is the mass density, \mathbf{u} denotes the displacement, \mathbf{b} is the external loads due to gravity and impact forces, $V_{\mathbf{x}'}$ is the volume of \mathbf{x}' , and \mathbf{x}' is one material point that belongs to the family $H_{\mathbf{x}}$ of \mathbf{x} . $\mathbf{T} < \mathbf{x}' - \mathbf{x} >$ and $\mathbf{T} < \mathbf{x} - \mathbf{x}' >$ are two essential terms in which the constitutive laws of materials are encoded. $\mathbf{T} < \mathbf{x}' - \mathbf{x} >$ represents the internal force density exerted by \mathbf{x}' on \mathbf{x} , and $\mathbf{T} < \mathbf{x} - \mathbf{x}' >$ is

the other way around. The two terms both appear in the governing equation to enforce the Newton's third law, and similar strategy was employed in SPH methods [MCG03]. The angle brackets representation $\langle \cdot \rangle$ was defined by Silling et al. [SEW*07] as a function inside the family H_x , which they called as a **state**. Please note the integral form of the equation, which is the key difference between peridynamics and classical theory. The entire framework is built on displacements \mathbf{u} instead of their spatial derivatives, thereby making discontinuity modeling trivial.

4. Elastoplastic Model

In this section we describe our constitutive model for peridynamics in detail. We start with the basic isotropic elastic model, then plasticity is incorporated, and finally we extend the model with anisotropy.

4.1. Isotropic Elasticity

As is discussed in Section 3, the key to peridynamics-based constitutive modeling is the design of proper internal force density $\mathbf{T} \langle \cdot \rangle$. Silling et al. [SEW*07] showed that peridynamic constitutive models can be designed to match many hyperelastic constitutive models under the classical elasticity theory. We derive our model based on the model described by Madenci and Oterkus [MO14], which matches the isotropic linear elasticity model in classical theory. The elastic internal force density exerted by particle j on particle i is defined as below:

$$\mathbf{T}_i \langle \mathbf{x}_j - \mathbf{x}_i \rangle = \frac{1}{2} A \frac{\mathbf{y}_j - \mathbf{y}_i}{|\mathbf{y}_j - \mathbf{y}_i|}, \quad (2)$$

where \mathbf{x} and \mathbf{y} denote the positions of particles before and after deformation respectively. The direction of the force density is along the deformed bond between the particles given by $\frac{\mathbf{y}_j - \mathbf{y}_i}{|\mathbf{y}_j - \mathbf{y}_i|}$. A is a scalar that represents the force magnitude, and it is composed of two terms by addition $A = A_{dil} + A_{dev}$, namely the dilatation term A_{dil} and the deviatoric term A_{dev} .

The dilatation term A_{dil} is due to the dilatation part of deformation, i.e., volume change without any shape distortion. It is defined as:

$$A_{dil} = 4\omega_{ij}a \frac{\mathbf{y}_j - \mathbf{y}_i}{|\mathbf{y}_j - \mathbf{y}_i|} \cdot \frac{\mathbf{x}_j - \mathbf{x}_i}{|\mathbf{x}_j - \mathbf{x}_i|} \theta_i, \quad (3)$$

where a is a peridynamics material parameter and ω_{ij} is the weight function between particle i and particle j . For isotropic materials ω_{ij} is monotonically decreasing with respect to the distance between particles:

$$\omega_{ij} = \frac{\delta}{|\mathbf{x}_j - \mathbf{x}_i|}. \quad (4)$$

Note that ω_{ij} is defined in the material space, therefore can be precomputed. The term θ_i measures the dilatation at

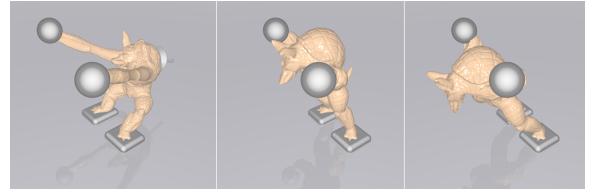


Figure 3: An armadillo is initially anchored on its back and four limbs, and it deforms elastically when its back is released.

particle i , which is defined with respect to the stretch of all bonds between particle i and its family:

$$\theta_i = \frac{9}{4\pi\delta^4} \sum_{k=1}^N \omega_{ik} s_{ik} \frac{\mathbf{y}_k - \mathbf{y}_i}{|\mathbf{y}_k - \mathbf{y}_i|} \cdot (\mathbf{x}_k - \mathbf{x}_i) V_k, \quad (5)$$

where N represents the number of family points k for point i , and V_k are their volumes. The stretch s_{ik} of the bond between particles is defined as:

$$s_{ik} = \frac{|\mathbf{y}_k - \mathbf{y}_i|}{|\mathbf{x}_k - \mathbf{x}_i|} - 1. \quad (6)$$

The deviatoric term A_{dev} is a result of distortion that does not cause change in volume. It is defined with respect to the deviatoric component of bond extension:

$$A_{dev} = 4\omega_{ij}b(e_{ij} - \frac{\delta}{4} \frac{\mathbf{y}_j - \mathbf{y}_i}{|\mathbf{y}_j - \mathbf{y}_i|} \cdot \frac{\mathbf{x}_j - \mathbf{x}_i}{|\mathbf{x}_j - \mathbf{x}_i|} \theta_i), \quad (7)$$

where b is a material parameter. We denote $e_{ij} = |\mathbf{y}_j - \mathbf{y}_i| - |\mathbf{x}_j - \mathbf{x}_i|$ as the extension of the bond between particle i and particle j . The term in brackets of Equation 7 is the deviatoric component of bond extension e_{ij}^d :

$$e_{ij}^d = e_{ij} - \frac{\delta}{4} \frac{\mathbf{y}_j - \mathbf{y}_i}{|\mathbf{y}_j - \mathbf{y}_i|} \cdot \frac{\mathbf{x}_j - \mathbf{x}_i}{|\mathbf{x}_j - \mathbf{x}_i|} \theta_i. \quad (8)$$

Intuitively, e_{ij}^d is constructed by removing the dilatation component of bond extension from the total bond extension e_{ij} .

In summary, the behavior of our isotropic elastic model is controlled by two material parameters a and b . The model is equivalent to the isotropic linear elasticity model in classical theory, please refer to the supplementary document for elaborated derivation. Here we directly provide the conversion between the material parameters in this model and those from continuum mechanics:

$$a = \frac{9\kappa}{8\pi\delta^4} \quad b = \frac{15\mu}{2\pi\delta^5}, \quad (9)$$

where κ and μ denote the bulk modulus and the shear modulus respectively. In Figure 3 we demonstrate an example of the hyperelastic deformations animated with our model.

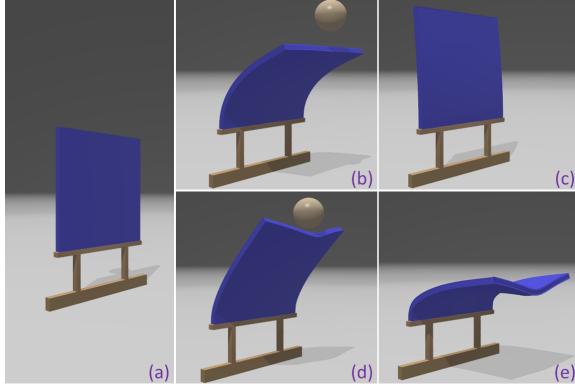


Figure 4: Comparison of elastic and plastic deformations by shooting a sphere at two walls made of different materials with identical initial configuration as in (a). The elastic wall deforms on impact (b), and recovers afterwards (c). The plastic wall undergoes permanent deformation (e) after the impact (d).

4.2. Plasticity

Plasticity model for peridynamics is less studied in literature due to its complexity. To our knowledge, Silling et al. [SEW*07] proposed the first plasticity model that is analogous to the von Mises flow model in classical theory. Mitchell presented a new framework for peridynamics-based plasticity modeling [Mit11] based on Silling et al.’s model, whereas the model hasn’t been verified by experiments thus far. We adopt Mitchell’s model for practical applications, and propose novel modifications based on their work.

Our plasticity model is based on purely deviatoric plastic flow theory, therefore we start by decomposing the deviatoric bond extension e_{ij}^d (see Equation 8) into two components by addition:

$$e_{ij}^d = e_{ij}^e + e_{ij}^p, \quad (10)$$

where e_{ij}^e and e_{ij}^p are the elastic and plastic part of the total deviatoric bond extension respectively. To incorporate plasticity into the constitutive model, the deviatoric part of internal force density (see Equation 7) is now redefined as below:

$$A_{dev} = 4\omega_{ij}b(e_{ij}^d - e_{ij}^p), \quad (11)$$

with the contribution of plastic deviatoric bond extension removed from force computation. In case of elastic deformations, the term e_{ij}^p vanishes and Equation 11 conforms to Equation 7.

A simple yield function $f(A_{dev})$ is used to determine whether deformation has entered the plasticity regime:

$$f(A_{dev}) = \frac{(A_{dev})^2}{2} - \Psi_0, \quad (12)$$

where Ψ_0 is a critical parameter. The deformation is elastic if $f(A_{dev}) \leq 0$, and plasticity is present if $f(A_{dev}) > 0$.

In case of plastic deformations, we project A_{dev} onto the yield surface to obtain a critical value of deviatoric force density A_{dev}^c :

$$A_{dev}^c = \sqrt{2\Psi_0}\text{sign}(A_{dev}), \quad (13)$$

where $\text{sign}(\cdot)$ is the sign function. A_{dev}^c is used to compute the increment of plastic deviatoric bond extension:

$$\Delta e_n^p = \frac{1}{2b}(A_{dev} - A_{dev}^c) \quad (14)$$

$$e_{n+1}^p = (e_n^p + \Delta e_n^p)\min\left(1, \frac{\gamma}{|e_n^p + \Delta e_n^p|}\right). \quad (15)$$

The subscripts n and $n+1$ denote the discretized point of time at which the bond extensions and corresponding increments are evaluated. The parameter γ which does not appear in the original model of Mitchell’s [Mit11] is used to enforce a limit on the amount of plasticity. We found in experiments that with this parameter we obtain more control over the plastic behaviors (see Figure 5) and the stability of simulation is improved as well. Figure 4 shows a comparison of the simulation results using our elastoplastic constitutive model. Our elastic model produces correct elastic behaviors, and permanent deformation is captured when plasticity is involved.

4.3. Anisotropy

Our constitutive model is isotropic up to now, and we extend it to anisotropy in this section. We model anisotropy by manipulating the weight functions between particles (see Equation 4) with direction information. The key idea is to associate an anisotropy matrix \mathbf{G} with each particle, so that applying the transformation to the bond between particles biases the influence weight toward preferred directions. The weight function ω_{ij} for anisotropic materials is computed as below:

$$\omega_{ij} = \frac{\delta}{|\mathbf{G}(\mathbf{x}_j - \mathbf{x}_i)|}. \quad (16)$$

Appealing anisotropic effects could be generated with our anisotropy model. See Figure 1 for a demonstration of the model applied to brittle and ductile fracture animation. Figure 6 compares the crack patterns generated by the brittle fracture examples in Figure 1.

5. Fracture

In this section we present our fracture model and introduce the mesh embedding strategy we employ to generate crack surfaces.

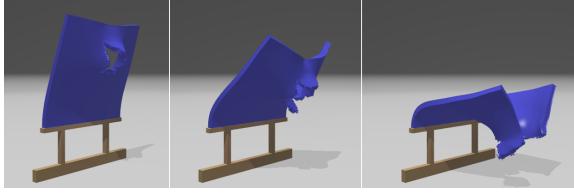


Figure 5: Simulation of ductile fracture with different amount of maximum plasticity. From left to right the parameter $\frac{\gamma}{|\mathbf{x}_j - \mathbf{x}_i|}$ is 0.1, 0.15, and 0.2.

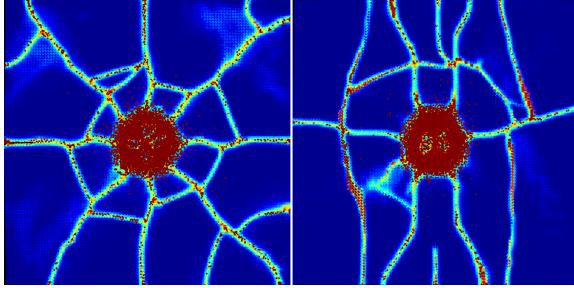


Figure 6: Comparison of crack patterns generated by isotropic (left) and anisotropic (right) brittle fracture. The color represents the damage of particles, with blue as no damage and red as complete damage.

5.1. Fracture Criterion

Material damage can be modeled in peridynamics by permanently eliminating the bonds between particles. The dynamics with discontinuities is trivial due to the integral-based nature of peridynamics. For elastic brittle materials, a simple critical stretch is generally used as the fracture criterion. This criterion conforms to the physically plausible energy release rate, and it has been validated before [SA05]. Levine et al. [LBC^{*}14] also utilized this criterion for brittle fracture modeling in computer graphics. In order to account for plasticity and model ductile fracture, we redefined the elastic critical stretch criterion as:

$$s_{ij}^e = \frac{e_{ij} - e_{ij}^p}{|\mathbf{x}_i - \mathbf{x}_j|}. \quad (17)$$

Unfortunately, we found in experiments that this fracture criterion would cause unrealistic artifacts since bonds with smaller rest lengths are sometimes more prone to breaking. We alleviate this problem by incorporating the weight function ω_{ij} into the criterion to increase the fracture criterion of closer family members. Shattering effects could arise for brittle materials, which lead to many tiny fragments. We could avoid the generation of too small fragments by continuously increasing the crack threshold as the material gets damaged. The final fracture criterion that we employ in

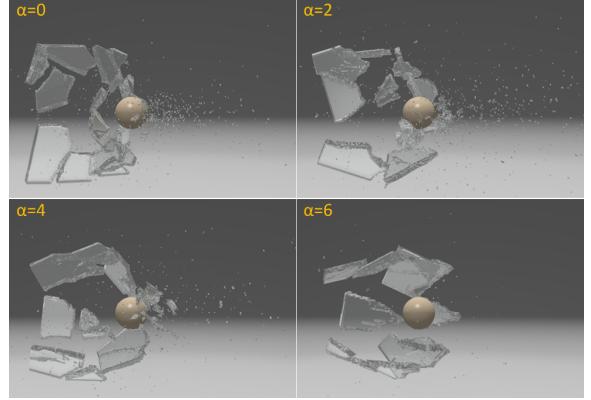


Figure 7: Glass wall fracture with different α values.

our model is formulated as below:

$$s_{ij}^\omega = (1 + \alpha\phi) \frac{s_{ij}^e}{\omega_{ij}} = (1 + \alpha\phi) \frac{e_{ij} - e_{ij}^p}{\delta}, \quad (18)$$

where $\phi = 1 - \frac{n_i}{N_i}$ measures the damage level of material point i . n_i and N_i are the numbers of active bonds connecting i with its family members in the deformed and initial configurations, respectively. n_i gradually decreases as more bonds around i are broken, increasing the damage level of i . The parameter α is set to 0 by default, and could be used to mitigate the shattering effect while non-zero values are given. Figure 7 provides an example of controlling the dust with different α values. It shows that our criterion is able to produce compelling results in practical use.

5.2. Embedded Mesh

While particle-based discretization offers great simplicity, this simplicity does come at a cost that it is difficult to generate surface representation. This naturally motivates the use of mesh embedding approach, in which the boundary of volumetric meshes could represent the original object surface and the newly generated crack surfaces during simulation. We use tetrahedron meshes to represent object geometry, and the particles are initialized at the barycenters of each mesh element. The particle family members are initialized according to the mesh connectivity and a pre-specified horizon δ . To accommodate the topology changes resulted from fracture, we propose a simple strategy that dynamically split the embedded mesh along the elements.

We achieve this by maintaining a crack face set and continuously adding the shared triangles to it for fracture happened between immediate elements. At each time step, we investigate merely those vertices that involved in crack face set and determine whether its connected tetrahedra have been separated by crack faces. If this is the case, we split the vertex, assign corresponding tetrahedra to each of them, and make a topological change on the embedded mesh. After

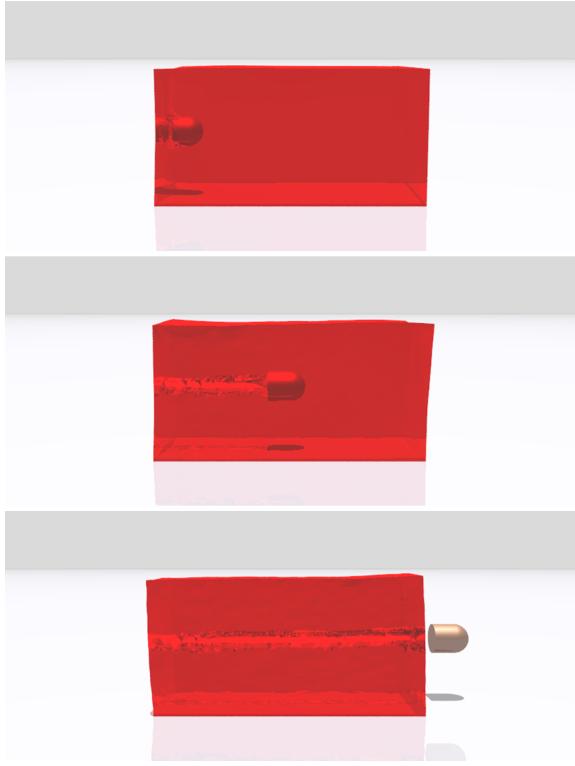


Figure 8: Shooting a bullet through a jello, producing complex crack surfaces.

splitting, we could safely remove only those crack faces that directly results in our vertex splitting. If the connected tetrahedra are separated into more than two groups, we accordingly split vertex into multiple copies. Simultaneous splitting of multiple vertices at one crack face are also compatible in our method. Figure 8 shows an example of using our strategy to produce complex crack surfaces.

After handling mesh topology, we update the vertex positions using the velocities of corresponding particles. A simple weighted-average approach is employed to update the vertex positions:

$$\omega_v = \sum_p \frac{1}{4} m_p \quad (19)$$

$$v_v = \frac{1}{\omega_v} \sum_p \frac{1}{4} m_p v_p \quad (20)$$

$$x_v^{t+1} = x_v^t + \Delta t v_v \quad (21)$$

where subscripts p and v represent the particle and the mesh vertex respectively. m_p and v_p are the mass and velocity of the particle, v_v and x_v are the velocity and position of the mesh vertex.

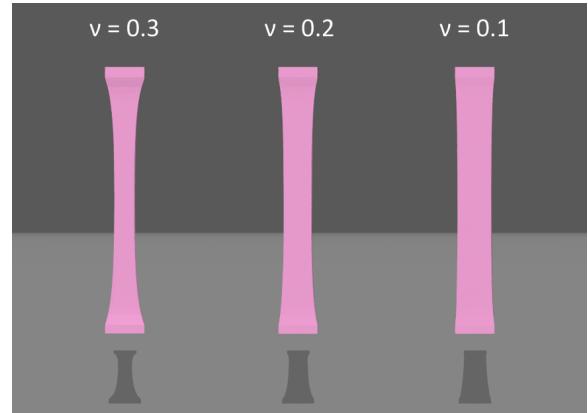


Figure 9: Simulation of stretching beams with different Poisson ratios.

6. Results

We present the results produced with our method in this section. All our examples are run on a 3.5GHz, Intel Core i7-5930K CPU with 32G RAM. The embedded tetrahedron meshes are generated using the open-source TetGen software [Si15]. All the figures in the paper are rendered off-line using the open-source POV-Ray software (<http://www.povray.org/>). We use explicit time integration for ease of implementation. We also performed a naive parallelized implementation of our method using OpenMP.

Constitutive model validation. We validate our constitutive model by simulating deformations with varied material properties and performing comparisons with the results of FEM. Figure 3 shows an example of isotropic elastic deformations, where the back and four limbs of an armadillo are initially anchored and then the anchor on the back is removed. The deformations are plausible and no undesirable numerical fracture occurs when the arms of the armadillo are over-stretched. In Figure 4 we compare the results of elasticity and plasticity, and our model produces correct deformation behaviors. Figure 5 demonstrates the varied effects produced by tuning the amount of maximum plasticity. Figure 9 compares the elastic deformations of stretching beams with different Poisson ratio values. Unlike Levine et al.'s model [LBC*14], our constitutive model is not limited to a single Poisson ratio. Finally, we conduct comparisons with FEM through Figure 10 to Figure 13. The deformations of a bending beam in Figure 10 produced with our method are almost identical to those generated by FEM, under both stiff and soft material settings. We further demonstrate the accuracy of our method in constitutive modeling using comparisons with quantitative error analysis. The swing (see Figure 11) and twist (see Figure 12) deformations of the bar produced by our method are as accurate as those by corotated linear FEM with position deviations less than 10%. The position deviations are measured over the diagonal

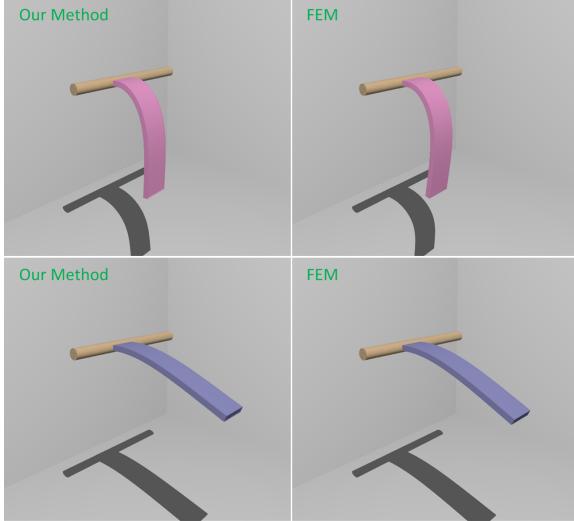


Figure 10: Simulation of bending beams with different stiffness (top:soft;bottom:stiff) using our method and FEM. The results of our method are indistinguishable from the results produced by FEM.

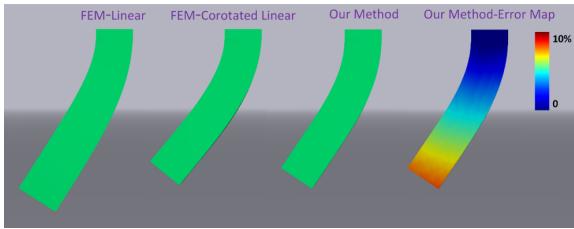


Figure 11: Simulation of a swinging bar using linear FEM (left), corotated linear FEM (middle left), and our method (middle right). Our method is accurate as the corotated linear FEM (right).

of the object's bounding box: $Error = \frac{|\mathbf{x}_p - \mathbf{x}_p^{ref}|}{d}$ where d is the length of the diagonal. Note that our constitutive model alleviates artifacts of the classical linear model albeit derived from it. It is because peridynamics does not employ the geometric linear approximation as the Cauchy strain in continuum mechanics does. Thus peridynamics does not suffer from ghost forces while undergoing rigid rotations, even though the constitutive relation is linear. We also obtain nice accuracy for the noncyclic vibrations of the armadillo presented in Figure 13. Therefore, we believe our peridynamics-based constitutive model is plausible for graphical animations.

Fracture animation. Our method could simulate brittle and ductile fracture with compelling visual realism. In Figure 1 a wide range of fracture behaviors are generated, including isotropic brittle fracture, anisotropic brittle fracture, isotrop-

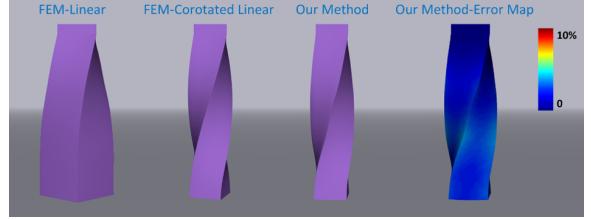


Figure 12: Simulation of a twisting bar using linear FEM (left), corotated linear FEM (middle left), and our method (middle right). Our method is accurate as the corotated linear FEM (right).

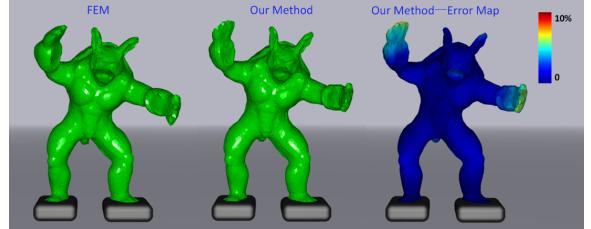


Figure 13: Simulation of noncyclic vibrations for an armadillo using FEM (left) and our method (middle). The accuracy of our method is acceptable compared with FEM (right).

ic ductile fracture, and anisotropic ductile fracture. This demonstrates the capability of our method in simulating fracture. We believe our approach is the first peridynamics-based framework in graphics with such flexibility. Figure 8 shows an example of shooting a bullet into a jello-like object. Our method handles well the generation of the complex crack surfaces. The armadillo in Figure 14 is stretched until its limbs tear off. The behavior of ductile fracture is correctly demonstrated, including the progressive generation of multiple cracks (see Figure 18). The glass wall in Figure 15 is pressed by a heavy metal ball. Cracks develop and propagate without shattering the glass into fragments. This phenomenon cannot be reproduced by the level set approach [HJST13] as mentioned in their paper, and it is challenging for the remeshing-based FEM methods [OH99, OH02] considering the mesh operations. In contrast, our method handles the complicated propagation process well, including the branching and merging of cracks. In Figure 16 a bunny made of elastic material falls to the ground and shatters. Our method is able to realistically capture the secondary fracture of the fragments. Figure 17 is another demonstration of our method in handling crack tips. A thin sheet with initial cracks is torn on two sides. The cracks gradually proceed and finally shatter the sheet. Please note the filaments generated as a result of crack branching and merging.

In Table 1 we list the detailed parameter settings and

Examples	particle num	δ	bond num	κ (KPa)	μ (KPa)	ρ (Kg/m ³)	Ψ_0	$\frac{r}{ \mathbf{x}_j - \mathbf{x}_i }$	$s_{(k)(j)}^w$	Δt (s)	performance (s/step)
Glass Wall[Figure 1(a)(b), Figure 7, 15]	3.0×10^5	1.45λ	5.0×10^7	3.3×10^7	2.0×10^7	2200	∞	0.0	0.0005	1.0×10^{-6}	~2.4
Plastic Wall [Figure 1(c)(d)]	3.0×10^5	1.45λ	5.0×10^7	3.3×10^3	2.0×10^3	2200	10^{25}	0.2	0.05	1.0×10^{-4}	~2.4
Armadillo [Figure 3]	4.2×10^5	1.50λ	7.5×10^7	6.3×10^4	3.8×10^4	1000	∞	0.0	∞	1.0×10^{-4}	~5.3
Elastic Wall [Figure 4(b)(c)]	3.0×10^5	1.45λ	5.0×10^7	1.0×10^4	6.0×10^3	1200	∞	0.0	∞	5.0×10^{-5}	~2.4
Plastic Wall [Figure 4(d)(e)]	3.0×10^5	1.45λ	5.0×10^7	1.0×10^4	6.0×10^3	1200	10^{26}	0.2	∞	5.0×10^{-5}	~2.4
Wall [Figure 5]	3.0×10^5	1.45λ	5.0×10^7	1.0×10^4	6.0×10^3	1200	10^{26}	(0.1,0.15,0.2)	∞	5.0×10^{-5}	~2.4
Jello [Figure 8]	4.6×10^5	1.45λ	8.6×10^7	1.0×10^3	4.6×10^2	1000	∞	0.0	0.4	5.0×10^{-5}	~3.8
Beam Stretch [Figure 9]	2.2×10^4	1.0λ	1.3×10^6	2.0×10^3	(9.2,1.5,2.2) $\times 10^2$	1000	∞	0.0	∞	1.0×10^{-5}	~0.2
Soft Beam Bend [Figure 10]	4.5×10^4	1.0λ	6.9×10^6	6.3×10^3	3.8×10^3	1000	∞	0.0	∞	5.0×10^{-5}	~0.6
Stiff Beam Bend [Figure 10]	4.5×10^4	1.0λ	6.9×10^6	3.3×10^4	2.0×10^4	1000	∞	0.0	∞	2.5×10^{-5}	~0.6
Bar Swing [Figure 11]	9.2×10^3	1.11λ	1.5×10^5	5.0×10^3	5.0×10^3	1000	∞	0.0	∞	1.0×10^{-4}	~0.05
Bar Twist [Figure 12]	9.2×10^3	1.34λ	1.5×10^5	1.0×10^3	6.0×10^2	1000	∞	0.0	∞	1.0×10^{-4}	~0.06
Armadillo Vibrate [Figure 13]	2.0×10^4	1.38λ	1.8×10^6	5.0×10^3	3.0×10^3	1000	∞	0.0	∞	5.0×10^{-4}	~0.2
Armadillo [Figure 14]	4.2×10^5	1.50λ	7.5×10^7	6.3×10^4	3.8×10^4	1000	∞	0.0	0.61	1.0×10^{-4}	~5.3
Bunny [Figure 16]	5.2×10^5	1.45λ	8.8×10^7	2.5×10^2	1.2×10^2	1000	∞	0.0	0.13	5.0×10^{-4}	~5.2
Thin Sheet [Figure 17]	1.6×10^5	1.45λ	2.0×10^7	5.0×10^3	3.0×10^3	1000	∞	0.0	0.1	5.0×10^{-5}	~1.8

Table 1: Model information, simulation parameters, and performance data for all our examples. λ is the average edge length of the embedded tetrahedron mesh used for particle initiation.

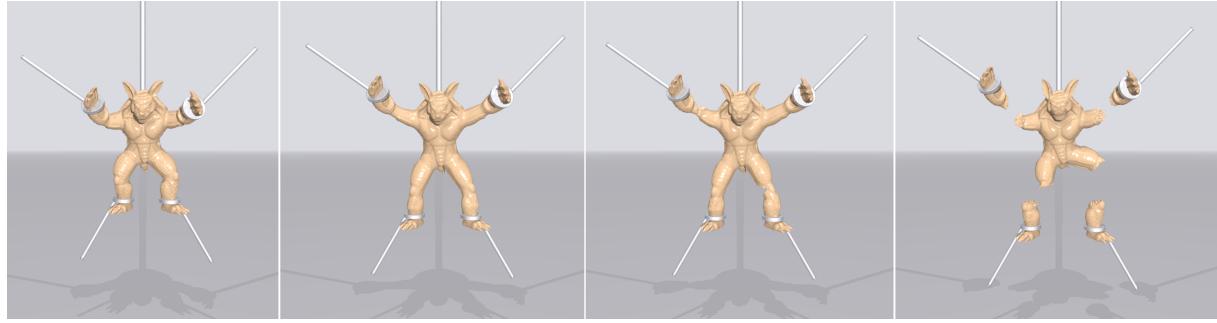


Figure 14: An armadillo is stretched until its limbs tear off.

the performance data for all the examples presented in the paper. The results in this paper are produced with practical computation time.

7. Discussions

We have introduced a novel meshless framework for graphical modeling and animation of elastoplastic solids. Our work is the first peridynamics-based framework in computer graphics that can simulate a wide range of material behaviors, including elasticity, plasticity, and fracture.

Our work is not without limitations. Currently we can not afford large time steps because we used explicit time integration in our implementation. In the future, we plan to incorporate implicit time integration into our framework to achieve less restricted time step size. Another limitation of our work stems from the mesh embedding strategy for crack surface representation. The level of crack detail is highly dependent on the embedded mesh resolution. In addition, we generate crack surfaces by separating the mesh elements, which could cause the zig-zag artifact (see Figure 18). This artifact might be alleviated by smoothing the crack surfaces somewhat, or employing the virtual node algorithm. While we used multi-threading in our implementation, the

performance of our method could be further improved with GPGPU techniques.

Other interesting avenues for future work include combining peridynamics theory with existing methods from classical theory such as FEM and SPH. These methods complement each other, and their combination could produce a new method that is more powerful for animation production.

References

- [ABL*08] ASKARI E., BOBARU F., LEHOUCQ R., PARKS M., SILLING S., WECKNER O.: Peridynamics for multiscale materials modeling. In *Journal of Physics: Conference Series* (2008), vol. 125, IOP Publishing, p. 012078. 3
- [AXS06] ASKARI E., XU J., SILLING S.: Peridynamic analysis of damage and failure in composites. In *44th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada*. Reston, VA: AIAA (2006). 3
- [BG00] BIELSER D., GROSS M. H.: Interactive simulation of surgical cuts. In *Proceedings of the 8th Pacific Conference on Computer Graphics and Applications* (Washington, DC, USA, 2000), PG '00, IEEE Computer Society, pp. 116–. 2
- [BHTF07] BAO Z., HONG J.-M., TERAN J., FEDKIW R.: Fracturing rigid materials. *IEEE Transactions on Visualization and Computer Graphics* 13, 2 (Mar. 2007), 370–378. 2

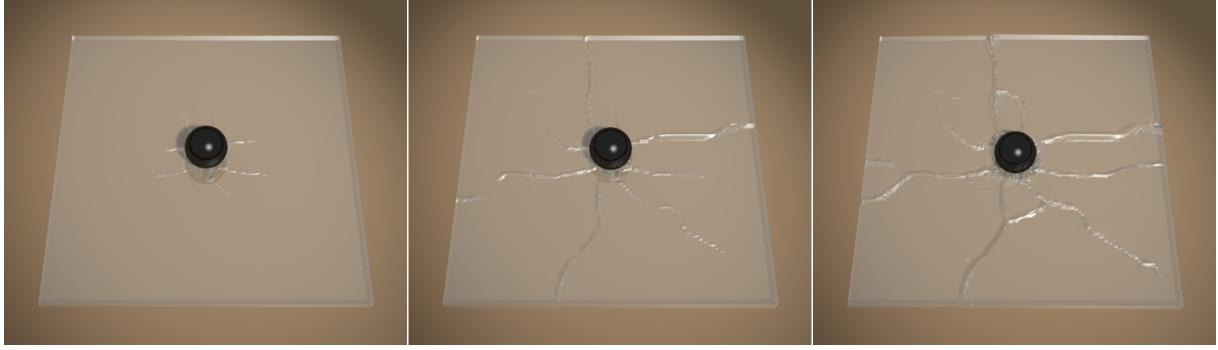


Figure 15: A glass wall on ground is pressed by a heavy metal ball and cracks without separating. Our method can model the complicated crack propagation process, including branching and merging.

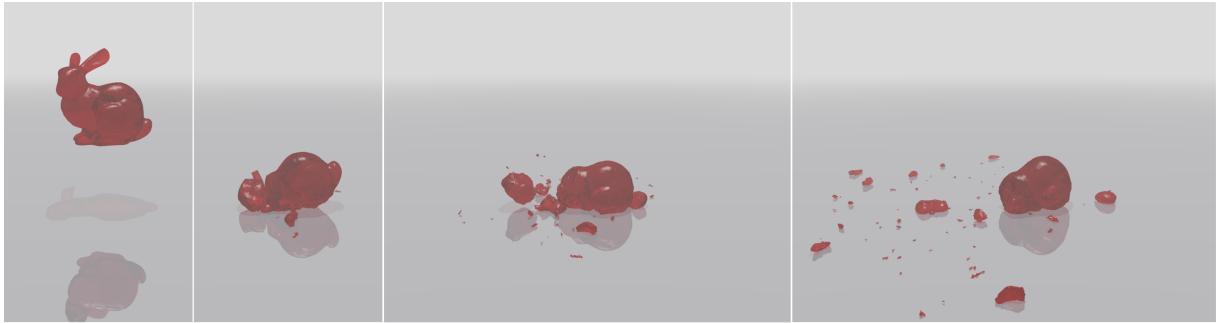


Figure 16: An elastic bunny falls to the ground and shatters into pieces. Note the secondary fracture of the fragments.

- [BWHT07] BARGTEIL A. W., WOJTAN C., HODGINS J. K., TURK G.: A finite element method for animating large viscoplastic flow. *ACM Trans. Graph.* 26, 3 (July 2007). 2
- [ELP13] EMMRICH E., LEHOUCQ R. B., PUHST D.: Peridynamics: a nonlocal continuum theory. In *Meshfree Methods for Partial Differential Equations VI*. Springer, 2013, pp. 45–65. 2, 3
- [FDA02] FOREST C., DELINGETTE H., AYACHE N.: Removing tetrahedra from a manifold mesh. In *Computer Animation, 2002. Proceedings of* (2002), IEEE, pp. 225–229. 2
- [GBB09] GERSZEWSKI D., BHATTACHARYA H., BARGTEIL A. W.: A point-based method for animating elastoplastic solids. In *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (New York, NY, USA, 2009), SCA ’09, ACM, pp. 133–138. 2
- [GMD13] GLONDU L., MARCHAL M., DUMONT G.: Real-time simulation of brittle fracture using modal analysis. *Visualization and Computer Graphics, IEEE Transactions on* 19, 2 (2013), 201–209. 2
- [HJST13] HEGEMANN J., JIANG C., SCHROEDER C., TERAN J. M.: A level set method for ductile fracture. In *Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (New York, NY, USA, 2013), SCA ’13, ACM, pp. 193–201. 3, 8
- [HW15] HAHN D., WOJTAN C.: High-resolution brittle fracture simulation with boundary elements. *ACM Trans. Graph.* 34, 4 (July 2015), 151:1–151:12. 2
- [ITF04] IRVING G., TERAN J., FEDKIW R.: Invertible finite elements for robust simulation of large deformation. In *Proceedings of the 2004 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Aire-la-Ville, Switzerland, Switzerland, 2004), SCA ’04, Eurographics Association, pp. 131–140. 1, 2
- [LBC*14] LEVINE J. A., BARGTEIL A. W., CORSI C., TESSENDORF J., GEIST R.: A peridynamic perspective on spring-mass fracture. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Aire-la-Ville, Switzerland, Switzerland, 2014), SCA ’14, Eurographics Association, pp. 47–55. 2, 3, 6, 7
- [LHLW11] LIU N., HE X., LI S., WANG G.: Meshless simulation of brittle fracture. *Comput. Animat. Virtual Worlds* 22, 2-3 (Apr. 2011), 115–124. 3
- [LZLW11] LIU N., ZHU F., LI S., WANG G.: Anisotropic kernels for meshless elastic solids. In *Proceedings of the 2011 12th International Conference on Computer-Aided Design and Computer Graphics* (Washington, DC, USA, 2011), CADGRAPHICS ’11, IEEE Computer Society, pp. 349–356. 2
- [MBF04] MOLINO N., BAO Z., FEDKIW R.: A virtual node algorithm for changing mesh topology during simulation. *ACM Trans. Graph.* 23, 3 (Aug. 2004), 385–392. 2
- [MBP14] MUGUERCIA L., BOSCH C., PATOW G.: Fracture modeling in computer graphics. *Computers & Graphics* 45 (2014), 86–100. 2
- [MCG03] MÜLLER M., CHARYPAR D., GROSS M.: Particle-based fluid simulation for interactive applications. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Aire-la-Ville, Switzerland, Switzerland,

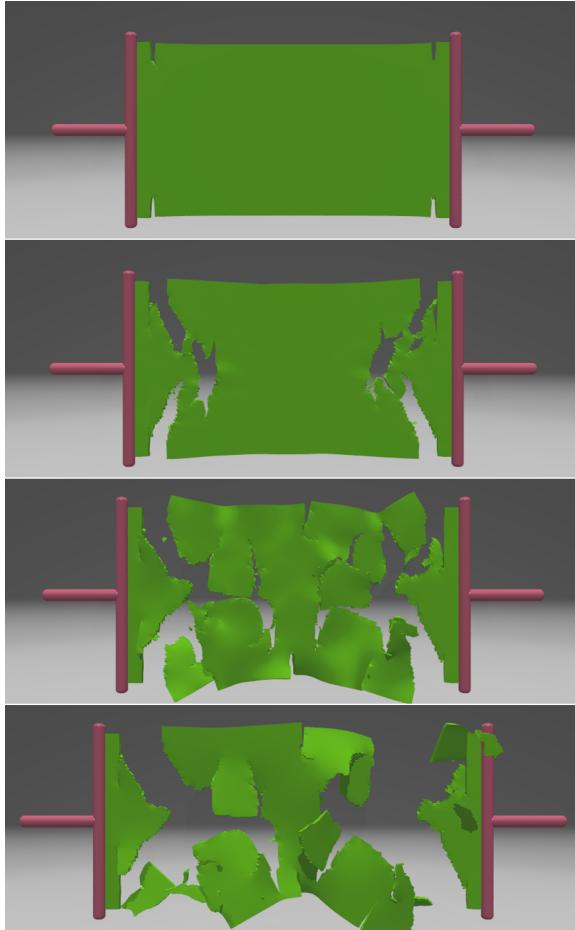


Figure 17: A thin sheet with initial cracks is torn apart. The cracks proceed with branching and merging, separating the sheet into pieces.

2003), SCA '03, Eurographics Association, pp. 154–159. [2](#), [3](#)

[\[MCK13\]](#) MÜLLER M., CHENTANEZ N., KIM T.-Y.: Real time dynamic fracture with volumetric approximate convex decompositions. *ACM Trans. Graph.* 32, 4 (July 2013), 115:1–115:10. [2](#)

[\[MG04\]](#) MÜLLER M., GROSS M.: Interactive virtual materials. In *Proceedings of Graphics Interface 2004* (School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 2004), GI '04, Canadian Human-Computer Communications Society, pp. 239–246. [1](#)

[\[Mit11\]](#) MITCHELL J. A.: A nonlocal, ordinary, state-based plasticity model for peridynamics. *SAND report 7597* (2011). [2](#), [5](#)

[\[MK00\]](#) MOR A. B., KANADE T.: Modifying soft tissue models: Progressive cutting with minimal new element creation. In *Proceedings of the Third International Conference on Medical Image Computing and Computer-Assisted Intervention* (London, UK, UK, 2000), MICCAI '00, Springer-Verlag, pp. 598–607. [2](#)

[\[MKN*04\]](#) MÜLLER M., KEISER R., NEALEN A., PAULY

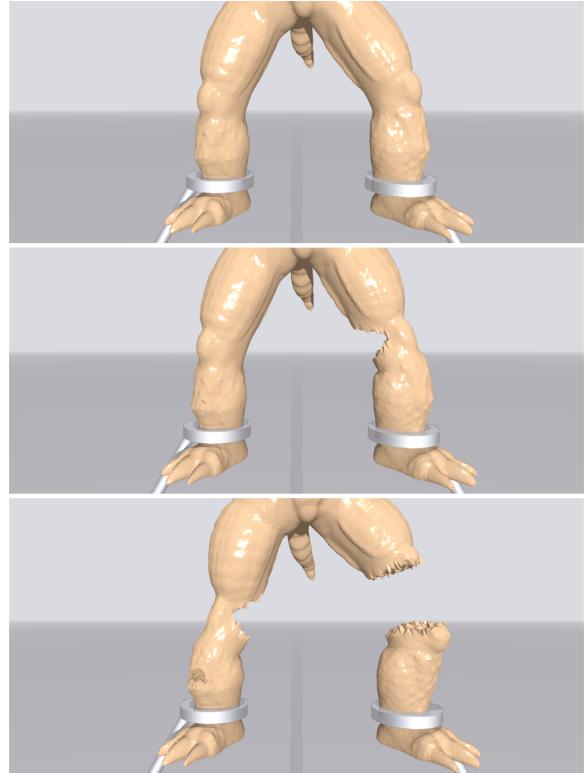


Figure 18: A close view of the progressive fracture of the armadillo's legs.

M., GROSS M., ALEXA M.: Point based animation of elastic, plastic and melting objects. In *Proceedings of the 2004 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Aire-la-Ville, Switzerland, Switzerland, 2004), SCA '04, Eurographics Association, pp. 141–151. [2](#), [3](#)

[\[MMA99\]](#) MAZARAK O., MARTINS C., AMANATIDES J.: Animating exploding objects. In *Proceedings of the 1999 Conference on Graphics Interface '99* (San Francisco, CA, USA, 1999), Morgan Kaufmann Publishers Inc., pp. 211–218. [2](#)

[\[MMDJ01\]](#) MÜLLER M., McMILLAN L., DORSEY J., JAGNOW R.: Real-time simulation of deformation and fracture of stiff materials. In *Proceedings of the Eurographic Workshop on Computer Animation and Simulation* (New York, NY, USA, 2001), Springer-Verlag New York, Inc., pp. 113–124. [2](#)

[\[MO14\]](#) MADENCI E., OTERKUS E.: *Peridynamic theory and its applications*. Springer, 2014. [2](#), [3](#), [4](#)

[\[NF99\]](#) NEFF M., FIUME E.: A visual model for blast waves and fracture. In *Proceedings of the 1999 Conference on Graphics Interface '99* (San Francisco, CA, USA, 1999), Morgan Kaufmann Publishers Inc., pp. 193–202. [2](#)

[\[NTB*91\]](#) NORTON A., TURK G., BACON B., GERTH J., SWEENEY P.: Animation of fracture by physical modeling. *Vis. Comput.* 7, 4 (July 1991), 210–219. [2](#)

[\[OBH02\]](#) O'BRIEN J. F., BARGTEIL A. W., HODGINS J. K.: Graphical modeling and animation of ductile fracture. *ACM Trans. Graph.* 21, 3 (July 2002), 291–294. [2](#), [3](#), [8](#)

[\[OH99\]](#) O'BRIEN J. F., HODGINS J. K.: Graphical modeling

- and animation of brittle fracture. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1999), SIGGRAPH '99, ACM Press/Addison-Wesley Publishing Co., pp. 137–146. [2](#), [3](#), [8](#)
- [PKA*05] PAULY M., KEISER R., ADAMS B., DUTRÉ P., GROSS M., GUIBAS L. J.: Meshless animation of fracturing solids. *ACM Trans. Graph.* 24, 3 (July 2005), 957–964. [2](#), [3](#)
- [PO09] PARKER E. G., O'BRIEN J. F.: Real-time deformation and fracture in a game environment. In *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (New York, NY, USA, 2009), SCA '09, ACM, pp. 165–175. [2](#)
- [SA05] SILLING S. A., ASKARI E.: A meshfree method based on the peridynamic model of solid mechanics. *Comput. Struct.* 83, 17–18 (June 2005), 1526–1535. [6](#)
- [SA14] SILLING S. A., ASKARI A.: *Peridynamic model for fatigue cracking*. Tech. rep., Sandia National Laboratories (SNL-NM), Albuquerque, NM (United States), 2014. [3](#)
- [SB12] SIFAKIS E., BARBIC J.: Fem simulation of 3d deformable solids: A practitioner's guide to theory, discretization and model reduction. In *ACM SIGGRAPH 2012 Courses* (New York, NY, USA, 2012), SIGGRAPH '12, ACM, pp. 20:1–20:50. [1](#)
- [SC14] SILLING S. A., COX J. V.: Hierarchical multiscale method development for peridynamics. *SAND Report 18565* (2014). [3](#)
- [SDF07] SIFAKIS E., DER K. G., FEDKIW R.: Arbitrary cutting of deformable tetrahedralized objects. In *Proceedings of the 2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Aire-la-Ville, Switzerland, Switzerland, 2007), SCA '07, Eurographics Association, pp. 73–80. [2](#)
- [SEW*07] SILLING S. A., EPTON M., WECKNER O., XU J., ASKARI E.: Peridynamic states and constitutive modeling. *Journal of Elasticity* 88, 2 (2007), 151–184. [2](#), [3](#), [4](#), [5](#)
- [Si15] SI H.: Tetgen, a delaunay-based quality tetrahedral mesh generator. *ACM Trans. Math. Softw.* 41, 2 (Feb. 2015), 11:1–11:36. [7](#)
- [Si100] SILLING S.: Reformulation of elasticity theory for discontinuities and long-range forces. *Journal of the Mechanics and Physics of Solids* 48, 1 (2000), 175 – 209. [2](#), [3](#)
- [SO14] SCHVARTZMAN S. C., OTADUY M. A.: Fracture animation based on high-dimensional voronoi diagrams. In *Proceedings of the 18th Meeting of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games* (New York, NY, USA, 2014), I3D '14, ACM, pp. 15–22. [2](#)
- [SOG09] STEINEMANN D., OTADUY M. A., GROSS M.: Splitting meshless deforming objects with explicit surface tracking. *Graph. Models* 71, 6 (Nov. 2009), 209–220. [3](#)
- [SSC*13] STOMAKHIN A., SCHROEDER C., CHAI L., TERAN J., SELLE A.: A material point method for snow simulation. *ACM Trans. Graph.* 32, 4 (July 2013), 102:1–102:10. [2](#), [3](#)
- [SWAB10] SILLING S., WECKNER O., ASKARI E., BOBARU F.: Crack nucleation in a peridynamic solid. *International Journal of Fracture* 162, 1-2 (2010), 219–227. [3](#)
- [SWB01] SMITH J., WITKIN A., BARAFF D.: Fast and controllable simulation of the shattering of brittle objects. *Comput. Graph. Forum* 20, 2 (2001), 81–90. [2](#)
- [TF88] TERZOPOULOS D., FLEISCHER K.: Modeling inelastic deformation: Viscoelasticity, plasticity, fracture. *SIGGRAPH Comput. Graph.* 22, 4 (June 1988), 269–278. [2](#)
- [TPBF87] TERZOPOULOS D., PLATT J., BARR A., FLEISCHER K.: Elastically deformable models. *SIGGRAPH Comput. Graph.* 21, 4 (Aug. 1987), 205–214. [1](#)
- [TSIF05] TERAN J., SIFAKIS E., IRVING G., FEDKIW R.: Robust quasistatic finite elements and flesh simulation. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (New York, NY, USA, 2005), SCA '05, ACM, pp. 181–190. [1](#)
- [WJST14] WANG Y., JIANG C., SCHROEDER C., TERAN J.: An adaptive virtual node algorithm with robust mesh cutting. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Aire-la-Ville, Switzerland, Switzerland, 2014), SCA '14, Eurographics Association, pp. 77–85. [2](#)
- [WWD15] WU J., WESTERMANN R., DICK C.: A survey of physically based simulation of cuts in deformable bodies. *Comput. Graph. Forum* 34, 6 (Sept. 2015), 161–187. [2](#)
- [ZBG15] ZHU Y., BRIDSON R., GREIF C.: Simulating rigid body fracture with surface meshes. *ACM Trans. Graph.* 34, 4 (July 2015), 150:1–150:11. [2](#)
- [ZZL*16] ZHU F., ZHAO J., LI S., TANG Y., WANG G.: Dynamically enriched mpmb for invertible elasticity. *Comput. Graph. Forum* (2016). In Press. [2](#)