

A Hybrid Material Point Method for Frictional Contact with Diverse Materials

XUCHEN HAN, University of California, Los Angeles, USA

THEODORE F. GAST, JIXIE EFFECTS, USA

QI GUO, University of California, Los Angeles, USA

STEPHANIE WANG, University of California, Los Angeles, USA

CHENFANFU JIANG, University of Pennsylvania, USA

JOSEPH TERAN, University of California, Los Angeles, USA



Fig. 1. Montage. Left: Simulation of a mannequin breaking through an elastic wall. Middle: Hair of a dancer in motion. Right: Colored sand and elastic characters are poured into a cabinet, setting rigid pinwheels in motion.

We present a new hybrid Lagrangian Material Point Method for simulating elastic objects like hair, rubber, and soft tissues that utilizes a Lagrangian mesh for internal force computation and an Eulerian mesh for self collision as well as coupling with external materials. While recent Material Point Method (MPM) techniques allow for natural simulation of hyperelastic materials represented with Lagrangian meshes, they utilize an updated Lagrangian discretization where the Eulerian grid degrees of freedom are used to take variations of the potential energy. This often coarsens the degrees of freedom of the Lagrangian mesh and can lead to artifacts. We develop a hybrid approach that retains Lagrangian degrees of freedom while still allowing for natural coupling with other materials simulated with traditional MPM, e.g. sand, snow, etc. Furthermore, while recent MPM advances allow for resolution of frictional contact with codimensional simulation of hyperelasticity, they do not generalize to the case of volumetric materials. We show that our hybrid approach resolves these issues. We demonstrate the efficacy of our technique with examples that involve elastic soft tissues coupled with

Authors' addresses: Xuchen Han, University of California, Los Angeles, 405 Hilgard Avenue, Los Angeles, CA, 90095, USA, xhan@math.ucla.edu; Theodore F. Gast, JIXIE EFFECTS, Los Angeles, USA, tgast@jixiefx.com; Qi Guo, University of California, Los Angeles, 405 Hilgard Avenue, Los Angeles, CA, 90095, USA, qiguo@ucla.edu; Stephanie Wang, University of California, Los Angeles, 405 Hilgard Avenue, Los Angeles, CA, 90095, USA, evast@ucla.edu; Chenfanfu Jiang, University of Pennsylvania, USA, cffjiang@seas.upenn.edu; Joseph Teran, University of California, Los Angeles, 405 Hilgard Avenue, Los Angeles, CA, 90095, USA, jteran@math.ucla.edu.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2019 Association for Computing Machinery.

2577-6193/2019/7-ART17 \$15.00

<https://doi.org/10.1145/3340258>

kinematic skeletons, extreme deformation, and coupling with multiple elastoplastic materials. Our approach also naturally allows for two-way rigid body coupling.

CCS Concepts: • Computing methodologies → Physical simulation.

Additional Key Words and Phrases: MPM, Elasticity, Hair, Collision, Hybrid

ACM Reference Format:

Xuchen Han, Theodore F. Gast, Qi Guo, Stephanie Wang, Chenfanfu Jiang, and Joseph Teran. 2019. A Hybrid Material Point Method for Frictional Contact with Diverse Materials. *Proc. ACM Comput. Graph. Interact. Tech.* 2, 2, Article 17 (July 2019), 24 pages. <https://doi.org/10.1145/3340258>

1 INTRODUCTION

The Material Point Method (MPM) [1994] was developed as a generalization of the Particle-In-Cell (PIC/FLIP) [1986; 1964] method to elastoplastic materials, and like PIC/FLIP, it has proven to be a very effective tool for many computer graphics problems. Phenomena like fracture/topological change, multiple material interactions, and challenging self contact scenarios with complex geometric domains are all commonplace in computer graphics applications. MPM naturally handles many of these. This was first demonstrated for snow dynamics by Stomakhin et al. [2013]. Since then a wide variety of other phenomena, particularly those that can be described as elastoplastic, have been simulated with MPM in graphics applications. This includes the dynamics of non-Newtonian fluids and foams [Ram et al. 2015; Yue et al. 2015], melting [Gao et al. 2017; Stomakhin et al. 2014], porous media [Fei et al. 2018; Gao et al. 2018a; Tampubolon et al. 2017], and frictional contact between granular materials [Daviet and Bertails-Descoubes 2016; Klár et al. 2016; Yue et al. 2018]. MPM has also been used to simulate contact and collision with volumetric elastic objects [Jiang et al. 2015; Zhu et al. 2017] and frictional contact between thin hyperelastic materials like clothing and hair [Fei et al. 2018; Guo et al. 2018; Jiang et al. 2017]. In this paper, we refer to methods that follows Sulsky et al.'s original idea to use the updated Lagrangian view and grid interpolation functions to compute deformation as traditional MPM.

However, there are drawbacks associated with MPM collision resolution. As noted in [Fu et al. 2017; Hammerquist and Nairn 2017; Jiang et al. 2015], information is typically lost when transferring from particles to grid, since there are generally many more particles than grid nodes. Even when utilizing Lagrangian meshes in the updated Lagrangian view as in [Guo et al. 2018; Jiang et al. 2017, 2015; Zhu et al. 2017] information is still lost which can lead to persistent wrinkles and apparent interaction at a distance, as discussed in [Guo et al. 2018; Jiang et al. 2017]. Volumetric elastic materials suffer from two additional drawbacks. First, while contact for materials such as grains [2016; 2016], membranes/shells and fibers [Guo et al. 2018; Jiang et al. 2017] can be envisioned as a continuum process where elastoplasticity associated with frictional contact is defined by the directions orthogonal to the grain, curve or surface, volumetric objects have no non-elastic directions for which to apply the condition. Hence, all self-collision resolution will result from volumetric elasticity, which means that frictional sliding cannot be regulated in a Coulomb fashion via plasticity. The second drawback is that the Eulerian grid spacing must be approximately the same as the edge lengths in the volumetric Lagrangian mesh. If the Eulerian grid resolution is significantly lower, there is non-negligible information loss in the transfer from particles to grid, and there will be spurious interaction at a distance. If the grid resolution is significantly higher, collisions will not be resolved (see Figure 7). This is problematic because visual separation between elastic bodies is proportionate to the Eulerian grid spacing, which therefore mandates high spatial resolution of the volumetric Lagrangian mesh to reduce separation thickness. This problem is not present when simulating cloth and hair because they admit the use of elastoplasticity frictional



Fig. 2. **Coupling hair with snow.** Our method captures the dynamics of a snowball falling on a head of hair.

contact particles [Guo et al. 2018; Jiang et al. 2017] and arbitrarily many can be added on each surface element or hair segment to accommodate high spatial grid resolution.

Our novel hybrid Lagrangian Material Point Method is designed to alleviate these drawbacks. Our approach utilizes more of the Lagrangian degrees of freedom to minimize artifacts while retaining aspects of MPM that allow for collision resolution without suffering from information loss when going from particles to grid. Our approach also resolves the Eulerian grid size (and artificial separation distance) limitations associated with volumetric elasticity, allowing for Coulomb frictional contact with volumetric elastic meshes. We support coupling with materials simulated with standard MPM discretizations and we provide for simple two-way coupling with rigid bodies. We demonstrate the effectiveness of our techniques with skinning, clothing, hair and multi-material simulation examples. In summary, our contributions are:

- Novel collision impulses defined from the MPM particle to grid transfers that resolve the drawbacks of the volumetric approaches in [Jiang et al. 2015; Zhu et al. 2017].
- A hybrid elastoplastic model for hair and strand self collision that supports bending, torsion and stretching resistance and that does not suffer from information loss in particle to grid transfers.
- Two-way coupling with rigid bodies.
- Removal of numerical cohesion between phases.
- Coupling with materials discretized with traditional MPM.

2 PREVIOUS WORK

Our method fits most naturally within the context of PIC/MPM methods, but also with hybrid approaches and those that make use of Lagrangian and Eulerian techniques for self collision. Here we discuss the relevant computer graphics techniques within these categories.

McAdams et al. [2009] use a hybrid PIC/geometric impulse technique to resolve self collision of many thin straight hairs. They assume that hair is incompressible and interpret the PIC grid



Fig. 3. **Braids.** Our method captures the dynamics of a braid by robustly resolving many collisions.

projection as a Lagrangian repulsion. They then apply the collision impulses of Bridson et al. [2002] to catch cases not resolved on the grid. Yue et al. [2018] develop a hybrid MPM/discrete element (DEM) technique. The DEM approach resolves frictional contact directly through constrained optimization and is generally much more detailed, but more expensive. MPM is used where the expense of DEM would be prohibitive, and their technique resolves the combination of these two representations. Sifakis et al. [2007] also use multiple representations of elastic materials to help resolve contact, including the use of a high-resolution surface mesh to aid in collision resolution.

Pai and colleagues [2013; 2014; 2011] pioneered a class of methods using Eulerian techniques for self collision with elastic objects. Li et al. [2013] show that the Eulerian view is useful for resolving close self contact between skin and other soft tissues. Teng et al. [2016] show that the approach can be naturally used to couple with incompressible fluids. Hybrid Eulerian/Lagrangian techniques are also useful for simulating crowd dynamics [Golas et al. 2014; Narain et al. 2009]. Our method is also similar to those of Müller et al [2015], Sifakis et al. [2008] and Wu et al. [2016]. These approaches mesh the space surrounding elastic objects and enforce positive volume and/or incompressibility constraints respectively on the air surrounding the objects to resolve collisions.

MPM techniques have proven very effective in graphics applications. Stomakhin et al. [2013] and Gaume et al. [2018] use the method to simulate snow. Various others have simulated more general granular materials like sand [Daviet and Bertails-Descoubes 2016; Klár et al. 2016], porous water and sand mixtures [Gao et al. 2018a; Tampubolon et al. 2017], viscoelastic foams and sponges

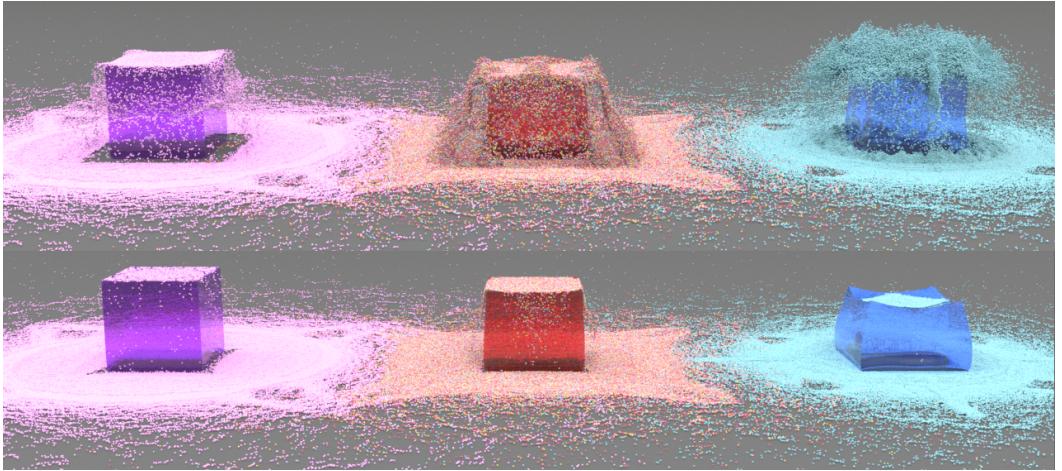


Fig. 4. **MPM particle coupling.** Elastic Jell-O's with varying stiffness are two-way coupled with MPM particles.

[Ram et al. 2015; Yue et al. 2015], coupling with rigid bodies and cutting [Hu et al. 2018], volumetric elastic materials [Jiang et al. 2015; Zhu et al. 2017], thin elastic membranes and shells [Guo et al. 2018; Jiang et al. 2017], and even wet clothing [Fei et al. 2018]. Various improvements to the method have been made, including removal of noise with angular momentum conservation [Fu et al. 2017; Jiang et al. 2015], adaptive spatial discretization [Gao et al. 2017], temporally asynchronous time stepping [Fang et al. 2018], and GPU acceleration [Gao et al. 2018b]. Also of relevance is the approach of Huang et al. [2011] to N-body collision, which has been used for self collision for fracture debris in graphical simulation of ductile fracture by Hegemann et al. [2013].

3 MATHEMATICAL BACKGROUND

Here we describe the governing equations for volumetric elastic solids and hair strands. We define the deformation of an elastic body as a map from its undeformed configuration consisting of points X to its deformed configuration consisting of points x at time t by $x(t) = \phi(X, t)$. We refer to the spatial derivative of this map as the deformation gradient $F = \frac{\partial \phi}{\partial X}$. The deformation gradient is used as a measure of strain, where its deviation from orthogonality indicates the local violation of rigid body motion. For hair, we decompose the deformation gradient into elastic and plastic parts $F = F^E F^P$, where F^E is the elastic deformation and F^P is the plastic deformation, as a means to resolve stress constraints associated with frictional contact as in [Guo et al. 2018; Jiang et al. 2017; Klár et al. 2016]. For elastic solids, we do not use an elastoplastic decomposition. Instead, we model elastic objects using hyperelasticity [Bonet and Wood 2008], where the potential energy in the system increases as ϕ deviates from rigid body motion. For frictional collision with hair strands, the potential energy density penalizes F^E . We adopt the fixed corotational model from [Stomakhin et al. 2012] for elastic solids, the Discrete Elastic Rod (DER) model from [Bergou et al. 2010, 2008] for hair and strands, and the St. Venant-Kirchhoff Hencky model from [Klár et al. 2016] for hair collision resistance.

The governing equations for the material deformation ϕ are described from conservation of mass



Fig. 5. **Hair.** A walking mannequin with a full head of hair.

and momentum

$$\frac{D\rho}{Dt} + \rho \nabla \cdot \mathbf{v} = 0, \quad \rho \frac{D\mathbf{v}}{Dt} = \nabla \cdot \boldsymbol{\sigma} + \rho \mathbf{g} \quad (1)$$

where

$$\boldsymbol{\sigma} = \frac{1}{J} \mathbf{P} \mathbf{F}^{E^T}, \quad \mathbf{P} = \frac{\partial \psi}{\partial \mathbf{F}^E}, \quad J = \det(\mathbf{F}). \quad (2)$$

ρ is material density, \mathbf{v} is velocity, \mathbf{g} is gravity constant, \mathbf{P} is the first Piola-Kirchhoff stress, and $\boldsymbol{\sigma}$ is the Cauchy stress. ψ is the potential energy density, which we assume varies with \mathbf{F}^E . For volumetric objects we do not use an elastoplastic decomposition and can so assume $\mathbf{F}^E = \mathbf{F}$ in this case.

3.1 Hyperelastic volumetric solids

For volumetric elastic objects, we adopt the fixed corotational model from [Stomakhin et al. 2012], though any hyperelastic potential may be used. With this choice, the stress satisfies

$$\begin{aligned} \psi(\mathbf{F}) &= \mu \sum_i (\sigma_i - 1)^2 + \frac{\lambda}{2} (J - 1)^2, \\ \mathbf{P} &= \mu(\mathbf{F} - \mathbf{R}) + \lambda(J - 1)\mathbf{F}^{-T}. \end{aligned} \quad (3)$$

Here μ and λ are the Lamé coefficients that express the material resistance for deformation and volume change, and σ_i are the singular values of the deformation gradient \mathbf{F} computed according to the polar SVD convention of [Irving et al. 2004] to allow for extreme deformation.

3.2 Hair strands

We follow the codimensional approaches of [Guo et al. 2018; Jiang et al. 2017] and penalize frictional contact between hairs and thin strands using a continuum assumption. Following their formulation,

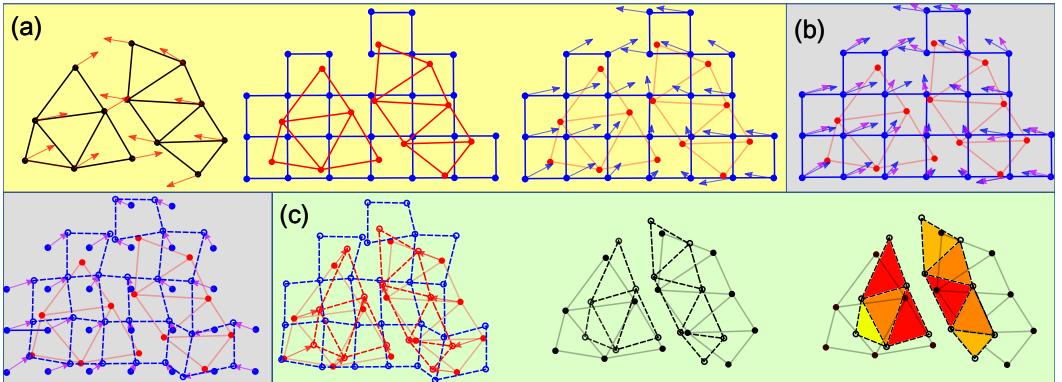


Fig. 6. MPM Overview. The steps in the MPM update are: (a) The Lagrangian quantities (black and red) are transferred to an Eulerian grid (blue), which may be viewed as a new FEM mesh. (b) Grid nodes receive new velocities (purple) from updated Lagrangian elastic updates and are temporarily moved with those velocities. (c) The Lagrangian quantities are updated by interpolating from the new positions and velocities of the Eulerian grid nodes. The triangles are colored based on the amount of compression.

we decompose the deformation of the material ϕ into the deformation of the individual strands ϕ^s and the deformation associated with frictional contact interactions among strands ϕ^d , namely

$$\phi = \phi^d \circ \phi^s. \quad (4)$$

Consequently, the deformation gradient is decomposed into $F = F^d F^s$. We treat the deformation of the strand F^s as purely elastic using standard rod and curve models [Bergou et al. 2010, 2008; Bertails et al. 2006; McAdams et al. 2009], and decompose F^d into elastic and plastic components,

$$F^d = F^{d,E} F^{d,P} \quad (5)$$

to handle frictional contact among hair strands.

We utilize the continuum Coulomb friction view from [Guo et al. 2018; Jiang et al. 2017; Klár et al. 2016] to place a constraint on admissible stress. Shear stresses resisting sliding motions between strands cannot be larger than a frictional constant times the normal stress holding them together. When the shear stress exceeds that threshold, the strands will start to slide against each other, inducing plastic deformation. Mathematically, the Coulomb friction model states that $s^T \sigma n + c_F n^T \sigma n \leq 0$, where n is the normal to the contact surface, s is any unit vector along the contact surface, and c_F is the friction coefficient. While Jiang et al. [2017] considers only directions n orthogonal to the tangent of the midline of the strand, we enforce this condition for all directions. The continuum assumption in Jiang et al. [2017] is that of a tube of parallel strands, which holds well for simulating knits but is less effective in the more complicated contact scenarios that occur when simulating hair and thin strands. To accommodate this more general constraint, we use an isotropic potential to resist collision, rather than the transversely isotropic potential of Jiang et al. [2017].

With this convention, we define the potential energy as a combination of the DER energy for strand elasticity and the St. Venant-Kirchhoff Hencky energy from [Klár et al. 2016] to penalize collision and shearing,

$$\Psi = \Psi^s(F^{d,E}) + \Psi^{DER}(F^s). \quad (6)$$

The St.Venant-Kirchhoff Hencky energy, chosen for the ease of plasticity return mapping, takes the form

$$\Psi^s = \int_{\Omega} \psi^s dV \quad (7)$$

$$\psi^s = \mu \text{tr} ((\ln \Sigma)^2) + \frac{1}{2} \lambda (\text{tr} (\ln \Sigma)^2) \quad (8)$$

where $\mathbf{F}^{d,E} = \mathbf{U}\Sigma\mathbf{V}^T$ is the singular value decomposition of the elastic deformation, Ω is the original domain the material occupies, and μ and λ are Lamé parameters. The DER energy Ψ^{DER} consists of stretching, twisting, and bending potentials. We refer readers to [Bergou et al. 2010] for details on this energy and the time parallel transport required to calculate the force. The derivatives of the potential with respect to deformation are needed for computation and satisfy

$$\frac{\partial \psi^s}{\partial \mathbf{F}^E} (\mathbf{F}^E) = \mathbf{U} (2\mu \Sigma^{-1} \ln(\Sigma) + \lambda \Sigma^{-1} \ln(\Sigma)) \mathbf{V}^T. \quad (9)$$

4 DISCRETIZATION: HYPERELASTIC SOLIDS

Our hybrid approach utilizes aspects of traditional Finite Element Methods (FEM) for hyperelasticity [Sifakis and Barbic 2012]. However, our approach is largely motivated by the the MPM treatment of volumetric objects from Jiang et al. [2015] and Zhu et al. [2017]. These methods were originally designed to prevent the numerical fracture that would occur with volumetric objects in traditional particle-based MPM. We first discuss this approach and how it resolves self collision, followed by its drawbacks.

In Jiang et al. [2015] and Zhu et al. [2017], the state at time t^n consists of particles with positions \mathbf{x}_p^n connected with a tetrahedron mesh with elements indexed by e , as in Lagrangian FEM. Furthermore, particles store velocities \mathbf{v}_p^n and masses m_p . The MPM time step from time t^n to t^{n+1} consists of three steps: (1) mass (m_p) and momentum ($m_p \mathbf{v}_p^n$) are transferred from particles to the grid using weights ($w_{ip}^n = N(\mathbf{x}_p^n - \mathbf{x}_i)$) that describe the degree of interaction between particle p and grid node i and which are defined by Eulerian grid interpolation functions $N(\mathbf{x})$, (2) the grid momentum ($m_i^n \mathbf{v}_i^n$) is updated in a variational way from the potential energy in the system and finally, (3) the motion of the grid under the updated momentum is interpolated to the particles. The process of updating the grid momentum in step (2) uses the updated Lagrangian [Belytschko et al. 2013; Guilkey and Weiss 2003; Jiang et al. 2016] convention where the time t^n configuration serves as the reference, rather than the $t = 0$ configuration in a Lagrangian discretization. With this updated Lagrangian convention, the particles \mathbf{x}_p^n are moved by the grid via interpolation $\mathbf{x}_p^{n+1} = \sum_i \mathbf{x}_i^{n+1} w_{ip}^n$, and they change the potential energy via the per-element deformation gradient computed as in standard FEM (see Equation (10)). The grid node vertices \mathbf{x}_i , which are allowed to move temporarily as $\mathbf{x}_i^{n+1} = \mathbf{x}_i + \Delta t \mathbf{v}_i^{n+1}$, serve as degrees of freedom. When the spatial discretization is done variationally from the potential energy, this step is almost identically what is done in a Lagrangian FEM discretization of elastoplasticity [Sifakis and Barbic 2012]. In this sense, the method can be interpreted as continually remeshing the domain of the material, where the transfer process in step (1) is all that is needed to define the mesh at a given time step (see Figure 6). We refer the reader to [Jiang et al. 2015, 2016] for more basic MPM details.

The MPM update only considers the variation of the potential energy with respect to grid degrees of freedom; nothing explicit is done to model self collision. Self collision is modeled as if it were an elastic phenomenon, and by virtue of switching between particle and grid representations.

We describe these two aspects of collision resolution as **type (i)** and **type (ii)**.

Type (i): The grid transfers in step (1) ultimately remesh the domain (see Figure 6). By transferring to the grid, and using an updated Lagrangian formulation where the grid nodes are updated based on the variation of the potential energy in Equation (6), MPM essentially uses a new FEM mesh (blue in Figure 6) to calculate the elastic update. This process creates new connections in the updated Lagrangian mesh and once they are made, collision inducing modes are penalized via the potential energy in the system (see Figure 6). For example, collision trajectories of the particles will induce compression in elements of the Eulerian grid which would be penalized from the elastic potential in the system.

Type (ii): In particle systems, collisions occur because of discontinuities in the velocity, e.g. consider two particles next to each other with opposing velocities. Transferring to and from the grid smooths the particle velocities, which ultimately prevents collision. Since the motion of the Eulerian grid after the momentum update in step (2) is interpolated to the particles using continuous interpolating functions, particle collisions cannot occur as long as the Eulerian mesh is not tangled by the motion. This can be guaranteed with a CFL restriction since the tangling is a temporal discretization artifact. In fact, an updated Lagrangian MPM simulation with no constitutive model on the particles at all can still prevent material collision, simply by virtue of the **type (ii)** interactions (see Figure 7).

These modes of collision resolution are simplistic, but limited by several drawbacks. For volumetric objects, the **type (i)** interactions are unable to regulate the potential energy with a plasticity model derived from Coulomb friction as in [Guo et al. 2018; Jiang et al. 2017]. The mesh is volumetric and therefore does not have the flexibility of codimension that can be used to model contact through the continuum. There are no directions left for plastic flow of the type designed in [Jiang et al. 2017] that could be used to satisfy the Coulomb friction stress constraints. This can lead to unregulated resistance to shearing and cohesion as the elastic potential will still increase with these modes, even though that is not consistent with Coulomb friction (see Figure 9). Furthermore, the updated Lagrangian treatment of the stress-based momentum leads to visual interaction at a distance and persistent wrinkling when the grid resolution is too low [Fu et al. 2017; Hammerquist and Nairn 2017; Jiang et al. 2015]. Additionally, when the grid resolution is too high, **type (i)** and **type (ii)** interactions have no effect and the method does not prevent collision (see Figure 7). To prevent this, the Lagrangian mesh resolution must be about the same as the Eulerian grid resolution. This is suboptimal when a coarse Lagrangian mesh suffices to resolve deformation.

4.1 Hybrid Lagrangian MPM for elastic solids

Our method is designed by abandoning the **type (i)** collision prevention for volumetric meshes and the updated Lagrangian integration of the elastic forces in general. Instead we use a splitting approach where elastic forces are applied in a Lagrangian way, and **type (ii)** interactions are integrated by MPM with no elastic force computation. We achieve this by introducing collision particles x_q^n which are sampled uniformly at random on the boundary of the volumetric elastic mesh. The mass of the collision particle m_q is found by dividing the mass of the boundary element by the number of collision particles on that element. These particles are not true degrees of freedom and are tied to the mesh during the Lagrangian update. They are then used to generate **type (ii)** collision prevention. We show that their response defines a type of impulse that can be regulated by Coulomb friction and applied to the mesh at the end of the time step. Furthermore, because the collision particles can be sampled at a density proportional to the grid spacing, we show that they remove the effect of grid resolution on collision resolution (see Figure 8).

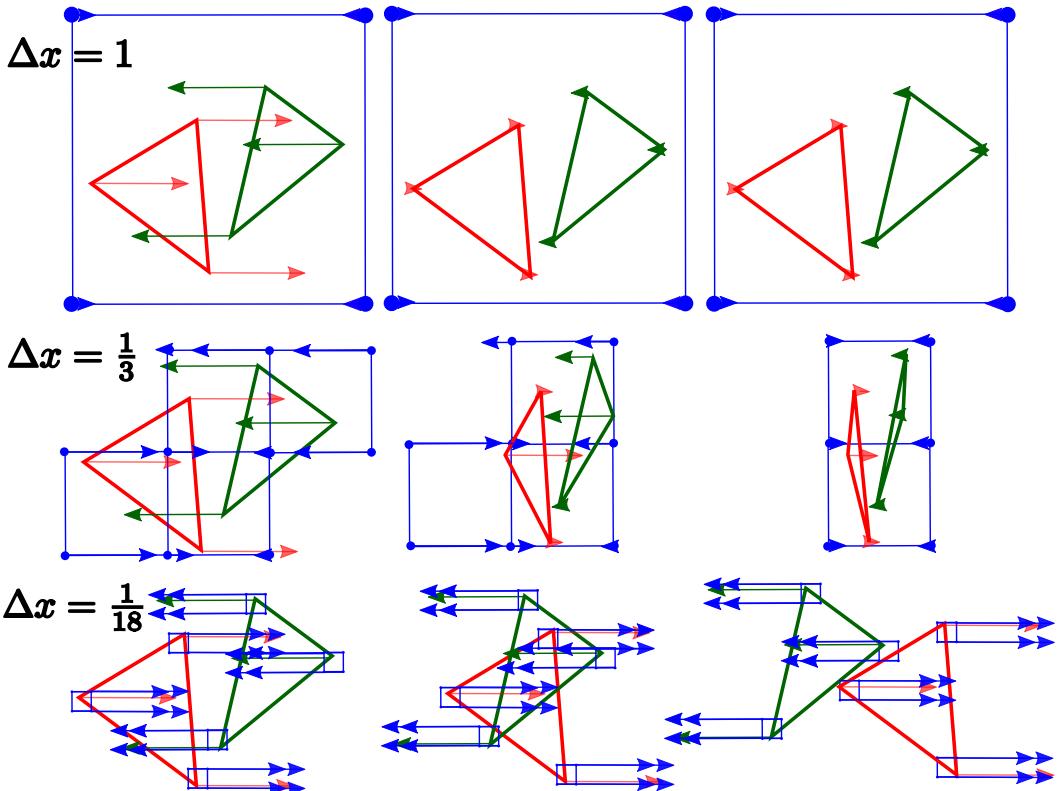


Fig. 7. Type (ii) interactions with different Δx , columns indicating consecutive time steps. At appropriate grid resolution (middle row), MPM prevents material collision even without constitutive model. However, when the grid resolution is too low (top row), objects are separated at a distance, and when the grid resolution is too high (bottom row), the MPM grids may miss a collision.

Our approach uses the same discrete state as in [Jiang et al. 2015]: time t^n , particle positions \mathbf{x}_p^n connected with a tetrahedron mesh, velocities \mathbf{v}_p^n , and masses m_p . In addition, we store the collision particles \mathbf{x}_q^n sampled on the boundary of the tetrahedron mesh. We summarize essential steps in the algorithm for updating our discrete state to time t^{n+1} below.

- (1) **Lagrangian update:** Update particle velocities from potential-energy-based and body forces, and interpolate velocities to collision particles. §4.2
- (2) **Transfer to grid:** Transfer mass and momentum from collision particles to grid. §4.3.1
- (3) **Transfer to collision particles:** Transfer velocities from grid back to collision particles. §4.3.2
- (4) **Apply impulses:** Calculate the impulse applied to each boundary mesh using the velocity change in collision particles and update velocities of particles on the boundary mesh.
- (5) **Update positions:** Update particle positions and elastic states. §4.5.

4.2 Lagrangian update

We consider the case of piecewise linear interpolation over a tetrahedron mesh. The deformation gradient varies in a piecewise constant manner with each element, which we denote as \mathbf{F}_e . With

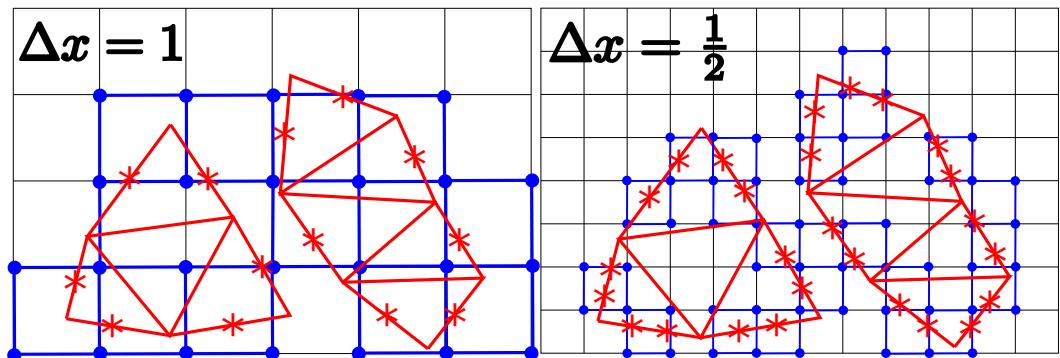


Fig. 8. **Collision particles.** Sampling density based on Eulerian grid Δx .

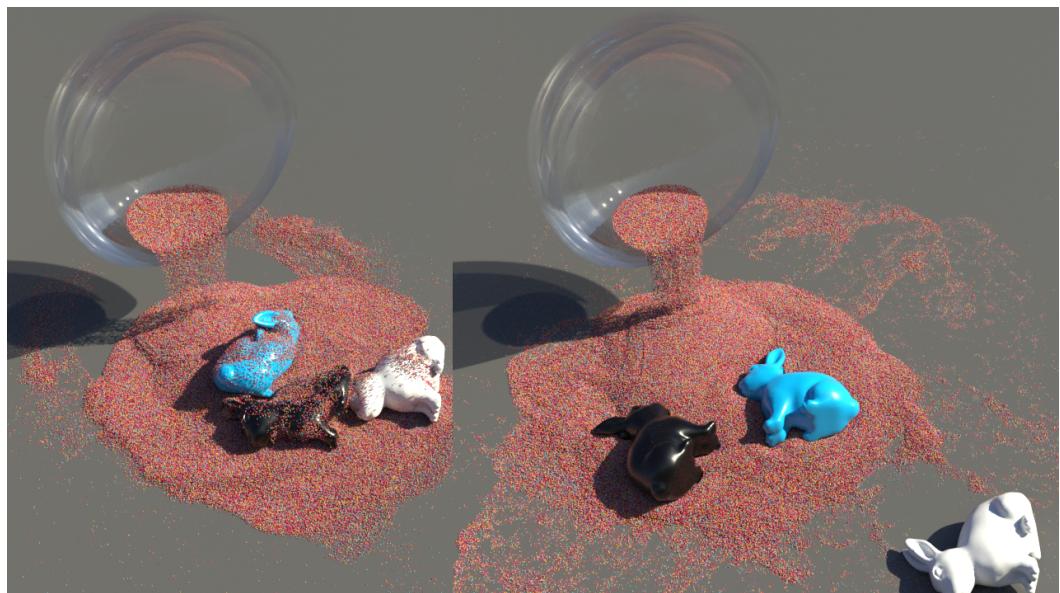


Fig. 9. **Friction.** Our method (right) removes the excessive numerical friction common to traditional MPM (left), and regulates friction with the Coulomb friction model. With low friction coefficients, the colored sand freely slides off the bunnies.

this convention, the FEM force per particle f_p can be seen as the negative gradient of the the total

potential energy Ψ with respect to particle positions:

$$\mathbf{F}_e(\mathbf{x}) = \sum_p \mathbf{x}_p \frac{\partial \tilde{N}_p}{\partial \mathbf{X}}(\mathbf{X}_e) \quad (10)$$

$$\Psi(\mathbf{x}) = \sum_e \psi(\mathbf{F}_e(\mathbf{x})) V_e^0 \quad (11)$$

$$\mathbf{f}_p(\mathbf{x}) = - \sum_e \frac{\partial \psi}{\partial \mathbf{F}}(\mathbf{F}_e(\mathbf{x})) : \frac{\partial \mathbf{F}_e}{\partial \mathbf{x}_p}(\mathbf{x}) V_e^0 \quad (12)$$

$$= - \sum_e \mathbf{P}(\mathbf{F}_e(\mathbf{x})) \frac{\partial \tilde{N}_p}{\partial \mathbf{X}} V_e^0. \quad (13)$$

Here $\mathbf{x} \in \mathbb{R}^{3n_p}$ refers to the vector of all particles \mathbf{x}_p , where n_p is the total number of particles, Ψ is the total potential energy which is a sum of tetrahedron element contributions $\psi(\mathbf{F}_e)V_e^0$, where ψ is the potential energy density in Equation (3), V_e^0 is the volume of the element in the initial state, \tilde{N}_p is the piecewise linear interpolating function associated with particle \mathbf{x}_p , and \mathbf{X}_e is the tetrahedron barycenter in the time $t = 0$ configuration. We refer the reader to Sifakis and Barbic [2012] for a more detailed derivation.

The FEM update uses the usual Lagrangian view of the governing physics. The internal force is the negative gradient of the potential energy in Equation (13). Particle velocities are updated according to forces computed at particle positions $\mathbf{x}_p^{n+\alpha}$, where symplectic Euler integration corresponds to $\alpha = 0$ and backward Euler corresponds to $\alpha = 1$:

$$\mathbf{v}_p^* = \mathbf{v}_p^n + \Delta t \frac{\mathbf{f}_p(\mathbf{x}^{n+\alpha})}{m_p}. \quad (14)$$

When damping is required while using symplectic Euler integration, we construct a background Eulerian grid with Δx comparable to the mesh size and transfer the velocity to and then back from the grid using APIC with RPIC damping as described in [Jiang et al. 2017]. We can even perform the transfers multiple times when more damping is desired. For interior particles, $\mathbf{v}_p^{n+1} = \mathbf{v}_p^*$. On the other hand, for particles on the boundary mesh, we interpolate their velocities and positions to collision particles using

$$\mathbf{v}_q^* = \sum_p b_{pq} \mathbf{v}_p^* \quad (15)$$

$$\mathbf{x}_q^n = \sum_p b_{pq} \mathbf{x}_p^n \quad (16)$$

where b_{pq} is the barycentric weight of the point q relative to p . We also assign to each point q an outward normal vector \mathbf{n}_q inherited from the face of the mesh that q is tied to.

4.3 Grid transfers

4.3.1 Particle to Grid. To process collision and contact, we transfer mass and momentum from collision particles \mathbf{x}_q^n to grid nodes \mathbf{x}_i using standard MPM transfers

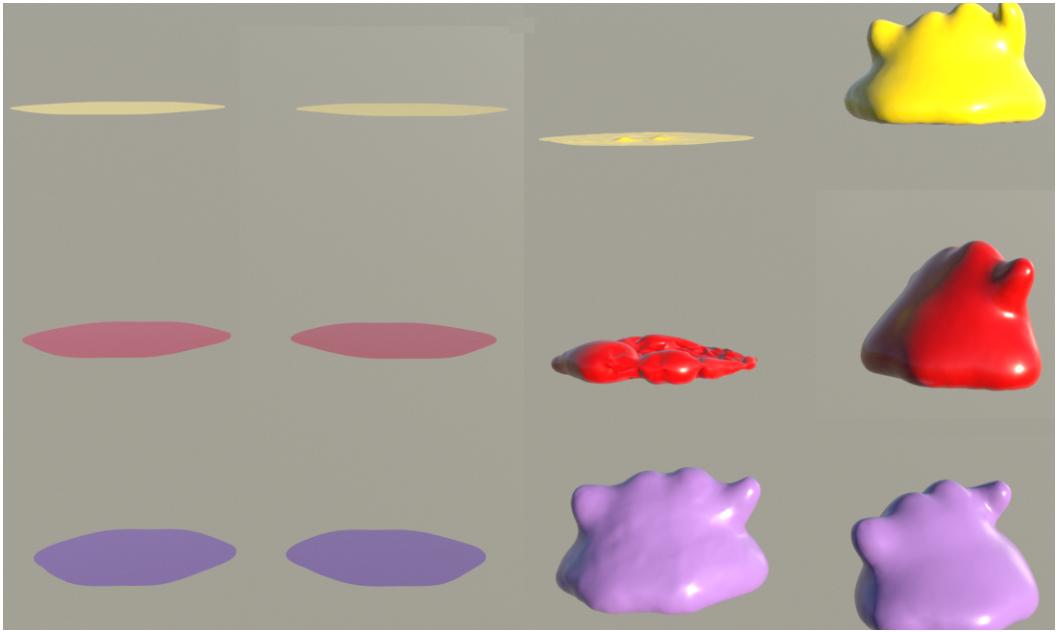


Fig. 10. **Element inversion.** MPM (left) has difficulties when elements invert, especially with low grid resolution (yellow and red). Our method (right) handles element inversions with ease.

$$m_i^n = \sum_q w_{iq}^n m_q \quad (17)$$

$$\mathbf{v}_i^* = \frac{1}{m_i^n} \sum_q w_{iq}^n m_q \mathbf{v}_q^*. \quad (18)$$

Here $w_{iq}^n = N(\mathbf{x}_q^n - \mathbf{x}_i)$ is the weight of interaction between particle \mathbf{x}_q^n and grid node \mathbf{x}_i , as in standard MPM.

4.3.2 Grid to Particle. Without any constitutive model on the grid, we proceed directly to the grid to particle step. The grid to particle transfer defines the velocity local to collision particle \mathbf{x}_q^n in terms of \mathbf{v}_q^* from

$$\mathbf{v}_q^* = \sum_i w_{iq}^n \mathbf{v}_i^*. \quad (19)$$

4.4 Apply impulse

Since the velocity \mathbf{v}_q^* is interpolated from an updated Lagrangian background grid, the boundary of the mesh is safe from self-intersection if it is moved with \mathbf{v}_q^* . However, the change may not be consistent with a Coulomb friction interaction, and the response can even be cohesive. In the case of a cohesive response after collision, we reject the change. That is, when

$$\mathbf{v}_r = \mathbf{v}_q^* - \mathbf{v}_q \quad (20)$$

$$\mathbf{v}_r \cdot \mathbf{n}_q \geq 0 \quad (21)$$

the updated Lagrangian mesh detects a separation instead of collision, and the collision particle keeps the velocity from the FEM update \mathbf{v}_q^* . On the other hand, if

$$\mathbf{v}_r \cdot \mathbf{n}_q < 0 \quad (22)$$

we apply an elastic impulse $I_q \mathbf{n}_q$ to the mesh at position \mathbf{x}_q^n where $I_q = 2m_q \mathbf{v}_r \cdot \mathbf{n}_q$. We also allow for friction using Coulomb's model with the friction parameter μ . When an elastic impulse of magnitude I_q would be applied based on condition (22), Coulomb friction admits a change in magnitude of tangential velocity of at most $-\mu \frac{I_q}{m_q}$. So the combined velocity change on collision particle q is then

$$\Delta \mathbf{v}_q = \frac{I_q \mathbf{n}_q}{m_q} + \min \left(\|\mathbf{v}_t\|, -\mu \frac{I_q}{m_q} \right) \frac{\mathbf{v}_t}{\|\mathbf{v}_t\|}, \quad (23)$$

where $\mathbf{v}_t = \mathbf{v}_r - \mathbf{v}_r \cdot \mathbf{n}_q \mathbf{n}_q$. We then transfer this change to the particles p as

$$\Delta \mathbf{v}_p = \mathbf{v}_p^{n+1} - \mathbf{v}_p^* = \sum_q \tilde{b}_{pq} \Delta \mathbf{v}_q \quad (24)$$

where

$$\tilde{b}_{pq} = \frac{b_{pq} m_q}{\sum_r b_{pr} m_r} \quad (25)$$

are the normalized weights defined from the barycentric weights used to transfer from particles to collision particles.

4.5 Update positions and elastic state

For boundary particles, we adopt symplectic Euler time integration

$$\mathbf{v}_p^{n+1} = \mathbf{v}_p^n + \Delta \mathbf{v}_p \quad (26)$$

$$\mathbf{x}_p^{n+1} = \mathbf{x}_p^n + \Delta t \mathbf{v}_p^{n+1} \quad (27)$$

For interior particles, the update is in accordance with either symplectic Euler or backward Euler, depending on the choice of α in Equation (14):

$$\mathbf{v}_p^{n+1} = \mathbf{v}_p^* \quad (28)$$

$$\mathbf{x}_p^{n+1} = \mathbf{x}_p^n + \Delta t \mathbf{v}_p^{n+1}. \quad (29)$$

5 DISCRETIZATION: HAIR STRANDS

As discussed in Section §3.2, we decompose the motion of the hair into that representing individual strand deformation ϕ^s and that of frictional sliding and compression ϕ^d . As in [Guo et al. 2018; Jiang et al. 2017], we discretize these two motions in different ways. Since ϕ^s only considers single hair strands, it suffices to discretize the energy and forces with traditional FEM. We do this using the approach of [Bergou et al. 2010, 2008]. However, unlike the approaches in [Guo et al. 2018; Jiang et al. 2017], we do not make use of an updated Lagrangian discretization of ϕ^s . To do so severely limits the ability of the hair to resolve collisions without a prohibitively high-resolution Eulerian grid (see Figure 11). Rather, we split the updates of ϕ^s and ϕ^d , where the velocities for ϕ^s are first updated in a Lagrangian manner and ϕ^d with a standard updated Lagrangian MPM discretization. We then adopt the approach of McAdams et al. [2009] where the grid-based updates are interpreted as impulsive changes in velocities on the strand that prevent self collision. However, by foregoing the updated Lagrangian discretization of ϕ^s , we cannot guarantee that self collision is prevented and thus revert to geometric impulses after the correction from ϕ^d .

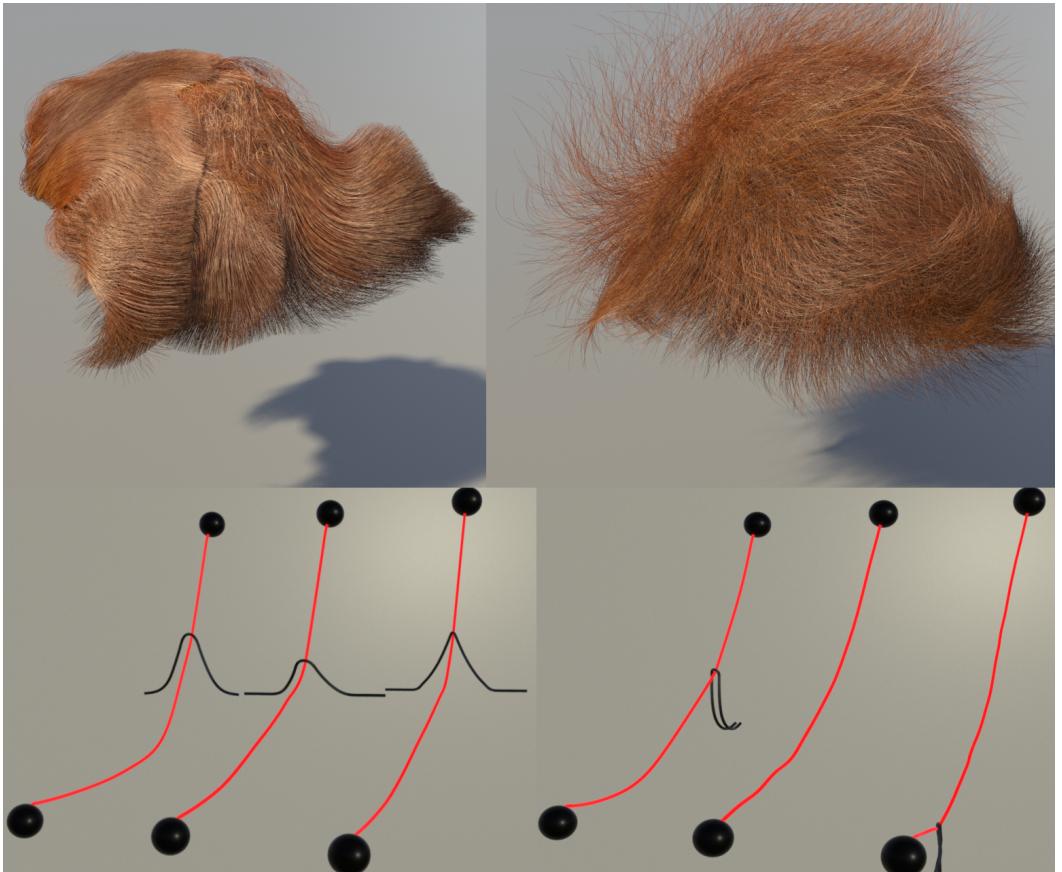


Fig. 11. Hair comparison with MPM. **Top row:** MPM simulation of hair exhibits excessive friction and cohesion whereas our method captures the rich dynamics of individual strands. **Bottom row:** We demonstrate the dynamics of two hair strands, colored black and red, at two time steps. MPM (left) results in uncontrolled friction. Hybrid method without geometric collision (middle) misses the collision. Our method (right) captures the sliding behavior between two strands.

The discrete state for each strand at time t^n consists of centerline particle positions \mathbf{x}_p^n , with velocities \mathbf{v}_p^n , masses m_p , APIC matrix C_p^n , and elastic and plastic deformation gradients associated with ϕ^d , $\mathbf{F}_p^{E,n}$ and $\mathbf{F}_p^{P,n}$. Furthermore, each edge e connecting particles \mathbf{x}_e^n and \mathbf{x}_{e+1}^n stores orientation angle θ_e as in [Bergou et al. 2010]. We summarize essential steps in the algorithm for updating the discrete state to time t^{n+1} below.

- (1) **Lagrangian update:** Update particle velocities from strand model of [Bergou et al. 2010]. §5.1
- (2) **Transfer to grid:** Transfer mass and momentum from particles to grid using APIC as in [2015].
- (3) **Update grid momentum:** Compute effect of collision potential and friction elastoplasticity. §5.2
- (4) **Apply impulses:** Interpolate the change in grid velocity to particles and then apply geometric collision handling. §5.3

(5) **Update positions:** Update particle positions as in Equation (29).

5.1 Lagrangian update

We adopt a time splitting scheme for the velocity update where the velocity is first updated according to the force induced by the energy Ψ^{DER} . Specifically, we have

$$\mathbf{v}_p^* = \mathbf{v}_p^n + \Delta t \frac{\mathbf{f}_p}{m_p} \quad (30)$$

where \mathbf{f}_p is calculated as in [Bergou et al. 2010]. This new velocity \mathbf{v}_p^* is then transferred to the MPM background grid \mathbf{v}_i^* as in Section 4.3.1.

5.2 Grid momentum update

The grid momentum is then updated according to the elastoplasticity model for the ϕ^s motion and associated potential energy Ψ^s :

$$\mathbf{v}_i^* = \mathbf{v}_i^* - \frac{dt}{m_i^n} \sum_p \frac{\partial \psi^s}{\partial \mathbf{F}_p^E} (\tilde{\mathbf{F}}_p^E(\tilde{\mathbf{x}}^{n+\alpha})) (\mathbf{F}_p^{E,n})^T \nabla w_{ip}^n V_p^0 + \Delta t g. \quad (31)$$

Here, $\tilde{\mathbf{F}}_p^E(\tilde{\mathbf{x}}^{n+\alpha})$ is the trial elastic strain and $\tilde{\mathbf{x}}^{n+\alpha}$ is the vector of all Eulerian grid node positions, moved according to

$$\mathbf{x}_i^{n+\alpha} = \mathbf{x}_i + \alpha \Delta t \mathbf{v}_i^*, \quad \tilde{\mathbf{F}}_p^E = (\mathbf{I} + \alpha \Delta t \sum_i \mathbf{v}_i^* \nabla w_{ip}^n) \mathbf{F}_p^{E,n} \quad (32)$$

where $\alpha = 0$ corresponds to symplectic Euler and $\alpha = 1$ corresponds to backward Euler for the grid momentum update. We also update APIC matrix C_p using grid velocity \mathbf{v}_i^* as in [Jiang et al. 2015, 2016].

5.3 Impulses

To interpret the motion in \mathbf{v}_i^* as inducing impulsive change in momentum on the midline, we interpolate the change in the grid velocity to the particles. However, we blend in the updated Lagrangian response weighted with parameter ξ

$$\mathbf{v}_p^* = (1 - \xi) \left(\mathbf{v}_p^* + \sum_i (\mathbf{v}_i^* - \mathbf{v}_i^*) w_{ip}^n \right) + \xi \sum_i \mathbf{v}_i^* w_{ip}^n. \quad (33)$$

This introduces a bit of the **type (i)** and **type (ii)** collision prevention, but without sacrificing the geometric detail of the Lagrangian motion. This is equivalent to the PIC/FLIP blend used in [McAdams et al. 2009]. Typically, we introduce $\xi = 0.95$. However, abandoning the updated Lagrangian update can leave collisional modes unresolved for hair. We apply geometric collision handling similar to [Bridson et al. 2002] to resolve remaining collisional modes.

Collision impulses are applied based on proximity between strand edges. We use acceleration structures for efficient proximity queries as in [Bridson et al. 2002]. However, we use regular grid-based structures inherent in MPM implementations. We divide the domain into calculation pads in space with edge length l . Then we extend the pad in the positive axis direction by proximity threshold δ so that neighboring pads have an overlap of length at least δ and thus any proximity pair will appear in at least one pad. In parallel, each extended pad collects all segments that have at least one endpoint contained in the pad, and then registers any proximity pairs contained in its set of segments. We apply an impulse to any proximity pair on a colliding trajectory as determined by relative velocity component on the direction separating the pair. The inelastic impulses from



Fig. 12. **Braiding.** Two bundles of hair are intertwined into a braid and then separated.

[Bridson et al. 2002] are then calculated and distributed to particles. Also as proposed in Bridson et al. [2002], we divide the total impulse on a particle by the number of impulses it receives from all pads and perform Jacobi iteration. After a fixed number of iterations, we obtain the particle velocity v_p^{n+1} , and then advect particles using Equation (29).

6 RIGID BODIES

Two-way rigid body coupling may be achieved with a treatment similar to volumetric elastic objects. We sample collision particles on the boundary in the same fashion as in Section 4.1 and then uniformly distribute the mass of the rigid body to the collision particles. However, we found that unlike for volumetric elastic objects, **type (ii)** interactions on the grid alone are not enough to resolve collisions. Instead we endow the collision particles with the potential described in [Guo et al. 2018; Jiang et al. 2017] to penalize contact. Specifically, we update the deformation gradient F_q from time t_n to t_{n+1} in the following way. Let \mathbf{x}_α and \mathbf{X}_α , $\alpha \in \{0, 1, 2\}$ be the current and initial positions of the vertices of the triangle that collision particle q is tied to. Let $\mathbf{D}_{q,\beta} = \mathbf{X}_\beta - \mathbf{X}_0$ be the undeformed mesh element edge vectors (where $\beta = 1, 2$), and $\hat{\mathbf{d}}_{q,\beta}^E = \mathbf{x}_\beta^n - \mathbf{x}_0^n$ be the deformed edge vectors. We choose each \mathbf{D}_3 to be unit-length and normal to \mathbf{D}_1 and \mathbf{D}_2 , and evolve each one as in traditional MPM via $\hat{\mathbf{d}}_{q,3}^E = \nabla \mathbf{x}_q \mathbf{d}_{q,3}^E$. Then $\hat{\mathbf{F}}_q^E = \hat{\mathbf{d}}_q^E \mathbf{D}_q^{-1}$. Following [Guo et al. 2018; Jiang et al. 2017], we let $\hat{\mathbf{F}}_q^E = \mathbf{Q}\hat{\mathbf{R}}$ be the QR decomposition of $\hat{\mathbf{F}}_q^E$ and design a collision energy density $\psi(\hat{\mathbf{R}}) = f(\hat{\mathbf{R}}) + g(\hat{\mathbf{R}})$,

$$f(\hat{\mathbf{R}}) = \begin{cases} \frac{k^c}{3}(1 - \hat{r}_{33})^3 & 0 \leq \hat{r}_{33} \leq 1 \\ 0 & \hat{r}_{33} > 1 \end{cases}, \quad g(\hat{\mathbf{R}}) = \frac{\gamma}{2}(\hat{r}_{13}^2 + \hat{r}_{23}^2) \quad (34)$$

where \hat{r}_{ij} is the ij -th entry of $\hat{\mathbf{R}}$. We resolve the force which is the negative derivative of this energy on the MPM background grid, and we refer the reader to [Jiang et al. 2015, 2016] for more details.

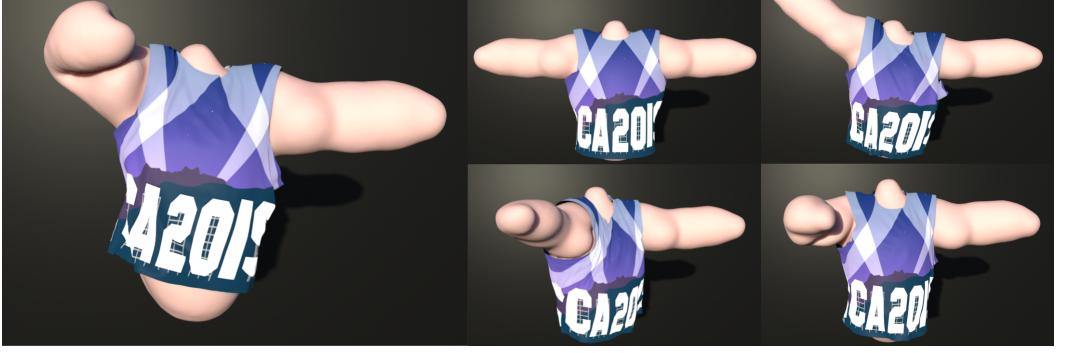


Fig. 13. **Skin and shirt.** The skin of a mannequin is coupled with clothing simulated with MPM.

Plasticity is then applied according to [Guo et al. 2018; Jiang et al. 2017] to give \mathbf{R}

$$r_{33} = \begin{cases} \hat{r}_{33} & 0 < \hat{r}_{33} \leq 1 \\ 1 & \hat{r}_{33} > 1 \end{cases}, r_{\beta 3} = h(\hat{r}_{13}, \hat{r}_{23}, r_{33})\hat{r}_{\beta 3} \quad (35)$$

$$h(\hat{r}_{13}, \hat{r}_{23}, r_{33}) = \min \left(1, \frac{c_F k^c (1 - r_{33})^2}{\gamma \sqrt{\hat{r}_{13}^2 + \hat{r}_{23}^2}} \right) \quad (36)$$

Finally, we update the deformation gradient with $\mathbf{F}_q^{n+1} = \mathbf{Q}\mathbf{R}$.

Let $\mathbf{v}_q^* = \sum_i w_{iq}^n \mathbf{v}_i^*$, where \mathbf{v}_i^* is the grid velocity after the MPM force update, and let $\mathbf{v}_r = \mathbf{v}_q^* - \mathbf{v}_q$. If $\mathbf{v}_r \cdot \mathbf{n}_q < 0$, we apply an impulse \mathbf{I}_q to the rigid bodies to update velocity \mathbf{v} and angular velocity $\boldsymbol{\omega}$ via

$$\mathbf{I}_q = m_q \mathbf{v}_r \cdot \mathbf{n}_q \quad (37)$$

$$\mathbf{v}_t = \mathbf{v}_r - \mathbf{v}_r \cdot \mathbf{n}_q \mathbf{n}_q \quad (38)$$

$$\mathbf{I}_q = I_q \mathbf{n}_q + m_q \min \left(\|\mathbf{v}_t\|, -\mu \frac{I_q}{m_q} \right) \frac{\mathbf{v}_t}{\|\mathbf{v}_t\|} \quad (39)$$

$$\mathbf{v}^{n+1} = \mathbf{v}^n + \sum_q \frac{\mathbf{I}_q}{m_q} \quad (40)$$

$$\boldsymbol{\omega}^{n+1} = \boldsymbol{\omega}^n + \sum_q \mathbf{J}^{-1}(\mathbf{r} \times \mathbf{I}_q) \quad (41)$$

where \mathbf{r} is the vector from the rigid body's center of mass to the application point of the impulse, and \mathbf{J} is the inertia tensor.

7 COUPLING WITH TRADITIONAL MPM

Our method easily couples with traditional MPM particles such as snow, sand and clothing. To prevent numerical cohesion between phases common to MPM, we adopt two separate background MPM grids, one for volumetric elastic and rigid objects, and the other for general MPM materials. We denote quantities associated with the two grids with subscripts 1 and 2 respectively. We denote quantities associated with traditional MPM particles with subscript p and quantities associated

with quadrature points with subscript q . So we have,

$$m_{1,i}^n = \sum_q w_{iq}^n m_q, m_{2,i}^n = \sum_p w_{ip}^n m_p \quad (42)$$

$$\mathbf{v}_{1,i}^* = \frac{1}{m_{1,i}^n} \sum_q w_{iq}^n m_q \mathbf{v}_q^* \quad (43)$$

$$\mathbf{v}_{2,i}^n = \frac{1}{m_{2,i}^n} \sum_p w_{ip}^n m_p (\mathbf{v}_p + C_p(\mathbf{x}_i - \mathbf{x}_p)) \quad (44)$$

$$\mathbf{n}_i^n = \frac{\sum_q w_{iq} \mathbf{n}_q}{\|\sum_q w_{iq} \mathbf{n}_q\|} \quad (45)$$

Grid velocity $\mathbf{v}_{2,i}^n$ is updated as in [Jiang et al. 2015, 2016] to get $\mathbf{v}_{2,i}^*$. Then the collision between phases is handled through an inelastic collision on collocated grid nodes.

$$\mathbf{v}_r = \mathbf{v}_{1,i}^* - \mathbf{v}_{2,i}^* \quad (46)$$

$$\mathbf{v}_t = \mathbf{v}_r - \mathbf{v}_r \cdot \mathbf{n}_i^n \mathbf{n}_i^n \quad (47)$$

$$I_i = \max \left(\frac{m_{2,i}^n m_{1,i}^n}{m_{2,i}^n + m_{1,i}^n} \mathbf{v}_r \cdot \mathbf{n}_i^n, 0 \right) \quad (48)$$

$$\mathbf{v}_{1,i}^{**} = \mathbf{v}_{1,i}^* - \frac{I_i \mathbf{n}_i}{m_{1,i}^n} - \min \left(\frac{\mu I_i}{m_{1,i}^n}, \|\mathbf{v}_t\| \right) \frac{\mathbf{v}_t}{\|\mathbf{v}_t\|} \quad (49)$$

$$\mathbf{v}_{2,i}^{n+1} = \mathbf{v}_{2,i}^* + \frac{I_i \mathbf{n}_i}{m_{2,i}^n} + \min \left(\frac{\mu I_i}{m_{2,i}^n}, \|\mathbf{v}_t\| \right) \frac{\mathbf{v}_t}{\|\mathbf{v}_t\|} \quad (50)$$

Finally, we interpolate the the grid velocity $\mathbf{v}_{2,i}^{n+1}$ to MPM particles with APIC as in [Jiang et al. 2015, 2016], and Equation (19) is replaced with

$$\mathbf{v}_q^* = \sum_i w_{iq}^n \mathbf{v}_{1,i}^{**}. \quad (51)$$

8 RESULTS

We demonstrate the efficacy of our method with a number of representative examples that illustrate the dynamics of hair and volumetric objects, and show that our method couples with granular materials, clothing and rigid bodies. We list the runtime performance for our examples in Table 1. All simulations were run on an Intel Xeon E5-2690 V2 system with 20 threads and 128GB of RAM. We report the timing in terms of average seconds of computation per frame. We chose Δt in an adaptive manner that is restricted by a CFL condition when the particle velocities are high, i.e., we do not allow particles to move further than the CFL number times Δx in a time step.

8.1 Hair

We demonstrate that our method preserves the intricate dynamics of individual hair strands and robustly handles the numerous collisions among them. In Figure 11, 32 thousand strands of hair with 60 segments per strand are simulated subject to intense boundary motions. Our algorithm is able to run this challenging example at 122 seconds per frame. In Figure 3 and Figure 12, we show that our method effortlessly resolves the intense self collisions occurring in braiding examples. In Figure 5 and Figure 1 (middle), we show a mannequin with a full head of hair in motions common in everyday life, such as walking and dancing. In Figure 15, we compare our method with McAdams et al. [2009] in a numerical experiment where a bundle of hair strands falls and

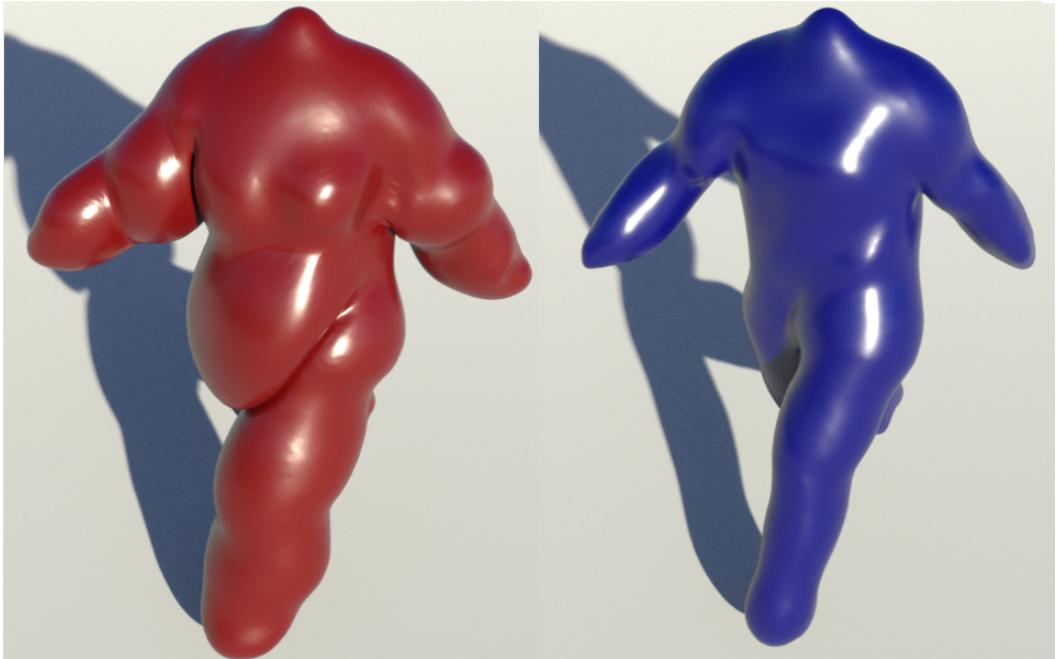


Fig. 14. Walking mannequins. Our method handles the numerous collisions occurring in the scene with walking characters.

bounces off another bundle. The experiments are run with a total of 2700 hair strands with 175 segments per strand. Five iterations of impulse application are applied to resolve the collisions missed by advecting the segments with the velocity in Equation (33) in our method and the velocity satisfying incompressibility condition in [McAdams et al. 2009]. Notice that our method preserves the volume of the hair bundle and does not suffer from numerical cohesion. We run the test for 100 frames until the hair bundles are apparently separated and track the missed collisions in the process by calculating the collision interactions between pairs of segments using the cubic solve proposed in [Bridson et al. 2002]. The test using McAdams et al. [2009] registers more than 543 thousand missed collisions whereas the test using our method registers 120 missed collisions. Our method runs three times faster (see Table 1). Note that our method not only avoids the expensive Poisson solve for incompressibility, but it also serves as a better approximate collision response and therefore reduces the runtime and number of missed collisions in the collision impulse step. We plot the total energy in the test run with our method in Figure 16.

8.2 Volumetric objects

We demonstrate the robustness of our method for resolving collisions between volumetric objects. Our method correctly resolves frictional sliding without artifacts. In Figure 14, we show a skin simulation with walking characters in various body shapes. In Figure 9, we compare our approach with updated Lagrangian MPM, which exhibits excessive cohesion and numerical friction. We also show that our method removes the requirement of comparable grid and mesh resolution. We use a moderate resolution Lagrangian mesh to resolve the dynamics of the bunnies and a high resolution Eulerian grid to resolve more detailed behaviors of the sand. In contrast, updated Lagrangian MPM would require a high resolution Lagrangian mesh for bunnies in order to resolve collisions between phases. Furthermore, traditional MPM methods often have difficulties recovering from element

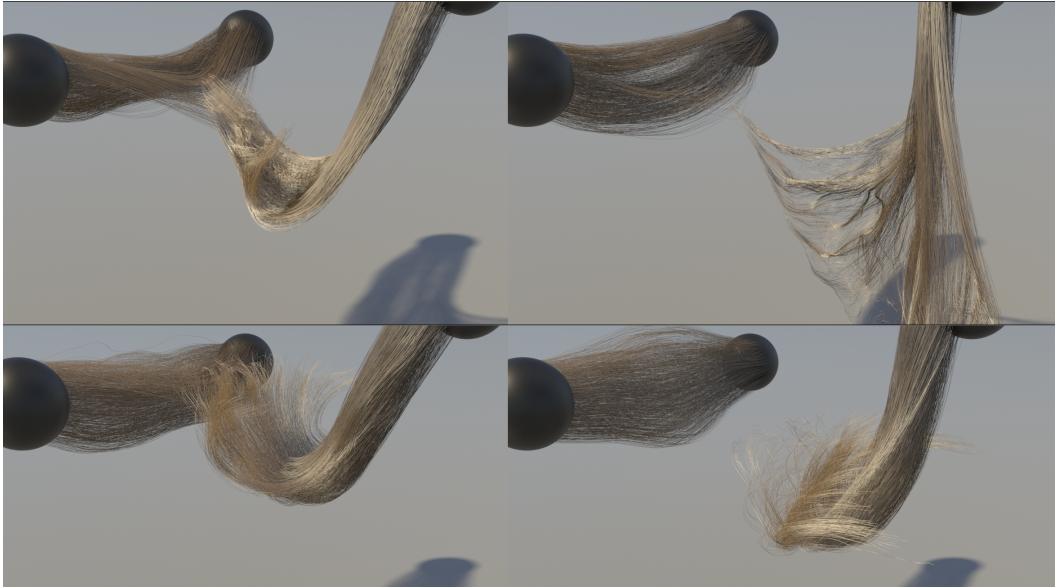


Fig. 15. Hair tubes comparison. Comparison between McAdams et al. [2009] (top row) and our method (bottom row) in resolving the collisions between two bundles of hair strands.

inversions, as the particle modes needed to uninvert the material are lost in the transfers between particles and the grid due to the **type (ii)** interactions discussed in Section 4. On the other hand, our method handles extreme deformation and even element inversion as demonstrated in Figure 10. MPM fails to recover the original shape of the object when the grid resolution is low and **type (ii)** interactions are effective and exhibits high frequency noise when the grid resolution is too high for **type (ii)** interactions to be effective. On the other hand, the elastic object recovers its original shape with any grid resolutions using our method.

8.3 Coupling with MPM and rigid bodies

Our method also supports coupling with rigid bodies as well as traditional MPM particles such as snow, sand and clothing. In Figure 13, we demonstrate the coupling of soft tissues with clothing material simulated with MPM as in [Jiang et al. 2017]. In Figure 2, we show a hairy ball that is first hit by a snowball and then shakes the snow off. In Figure 1 (right), elastic characters and a column of sand are poured on a series of pinwheels simulated as rigid bodies, setting them in motion. In Figure 4, colored sand is poured on top of three Jell-O's with various stiffness, generating interesting patterns.

9 DISCUSSION AND LIMITATIONS

While our approaches address many shortcomings in existing techniques, there are a number of limitations that persist. First, while abandoning the transversely isotropic elastoplasticity assumption of Jiang et al. [2017] does improve the resolution of more complicated strand interactions, as shown in Figure 11, it also causes the potential energy associated with collision and shearing to interfere with that of the strand. Interestingly, this does not have an effect under extension. Only under compression of a strand will there be an additional resistance. Furthermore, while our treatment of rigid body dynamics is useful for coupling with elastoplastic materials like sand, soft tissues, etc., our approach is not ideally suited for interactions between rigid bodies. Our approach

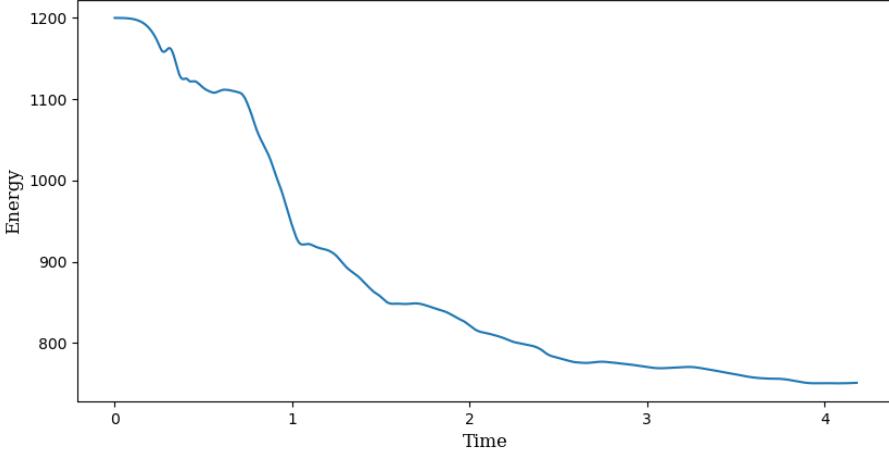


Fig. 16. **Energy Plot.** We plot the total energy as a function of time for the hair tubes test. The energy is calculated as the sum of elastic and gravitational potential energy and the kinetic energy on the particles.

	Time	Element #	Particle #	Δx	CFL
Mannequin (Fig. 14 left)	39	933K	41K/41K	0.05	0.6
Mannequin (Fig. 14 right)	27	641K	31K/31K	0.05	0.6
Pinwheel (Fig. 1 right)	89	93K	930K/57K	0.5	0.6
Bunnies (MPM) (Fig. 9 left)	186	3.97M	2.67M	0.1	0.6
Bunnies (Hybrid) (Fig. 9 right)	66	201K	1.99M/25K	0.1	0.6
Hair ball (MPM) (Fig. 11 top left)	84	1.92M	N/A	0.05	0.1
Hair ball (Hybrid) (Fig. 11 top right)	122/83	1.92M	N/A	0.05	0.1
Hair tubes ([2009]) (Fig. 15 top)	156/56	47.5K	N/A	0.08	0.6
Hair tubes (Hybrid) (Fig. 15 bottom)	55/11	47.5K	N/A	0.08	0.6
Skin and shirt (Fig. 13)	3	207K	120K/40K	0.006	0.6
Braiding (Fig. 12)	87/73	372K	N/A	0.15	0.2
Braids (Fig. 3)	25/9	323K	N/A	0.03	0.2
Hair (Fig. 5)	127/46	1.01M	N/A	0.05	0.6
Snow on hair (Fig. 2)	153/38	1.92M	2.16M	0.05	0.2
Wall breaking (Fig. 1 left)	50	933K	2.29M/41K	0.05	0.6
Dancer (Fig. 1 middle)	117/27	490K	N/A	0.04	0.2

Table 1. All simulations were run on an Intel Xeon E5-2690 V2 system with 20 threads and 128GB of RAM. Simulation time is measure in seconds per frame. Time spent on geometric collision per frame is recorded in the second entry of the timing column where applicable. Element # denotes number of segments for hair simulations and number of tetrahedra for volumetric simulations. Particle # denotes the total number of MPM particles, and the number of collision particles are recorded in the second entry where applicable.

fails to resolve simple cases like stacking of a few rigid bodies without penetration and/or grid based separation artifacts. Lastly, our collision impulses do not provide any geometric guarantees against self collision, as in e.g. [Bridson et al. 2002]. If large time steps are taken, material will interpenetrate. In general this can be avoided by obeying a CFL condition, as is generally true with MPM.

ACKNOWLEDGMENTS

The work is supported by NSF CCF-1422795, ONR (N000141110719, N000141210834), DOD (W81XWH-15-1-0147), Intel STC-Visual Computing Grant (20112360) as well as a gift from Adobe Inc.

REFERENCES

- T. Belytschko, W. Liu, B. Moran, and K. Elkhodary. 2013. *Nonlinear finite elements for continua and structures*. John Wiley and sons.
- Miklós Bergou, Basile Audoly, Etienne Vouga, Max Wardetzky, and Eitan Grinspun. 2010. Discrete viscous threads. In *ACM Transactions on Graphics (TOG)*, Vol. 29. ACM, 116.
- Miklós Bergou, Max Wardetzky, Stephen Robinson, Basile Audoly, and Eitan Grinspun. 2008. Discrete elastic rods. *ACM transactions on graphics (TOG)* 27, 3 (2008), 63.
- F. Bertails, B. Audoly, M. Cani, B. Querleux, F. Leroy, and J. Lévéque. 2006. Super-helices for Predicting the Dynamics of Natural Hair. *ACM Trans Graph* 25, 3 (2006), 1180–1187.
- J. Bonet and R. Wood. 2008. *Nonlinear continuum mechanics for finite element analysis*. Cambridge University Press.
- J. Brackbill and H. Ruppel. 1986. FLIP: A method for adaptively zoned, Particle-In-Cell calculations of fluid flows in two dimensions. *J Comp Phys* 65 (1986), 314–343.
- R. Bridson, R. Fedkiw, and J. Anderson. 2002. Robust Treatment of Collisions, Contact and Friction for Cloth Animation. *ACM Trans Graph* 21, 3 (2002), 594–603.
- G. Daviet and F. Bertails-Descoubes. 2016. A Semi-implicit Material Point Method for the Continuum Simulation of Granular Materials. *ACM Trans Graph* 35, 4 (2016), 102:1–102:13.
- Y. Fan, J. Litven, D. Levin, and D. Pai. 2013. Eulerian-on-lagrangian Simulation. *ACM Trans Graph* 32, 3 (2013), 22:1–22:9.
- Y. Fan, J. Litven, and D. Pai. 2014. Active Volumetric Musculoskeletal Systems. *ACM Trans Graph* 33, 4 (2014), 152:1–152:9.
- Y. Fang, Y. Hu, S. Hu, and C. Jiang. 2018. A temporally adaptive material point method with regional time stepping. In *Computer Graph Forum*, Vol. 37. Wiley Online Library, 195–204.
- Y. Fei, C. Batty, E. Grinspun, and C. Zheng. 2018. A multi-scale model for simulating liquid-fabric interactions. *ACM Trans Graph* 37, 4 (2018), 51:1–51:16. <https://doi.org/10.1145/3197517.3201392>
- C. Fu, Q. Guo, T. Gast, C. Jiang, and J. Teran. 2017. A Polynomial Particle-in-cell Method. *ACM Trans Graph* 36, 6 (Nov. 2017), 222:1–222:12.
- M. Gao, A. Pradhana, X. Han, Q. Guo, G. Kot, E. Sifakis, and C. Jiang. 2018a. Animating fluid sediment mixture in particle-laden flows. *ACM Trans Graph* 37, 4 (2018), 149:1–149:11. <https://doi.org/10.1145/3197517.3201309>
- M. Gao, A. Tampubolon, C. Jiang, and E. Sifakis. 2017. An adaptive generalized interpolation material point method for simulating elastoplastic materials. *ACM Trans Graph* 36, 6 (2017), 223:1–223:12. <https://doi.org/10.1145/3130800.3130879>
- M. Gao, X. Wang, Kui KWu, A. Pradhana, E. Sifakis, C. Yuksel, and C. Jiang. 2018b. GPU optimization of material point methods. In *SIGGRAPH Asia 2018 Technical Papers (SIGGRAPH Asia '18)*. ACM, New York, NY, USA, Article 254, 12 pages. <https://doi.org/10.1145/3272127.3275044>
- J. Gaume, T. Gast, J. Teran, A. van Herwijnen, and C. Jiang. 2018. Dynamic anticrack propagation in snow. *Nature communications* 9, 1 (2018), 3047.
- A. Golas, R. Narain, and M. Lin. 2014. Continuum modeling of crowd turbulence. *Phys Rev E* 90 (2014), 042816. Issue 4.
- J. E. Guilkey and J. A. Weiss. 2003. Implicit time integration for the material point method: Quantitative and algorithmic comparisons with the finite element method. *Int J Numer Meth Eng* 57, 9 (2003), 1323–1338.
- Q. Guo, X. Han, C. Fu, T. Gast, R. Tamstorf, and J. Teran. 2018. A material point method for thin shells with frictional contact. *ACM Trans Graph* 37, 4 (2018), 147. <https://doi.org/10.1145/3197517.3201346>
- C. Hammerquist and J. Nairn. 2017. A new method for material point method particle updates that reduces noise and enhances stability. *Comp Meth App Mech Eng* 318 (2017), 724 – 738.
- F. Harlow. 1964. The particle-in-cell method for numerical solution of problems in fluid dynamics. *Meth Comp Phys* 3 (1964), 319–343.
- Jan Hegemann, Chenfanfu Jiang, Craig Schroeder, and Joseph M. Teran. 2013. A Level Set Method for Ductile Fracture. In *Proc ACM SIGGRAPH/Eurograp Symp Comp Anim*. 193–201.
- Y. Hu, Y. Fang, Z. Ge, Z. Qu, Y. Zhu, A. Pradhana, and C. Jiang. 2018. A moving least squares material point method with displacement discontinuity and two-way rigid body coupling. *ACM Trans Graph* 37, 4 (2018), 150:1–150:14. <https://doi.org/10.1145/3197517.3201293>
- P. Huang, X. Zhang, S. Ma, and X. Huang. 2011. Contact algorithms for the material point method in impact and penetration simulation. *Int J Num Meth Eng* 85, 4 (2011), 498–517. <https://doi.org/10.1002/nme.2981>
- G. Irving, J. Teran, and R. Fedkiw. 2004. Invertible Finite Elements for Robust Simulation of Large Deformation. In *Proc ACM SIGGRAPH/Eurograph Symp Comp Anim*. 131–140.
- C. Jiang, T. Gast, and J. Teran. 2017. Anisotropic elastoplasticity for cloth, knit and hair frictional contact. *ACM Trans Graph* 36, 4 (2017), 152.
- C. Jiang, C. Schroeder, A. Selle, J. Teran, and A. Stomakhin. 2015. The Affine Particle-In-Cell Method. *ACM Trans Graph* 34, 4 (2015), 51:1–51:10.
- Chenfanfu Jiang, Craig Schroeder, Joseph Teran, Alexey Stomakhin, and Andrew Selle. 2016. The Material Point Method for Simulating Continuum Materials. In *ACM SIGGRAPH 2016 Course*. 24:1–24:52.

- G. Klár, T. Gast, A. Pradhana, C. Fu, C. Schroeder, C. Jiang, and J. Teran. 2016. Drucker-prager Elastoplasticity for Sand Animation. *ACM Trans Graph* 35, 4 (2016), 103:1–103:12.
- D. Levin, J. Litwen, G. Jones, S. Sueda, and D. Pai. 2011. Eulerian Solid Simulation with Contact. *ACM Trans Graph* 30, 4 (2011), 36:1–36:10.
- D. Li, S. Sueda, D. Neog, and D. Pai. 2013. Thin Skin Elastodynamics. *ACM Trans Graph* 32, 4 (2013), 49:1–49:10.
- A. McAdams, A. Selle, K. Ward, E. Sifakis, and J. Teran. 2009. Detail Preserving Continuum Simulation of Straight Hair. *ACM Trans Graph* 28, 3 (2009), 62:1–62:6.
- M. Müller, N. Chentanez, T. Kim, and M. Macklin. 2015. Air Meshes for Robust Collision Handling. *ACM Trans. Graph.* 34, 4 (2015), 133:1–133:9.
- R. Narain, A. Golas, S. Curtis, and M. Lin. 2009. Aggregate Dynamics for Dense Crowd Simulation. *ACM Trans Graph* 28, 5 (2009), 122:1–122:8.
- D. Ram, T. Gast, C. Jiang, C. Schroeder, A. Stomakhin, J. Teran, and P. Kavehpour. 2015. A material point method for viscoelastic fluids, foams and sponges. In *Proc ACM SIGGRAPH/Eurograph Symp Comp Anim.* 157–163.
- E. Sifakis and J. Barbic. 2012. FEM simulation of 3D deformable solids: a practitioner’s guide to theory, discretization and model reduction. In *ACM SIGGRAPH 2012 Courses (SIGGRAPH ’12)*. ACM, New York, NY, USA, 20:1–20:50. <https://doi.org/10.1145/2343483.2343501>
- E. Sifakis, S. Marino, and J. Teran. 2008. Globally Coupled Collision Handling Using Volume Preserving Impulses. In *Proc 2008 ACM SIGGRAPH/Eurographics Symp Comp Anim.* 147–153.
- E. Sifakis, T. Shinar, G. Irving, and R. Fedkiw. 2007. Hybrid simulation of deformable solids. In *Proc ACM SIGGRAPH/Eurograph Symp Comp Anim.* Eurographics Association, 81–90.
- A. Stomakhin, R. Howes, C. Schroeder, and J. Teran. 2012. Energetically consistent invertible elasticity. In *Proc Symp Comp Anim.* 25–32.
- A. Stomakhin, C. Schroeder, L. Chai, J. Teran, and A. Selle. 2013. A Material Point Method for snow simulation. *ACM Trans Graph* 32, 4 (2013), 102:1–102:10.
- A. Stomakhin, C. Schroeder, C. Jiang, L. Chai, J. Teran, and A. Selle. 2014. Augmented MPM for phase-change and varied materials. *ACM Trans Graph* 33, 4 (2014), 138:1–138:11.
- D. Sulsky, Z. Chen, and H. Schreyer. 1994. A particle method for history-dependent materials. *Comp Meth App Mech Eng* 118, 1 (1994), 179–196.
- A. P. Tampubolon, T. Gast, G. Klár, C. Fu, J. Teran, C. Jiang, and K. Museth. 2017. Multi-species simulation of porous sand and water mixtures. *ACM Trans Graph* 36, 4 (2017).
- Y. Teng, D. Levin, and T. Kim. 2016. Eulerian Solid-fluid Coupling. *ACM Trans Graph* 35, 6 (2016), 200:1–200:8.
- K. Wu and C. Yuksel. 2016. Real-time Hair Mesh Simulation. In *ACM SIGGRAPH Symp Int 3D Graph Games*. ACM.
- Y. Yue, B. Smith, C. Batty, C. Zheng, and E. Grinspun. 2015. Continuum foam: a material point method for shear-dependent flows. *ACM Trans Graph* 34, 5 (2015), 160:1–160:20.
- Y. Yue, B. Smith, P. Chen, M. Chantharayukhonthorn, K. Kamrin, and E. Grinspun. 2018. Hybrid grains: adaptive coupling of discrete and continuum simulations of granular media. *ACM Trans Graph* 37, 6 (2018), 283:1–283:19.
- F. Zhu, J. Zhao, S. Li, Y. Tang, and G. Wang. 2017. Dynamically enriched MPM for invertible elasticity. In *Comp Graph Forum*, Vol. 36. Wiley Online Library, 381–392.