DOI: 10.1111/cgf.13914
EUROGRAPHICS 2020 / U. Assarsson and D. Panozzo
(Guest Editors)
Volume 39 (2020), Number 2

A Practical Method for Animating Anisotropic Elastoplastic Materials

Camille Schreck[1] and Chris Wojtan[1]

[1]IST Austria

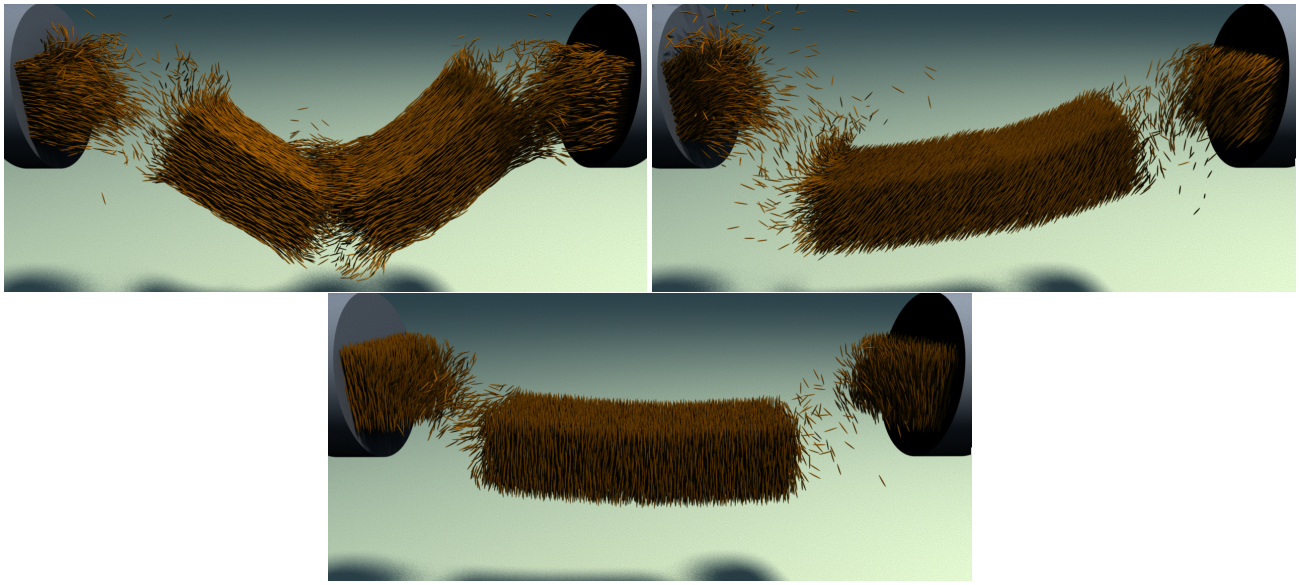


Figure 1: *Pulling apart a bundle of fibrous material. Different fiber orientations produce different behaviors –horizontal (left), tilted (right) and vertical (bottom).*

Abstract

This paper introduces a simple method for simulating highly anisotropic elastoplastic material behaviors like the dissolution of fibrous phenomena (splintering wood, shredding bales of hay) and materials composed of large numbers of irregularly-shaped bodies (piles of twigs, pencils, or cards). We introduce a simple transformation of the anisotropic problem into an equivalent isotropic one, and we solve this new "fictitious" isotropic problem using an existing simulator based on the material point method. Our approach results in minimal changes to existing simulators, and it allows us to re-use popular isotropic plasticity models like the Drucker-Prager yield criterion instead of inventing new anisotropic plasticity models for every phenomenon we wish to simulate.

CCS Concepts

• ***Computing methodologies*** → ***Simulation by animation; Physical simulation;***

1. Introduction

Computer animation greatly benefits from the ability to simulate a wide range of different phenomena. The research literature is rich with techniques for simulating *elastic* deformable materials, which exhibit large deformations but do not permanently deform, and isotropic *elastoplastic* materials which can flow plastically but assume the same behavior in all directions.

However, materials with extremely anisotropic elastoplastic be-

haviors are still difficult to simulate, for practical as well as technical reasons. Without a general method for simulating this class of materials, artists are ill-equipped when animating the dissolution of fibrous phenomena, like splintering wood and shredding bales of hay. They similarly have sub-optimal tools for animating materials composed of large numbers of irregularly-shaped bodies, like vaults full of money, or piles of twigs, pencils, or cards.

In this paper, we introduce a simple method for simulating such materials by transforming the anisotropic problem into an equivalent one that can be solved with an isotropic material solver. This means that we can simulate complicated new material models with existing pipelines, and we illustrate the effectiveness of our technique by integrating it into a standard isotropic solver based on the material point method (MPM). This anisotropic mapping also allows us to re-purpose popular isotropic plasticity models instead of inventing new anisotropic plasticity models for each new phenomenon we wish to simulate. The main contributions of this paper are:

- A generalized model of anisotropy that can be integrated into current MPM elastoplasticity solvers
- An orthotropic law that reduces the generalized model to a few intuitive parameters.

We use these ideas to create animations of diverse phenomena like piles of irregularly-shaped objects and splintering bundles of fibers (like those in Figures 1 and 3). We also propose a practical post-process technique for visualizing our anisotropic MPM simulation (described in Section 7) and increasing its apparent visual detail.

2. Related work

The computer graphics literature on animating deformable bodies is too plentiful to list in detail. The bulk of this section is organized around techniques for simulating various material models, like isotropy/anisotropy and elasticity/elastoplasticity, and we end with a discussion of miscellaneous techniques for simulating related phenomena.

Isotropic Elasticity

Most techniques in computer graphics focus on the specific case of isotropic elastic materials. These techniques take advantage of symmetry (no direction-dependent effects) and they ignore plasticity (no permanent deformations accumulating over time). We refer readers to the course notes of Sifakis & Barbič [SB12] and the recent book by Cai *et. al.* [CLS16] for an overview on isotropic elastic material simulation.

Anisotropic Elasticity

To expand the range of material behaviors, researchers have also explored methods for simulating anisotropic elastic materials. Paper and cardboard [XBP02], biological materials such as brain tissue [VFA06] and teeth [Huo05], and geological [GZY10] and composite materials [OBMO95, COO00] have all been simulated with anisotropic constitutive models. In computer graphics, anisotropic elastic materials are especially useful for approximating biological tissues [PDA03, LST09, MS*15], plants [LB15], and inhomogeneous isotropic materials [KMOD09]. More recently, Kim *et. al.* [KDGI19] proposed the use of anisotropic elasticity to rehabilitate degenerate elements of a mesh.

Isotropic Elastoplasticity

Elastoplastic materials allow some amount of permanent deformation. Early work in computer graphics covered limited plastic behaviors by directly updating the reference frame used for elasticity [NMK*06]. However, these methods become numerically unstable when the reference configuration becomes too distorted. To avoid maintaining a hopelessly entangled reference configuration, researchers introduced Eulerian methods [GBO04], mesh-free methods [MKN*04, CBP05, PKA*05], and methods for continually updating the simulation mesh [BWHT07, WT08, WRK*10].

Subsequently, Sulsky's Material Point Method (MPM) [SZS95] was introduced to computer graphics with the goal of animating snow [SSC*13]. MPM elegantly handles elastoplastic flows by combining Lagrangian particles with Eulerian grids for maintaining a stable simulation. Researchers in physics and material science have since used MPM for simulating soft tissue [IGB*], landslides [AA10], and granular materials [MAMHM15]. Computer graphics researchers have adapted the method for simulating foam [YSB*15, RGJ*15], sand [KGP*16, DBD16], and interactions between liquid and granular material [TGK*17, GPH*18], rigid bodies [HFG*18], or fabrics [FBGZ18]. Recent research has improved MPM by expanding its ability to preserve affine [JSS*15] and higher order information [FGG*17], and introducing spatial adaptivity [GTJS17], and temporal adaptivity [FHHJ18].

Anisotropic Elastoplasticity

As yet, the computer graphics community has not thoroughly explored the simulation of materials that are both elastoplastic and anisotropic. Given the utility of anisotropic elastic material models for animating fibrous material, wood, plants, muscles, and skin, adding plasticity to the models should be useful for allowing ductility, tearing, and fracture effects to these animations.

The work of Jiang *et. al.* [JGT17] addressed some of these effects, by modeling frictional contact as an anisotropic elastoplastic material. Their method introduces additional Lagrangian fibers or sheet structures to define the anisotropic frame, and it simulates anisotropic elastoplasticity based on these frames. Their method is perhaps ideal for their stated result of simulating frictional contact, because these helper structures naturally map to hairs and fabrics. Furthermore, the choice to define anisotropic frames based on the fiber/sheet structures elegantly avoids the complications involved with tracking an evolving anisotropic frame through a general material. The method produces impressive results and lays the groundwork for more general anisotropic elastoplastic behaviors.

However, the method of Jiang *et. al.* [JGT17] requires anisotropic elastoplastic behaviors with an obvious coherent fibrous or sheet structure; it cannot be used for animating volumes

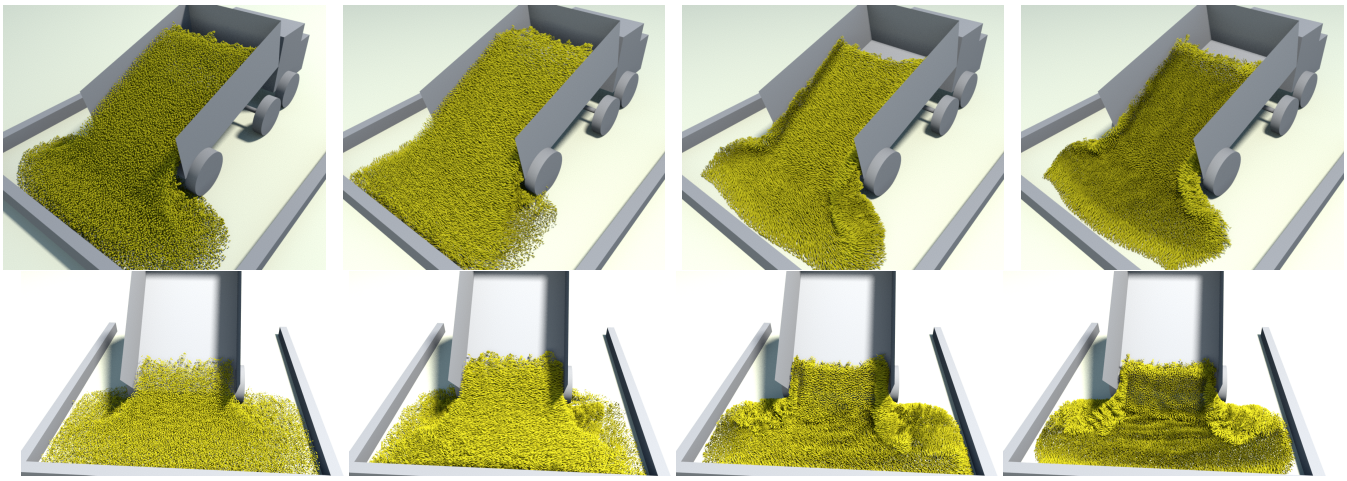


Figure 2: *A simulation of a dump truck unloading a bed full of various types of granular materials. The leftmost images shows an isotropic (sand-like) material, while the others show materials with preferred orientations –horizontal (middle-left), tilted (middle-right) and vertical (rightmost).*

composed of anisotropic granules, and it cannot be used for simulating fracture without explicitly fracturing the fiber/sheet structures as well. Our work aims to solve the more general problem of anisotropic elastoplastic materials without any helper structures like Lagrangian fibers or sheets.

Daviet & Bertails-Descoubes [DBD16] made a breakthrough in anisotropic MPM simulation by specially deriving a material law for anisotropic granular materials. They also introduce a method for modifying anisotropy over time, which we build upon in Section 6.4. In contrast, we want to approximate a wider set of elastoplastic material behavior: instead of proposing a particular plasticity model for a particular phenomenon, our method serves as a simple anisotropic extension to current and future isotropic simulation codes, with little implementation overhead.

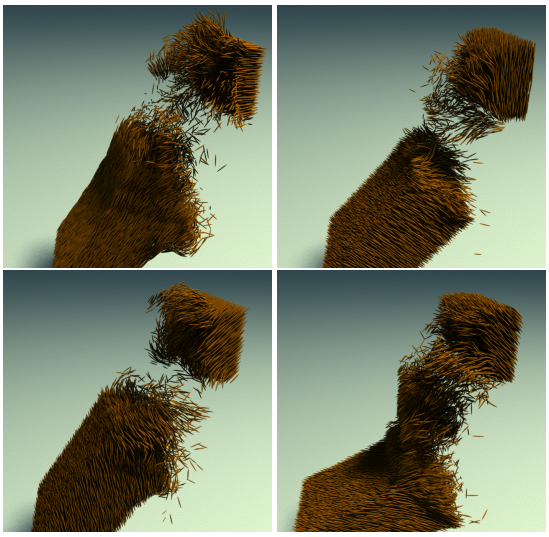


Figure 3: *This fibrous material tears apart under its own weight. Different fiber orientations produce different behaviors.*

Miscellaneous Techniques

Narain *et. al.* [NGL10] introduced a unilateral incompressibility model for approximating flowing granular materials, which led to the more specific granular plasticity models discussed above. Alduán *et. al.* [ATO09] showed how to increase the effective visual resolution of granular media by adding additional high resolution particles, similar to the visualization technique we propose.

Bell *et. al.* [BYM05] animated granular materials using discrete elements, while Hsu and Keyser [HK10] substituted piles of interacting objects with static "sleeping" bodies, in order to significantly speed up simulation time. Follow-up work by Han *et. al.* [HHMK13] performed perceptual studies to determine which types of behaviors are physically plausible when simulating large piles of objects. We argue that, while these methods obviously produce realistic results, approaches based on continuum mechanics are ultimately more efficient than simulating every grain individually. Some middle ground between these two approaches may be found. For example Yue *et. al.* [YSC*18] couple discrete particle simulation and MPM to obtain simulations of granular material with both detailed dynamic behavior and computational efficiency.

Finally, anisotropy is also important for mesh generation. Similar to our approach for simulating anisotropic materials by mapping to a fictitious isotropic space, geometric warping methods create meshes by combining traditionally isotropic meshing techniques with non-Euclidean metrics [PPTSH14, ZGW*13, KMZ10].

3. Continuum Mechanics Background

3.1. Elasticity

We model a deformation as a function $\Phi : \{X,t\} \mapsto x$ that maps the position in material space X and the current time t to the deformed

position x in the global reference frame. The deformation gradient matrix $\boldsymbol{F}$ represents the deformation of an infinitesimal vector dX: $dx = \frac{\partial \Phi}{\partial X} dX = \boldsymbol{F} dX$, the strain $\boldsymbol{E} = \frac{1}{2}(\boldsymbol{F}^T\boldsymbol{F} - \boldsymbol{I})$ measures the deformation as the relative displacement within the deformed body, and the stress $\boldsymbol{\sigma}$ measures the internal forces experienced by the body.

The stress is generally a non-linear function of the strain, $\boldsymbol{\sigma}(\boldsymbol{E})$. To enable large deformations regardless of the stress model, corotational methods map the problem to a local with a rotation matrix $\boldsymbol{R}$, compute the stress $\widetilde{\boldsymbol{\sigma}}$ in the local frame, and then rotate it back: $\boldsymbol{\sigma} = \boldsymbol{R}^T \widetilde{\boldsymbol{\sigma}} \boldsymbol{R}$.

Stress is often linearized with $\widetilde{\boldsymbol{\sigma}} \approx \widetilde{\mathcal{C}} : \boldsymbol{E}$, a tensor contraction between an elasticity tensor $\widetilde{\mathcal{C}}$ and the strain $\boldsymbol{E}$. After factoring out symmetries due to conservation laws, the elasticity tensor $\widetilde{\mathcal{C}}$ has 21 independent degrees of freedom. *Isotropic* materials, which have direction-independent stresses – the stress response would stay the same when the material is rotated–, use these additional symmetries to reduce the 21 degrees of freedom down to only 2: Young's modulus and Poisson's ratio. Due to these few degrees of freedom, isotropic elasticity models are easier to implement and tune, and there are far more isotropic elasticity models in the animation literature than anisotropic ones.

Anisotropic models of elasticity depend not only on the strain, but also on the alignment with the local reference frame, which can vary depending on its position in the material. Anisotropic stress models change the material's elastic response depending on how the deformation aligns with the local reference frame. Anisotropic elasticity models tend to have more parameters and are more cumbersome to implement than isotropic ones.

3.2. Plasticity

State-of-the-art plasticity models in computer graphics consider a *plastic yield surface*, which we encode as the level set of a function $\alpha(\boldsymbol{\sigma})$. If the stress lies within this surface (if $\alpha(\boldsymbol{\sigma}) \leq 0$), then the material undergoes elastic behavior. Otherwise, (if $\alpha(\boldsymbol{\sigma}) > 0$) the stress is projected onto the yield surface. This operation effectively discards stresses beyond the yield surface, giving rise to irreversible deformation.

Plasticity is often implemented using a multiplicative model of the deformation gradient:

$$\boldsymbol{F} = \boldsymbol{F}_e \boldsymbol{F}_{pl} \tag{1}$$

where the deformation gradient $\boldsymbol{F}$ has an elastic part $\boldsymbol{F}_e$ and a plastic part $\boldsymbol{F}_{pl}$. In practice, $\boldsymbol{F}_{pl}$ is computed with a projection onto the yield surface, α [SH06].

Many plastic yield criteria have been proposed in computer graphics: Stomakhin *et. al.* [SSC*13] introduced a yield surface based on the magnitude of the singular values of $\boldsymbol{F}$ to model snow. Bargteil *et. al.* [BWHT07] propose a creeping yield surface that can also exhibit work hardening, for animating goopy materials. Klar *et. al.* [KGP*16] achieve sand behavior by using the Drucker-Prager cone as their yield surface. Clearly, the yield surface plays an essential role in the material's behavior.

All of these plasticity models rely on an *isotropic* yield surface that ignores the orientation of $\boldsymbol{F}$ or $\boldsymbol{\sigma}$ relative to the local frame. Consequently, these methods can be optimized to only use the three singular values as input parameters, instead of using the full tensor. These three-dimensional yield surfaces have simple geometries that are simple to parameterize and understand, like rectangular solids [SSC*13] and cones [KGP*16].

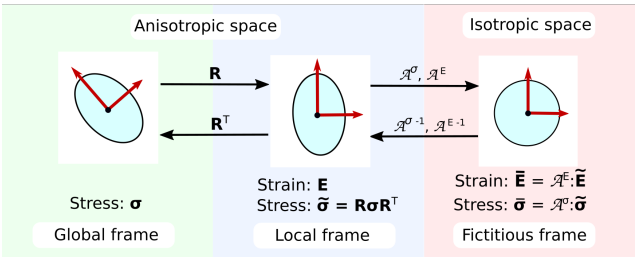


Figure 4: *Our method considers three reference frames. The global frame and local frame consider anisitropic material laws, while fictitious frame warps space in such a way that the material laws become isotropic.*

Anisotropic plasticity models require a yield surface that depends both on stress and the local reference frame. If we want to make all of the tried-and-true isotropic plasticity models anisotropic, we must modify them to take into account the alignment between the deformation and the local frame. Unfortunately, this added dependence on orientation may nullify the elegance and simplicity of many isotropic plasticity models. The yield surfaces are necessarily higher dimensional, and they encode more complicated geometries than the simple isotropic models.

4. Anisotropic mapping

Our work directly addresses these problems with designing and simulating anisotropic elastoplastic materials by introducing the concept of a fictitious *isotropic* space [OBMO95, COO00]. The fictitious space is specifically designed to remove any directional bias caused by anisotropy, making the stress in that space isotropic. By mapping our anisotropic problems into this new space, we are able to model complex anisotropic behaviors by re-using purely isotropic algorithms.

So far we have discussed quantities in the global reference frame ($\boldsymbol{\sigma}$) and in the local frame ($\widetilde{\boldsymbol{\sigma}}$ and $\widetilde{\boldsymbol{E}}$), and we must now introduce quantities in the fictitious isotropic frame ($\overline{\boldsymbol{\sigma}}$ and $\overline{\boldsymbol{E}}$). We use two linear operators $\mathcal{A}^\sigma$ and $\mathcal{A}^E$ to map the stress and strain from the local frame to this isotropic frame. We represent these maps as fourth-order tensors:

$$\overline{\boldsymbol{\sigma}} = \mathcal{A}^\sigma : \widetilde{\boldsymbol{\sigma}} \qquad \text{and} \qquad \overline{\boldsymbol{E}} = \mathcal{A}^E : \widetilde{\boldsymbol{E}}. \tag{2}$$

Please see Figure 4 for an illustration of these three reference frames. We see how to create these maps in Section 6.

We can now simulate anisotropic materials by computing $\overline{\boldsymbol{E}}$ and $\overline{\boldsymbol{\sigma}}(\overline{\boldsymbol{E}})$ using an existing isotropic solver, then mapping them back out of the isotropic frame with $(\mathcal{A}^E)^{-1}$ and $(\mathcal{A}^\sigma)^{-1}$ (see Figure 4). Putting this all together, we convert from the global frame to the

local frame via rotations ($\widetilde{\boldsymbol{\sigma}} = \boldsymbol{R}\boldsymbol{\sigma}\boldsymbol{R}^T$), and we convert from the local frame to the fictitious frame via our mapping ($\overline{\boldsymbol{\sigma}} = \mathcal{A}^{\sigma} : \widetilde{\boldsymbol{\sigma}}$ and $\overline{\boldsymbol{E}} = \mathcal{A}^{E} : \widetilde{\boldsymbol{E}}$). Composition gives us a direct mapping from the global frame to the fictitious isotropic frame ($\overline{\boldsymbol{\sigma}} = \mathcal{A}^{\sigma} : \boldsymbol{R}\boldsymbol{\sigma}\boldsymbol{R}^T$ and $\overline{\boldsymbol{E}} = \mathcal{A}^{E} : \widetilde{\boldsymbol{E}}$). We can bring it back to the global frame again by inverting this process ($\boldsymbol{\sigma} = \boldsymbol{R}^T(\mathcal{A}^{\sigma})^{-1} : \overline{\boldsymbol{\sigma}}\boldsymbol{R}$ and $\widetilde{\boldsymbol{E}} = (\mathcal{A}^{E})^{-1} : \overline{\boldsymbol{E}}$).

We note that the idea of using an anisotropic mapping tensor for simulation originated in the engineering field with the work of Oller *et. al.* [OBMO95], which used the technique to simulate fiber-reinforced composite materials. However, their approach (along with their follow-up work [COO00]) is limited to small deformations with limited plasticity, while modern computer animation requires stable behavior with large deformations. Similarly, their limited plasticity prevents flows, like pouring and fracturing. To make the method more robust, our adaptation introduces a local material frame to ensure stability for large deformations, and it integrates with MPM to enable large plastic flows.

5. Integration into MPM

This section illustrates how we integrate this anisotropic mapping idea into an MPM simulator. We first describe how an MPM solver simulates elastoplastic materials in Section 5.1, and then we describe how to modify the method to animate anisotropic materials in Section 5.2.

5.1. MPM for isotropic materials

Figure 6 shows an overview of the MPM pipeline; we encourage readers to peruse the overview by Jiang *et. al.* [JST*16] for more details. We are using our own implementation of MPM, that mostly follows the pipeline described in [KGP*16].

MPM stores simulation information like masses m_p, positions $\boldsymbol{x}_p$, and velocities $\boldsymbol{v}_p$ on Lagragian particles (indexed by the subscript p), but it uses an Eulerian grid (with nodes indexed by subscript i) to compute forces $\boldsymbol{f}_i$. During each time step of an MPM simulation, the simulator executes the numbered steps in Algorithm 1: First it interpolates particle data onto the grid (1). Then it computes forces –based on the stresses computed at step 7 of the previous loop (2), numerically integrates them to get velocities (3), and modifies grid velocities to account for collisions (4). Afterward, it interpolates velocity data back to the particles (5), computes stress (6), and induces plastic behavior by projecting the stresses and strains onto the yield surface and subsequently updates the deformation gradient (7). Finally, it updates the particle positions by numerically integrating velocity (8). Further details can be found in [JST*16].

Naïvely using this basic MPM algorithm for animating anisotropic behavior requires users to derive custom stress laws in step (6) and anisotropic yield surfaces in step (7). We explain how to avoid this problem by re-using the existing isotropic code in the next section.

5.2. MPM with anisotropic maps

We use the ideas from Section 4 to simulate anisotropic behavior with this MPM algorithm. Instead of re-writing code for steps (6) and (7) to create anisotropic behaviors, we preserve these steps exactly and insert anisotropic maps before and after them. The isotropic code for steps (6) and (7) is re-used, and mapping functions are responsible for the anisotropic behavior. Please see Figure 5 for a modified MPM algorithm which incorporates our anisotropic maps.

This new algorithm deviates from isotropic MPM (Figure 6) after step (5). Instead of proceeding directly to the strain/stress computation, we map to the fictitious isotropic space in step (a). Then we re-use the isotropic code for step (6) to compute $\overline{\boldsymbol{\sigma}}(\overline{\boldsymbol{E}})$ based on the mapped strain $\overline{\boldsymbol{E}}$. We then re-use the isotropic plasticity code in step (7) based on the mapped stresses $\overline{\boldsymbol{\sigma}}$ and strains $\overline{\boldsymbol{E}}$. Afterward, we map the isotropic results back to anisotropic space ($\widetilde{E}$ and $\widetilde{\boldsymbol{\sigma}}$) in step (b). The deformation gradient can then be updated from $\widetilde{E}$. Step (c) then updates the maps based on an updated linearization of the strain (Equation 5) and possible modifications to the anisotropy over time (Section 6.4). The entire algorithm is spelled out in Algorithm 1, with the lines specific to our method highlighted in yellow. Note that we make no modifications to the existing MPM algorithm (numbered lines), so our method can be thought of as a plug-in to any MPM solver. To make the modularity of our algorithm more apparent, we will release our source code upon publication of this work.

This approach gives us a couple of benefits over more straightforward approaches to animating anisotropic materials. First, it automatically derives new anisotropic elasticity and plasticity laws from existing isotropic laws. Second, it allows us to re-use exist-

Algorithm 1: Anisotropic MPM
Our modifications are highlighted in yellow

Initialize grid and particles;
Pre-compute particle volumes V_p;
Initialize maps $\mathcal{A}^E_p$ and $\mathcal{A}^{\sigma}_p$;
foreach *time step* **do**
(1) Interpolate m_p and $\boldsymbol{v}_p$ from the particles onto grid nodes;
(2) Compute forces $\boldsymbol{f}_i$ on grid nodes;
(3) Numerically integrate $\boldsymbol{f}_i$ (Forward Euler);
(4) Compute collisions with obstacles on the grid;
(5) Interpolate velocities $\boldsymbol{v}_i$ back to the particles, and compute $\boldsymbol{F}_p$ and $\boldsymbol{E}_p$;
(a) Apply anisotropic map to co-rotated strain $\overline{\boldsymbol{E}}_p := \mathcal{A}^E_p : \widetilde{\boldsymbol{E}}_p$;
(6) Compute stress $\overline{\boldsymbol{\sigma}}_p(\overline{\boldsymbol{E}}_p)$ for each particle;
(7) Project $\overline{\boldsymbol{E}}_p$ and $\overline{\boldsymbol{\sigma}}_p$ onto the plastic yield surface;
(b) Invert maps $\widetilde{\boldsymbol{E}}_p := (\mathcal{A}^E_p)^{-1} : \overline{\boldsymbol{E}}_p$; $\boldsymbol{\sigma}_p := \boldsymbol{R}^T(\mathcal{A}^{\sigma}_p)^{-1} : \overline{\boldsymbol{\sigma}}_p\boldsymbol{R}$;
(c) Update maps $\mathcal{A}^E_p$ and $\mathcal{A}^{\sigma}_p$;
(8) Numerically integrate $\boldsymbol{v}_p$ (Forward Euler);
end

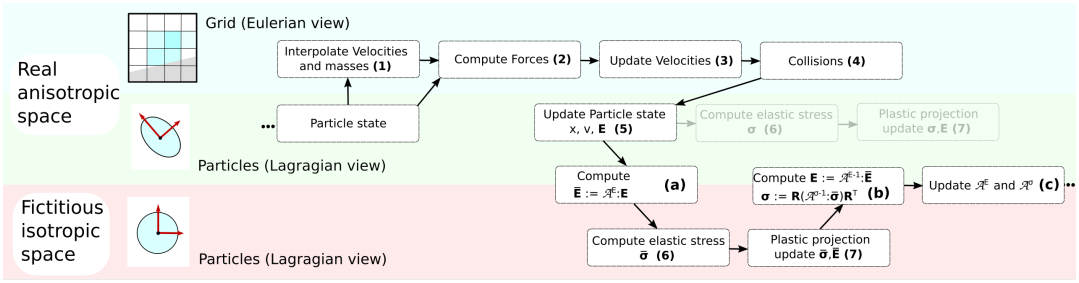


Figure 5: *The MPM algorithm modified to incorporate anisotropy. The only changes to the isotropic MPM pipeline (Figure 6) are steps (a), (b), and (c).*

ing simulation code; the anisotropic mapping idea is easily inserted into existing isotropic simulators, expanding their material range without much additional work.

6. Creation of the maps

Using this method, the anisotropic behavior is entirely defined by the two maps $\mathcal{A}^\sigma$ and $\mathcal{A}^E$. In this section, we discuss the creation of these maps. General anisotropy enables us to create a very large range of behaviors (see for example Figure 7). But the large number of degrees of freedom makes it also rather unintuitive and unpractical to tune. However, most material behaviors present symmetries that reduce the number of degrees of freedom (DOF). In Section 6.2, we propose a simple orthotropic law and show how to create the corresponding map. We thus reduce the number of DOF to only a few, potentially more intuitive, control knobs for navigating this complicated design space.

6.1. General anisotropy

The mapping tensors are constrained to be symmetric to preserve the symmetry of the stress and strain matrices:

$$\mathcal{A}_{ijkl} = \mathcal{A}_{klij}, \quad \mathcal{A}_{ijkl} = \mathcal{A}_{jikl}, \quad \mathcal{A}_{ijkl} = \mathcal{A}_{ijlk}. \tag{3}$$

This means that in the general case they have 36 degrees of freedom. .

$\mathcal{A}^\sigma$ defines how the isometric yield surface is distorted in the anisotropic space. The further $\mathcal{A}^\sigma$ is from the identity tensor $\mathcal{I}$, the more anisotropic the plasticity is going to be. $\mathcal{A}^E$ controls the anisotropy of the elasticity. It can be chosen by tuning it away from $\mathcal{I}$ just like $\mathcal{A}^\sigma$, but it can be also computed from an already-existing anisotropic elasticity law if needed: By linearizing the elastic law for respectively the isotropic and anisotropic laws, we obtain:

$$\overline{\boldsymbol{\sigma}} \approx \overline{\mathcal{C}} : \overline{\boldsymbol{E}} \quad \text{and} \quad \widetilde{\boldsymbol{\sigma}} \approx \widetilde{\mathcal{C}} : \widetilde{\boldsymbol{E}}, \tag{4}$$

where $\overline{\mathcal{C}}$ and $\widetilde{\mathcal{C}}$ are respectively the tangent elasticities of each law.

Plugging the definitions from (2) into this equation allows us to calculate the strain mapping:

$$\mathcal{A}^E = \overline{\mathcal{C}}^{-1} : \mathcal{A}^\sigma : \widetilde{\mathcal{C}}. \tag{5}$$

If an elastic law is described by $\boldsymbol{E} = \boldsymbol{\sigma}(\boldsymbol{E})$ then the tangent elasticity would be the tensor $\mathcal{C} = \frac{\partial \boldsymbol{\sigma}}{\partial \boldsymbol{E}}(\boldsymbol{E})$.

The isometric elastic law can be chosen arbitrarily, however choosing it such that $\mathcal{A}^E$ is close to identity may improve the conditioning of the system.

We show in Figure 7 an example of general anisotropic behavior obtained by playing with the DOF. Nevertheless to make it easier to manipulate the behavior, we propose a formula to obtain an orthotropic law with only a few tuning parameters. This simplified model is what we use in the rest of the article as well as for the other examples.

6.2. Orthotropic anisotropy

An orthotropic material is a material whose mechanic properties have locally three orthogonal planes of symmetry. The material parameters can thus be defined along three orthogonal axes (the normals of the symmetry planes). Orthotropic law are very common in nature. In particular, in the case of granular material, we can make the hypothesis that the grains can be approximated by ellipsoids and that the distribution of their orientation has locally three orthogonal planes of symmetry. These symmetries reduce the number of parameters from 36 to 9. Taking these parameter simplifications even further, we found that, for $\mathcal{A}^\sigma$, only three degrees of freedom, aligned with the axes of the local material frame, were sufficient to achieve several different anisotropic behaviors in our results: we set the stretching along each axis i in the local frame equal to a scale factor a_i, with each a_i can be seen as controlling the plastic fragility along each axis. We capture this behavior by requiring that $\mathcal{A}^\sigma : \boldsymbol{S} = \boldsymbol{DSD}$ for all symmetric matrix $\boldsymbol{S}$, with $\boldsymbol{D} = \mathrm{Diag}(a_x, a_y, a_z)$.

Specifically, we define the stress mapping tensor to be:

$$\mathcal{A}^\sigma_{ijkl} = \begin{cases} a_i^2, & \text{if } i=j=k=l \\ a_i a_j/2, & \text{if } i=k \text{ and } j=l \\ & \text{or } i=l \text{ and } j=k \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

In our implementation we use linear orthotropic elastic materials [LB15]. For these materials, the elasticity tensor in the local frame

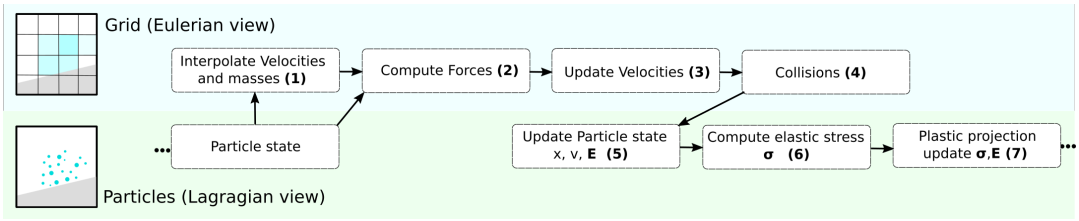


Figure 6: *A schematic representation of the MPM algorithm in Section 5.1. Steps (1)-(4) on the top portion of the figure take place in the Eulerian frame (grid), while steps on the bottom half use the Lagrangian frame (particles).*

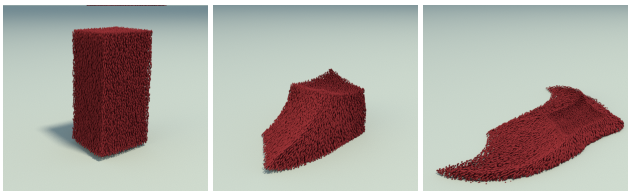


Figure 7: *A collapsing column of vertical particles of non-orthotropic material.*

(expressed in Voigt tensor notation) is defined by its inverse:

$$\widetilde{\mathcal{C}}^{-1} = \begin{pmatrix} \frac{1}{E_1} & -\frac{\nu_{21}}{E_2} & -\frac{\nu_{31}}{E_3} & 0 & 0 & 0 \\ -\frac{\nu_{12}}{E_1} & \frac{1}{E_2} & -\frac{\nu_{32}}{E_3} & 0 & 0 & 0 \\ -\frac{\nu_{13}}{E_1} & -\frac{\nu_{23}}{E_2} & \frac{1}{E_3} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\mu_{23}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{\mu_{31}} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{\mu_{12}} \end{pmatrix} \quad (7)$$

where E_i is the Young's modulus for a uniaxial tension in the direction of axis i, μ_{ij} is the shear modulus in direction j on the plane whose normal is in direction i, and ν_{ij} is the Poisson ratio corresponding to a contraction in direction j for an extension in direction i. Given $\widetilde{\mathcal{C}}$ from this anisotropic law and $\overline{\mathcal{C}}$ from the isotropic elastic law, we can compute $\mathcal{A}^E$ from Equation 5.

We can also easily define a transverse isotropic law (in the x-axis direction) by setting $a_y = a_z$, $E_y = E_z$ and $\mu_{xy} = \mu_{zx}$.

Note that the anisotropic maps $\mathcal{A}^E$ and $\mathcal{A}^\sigma$ may vary over time. For piles of granular materials, the anisotropy should reflect the average orientation of the grains in the continuum, which may change if mixing or alignment occurs. In this case, the maps should be recomputed each step to reflect the changing anisotropy, as we describe in Section 6.4. Also, plasticity may result in a change in the elastic law (hardening or softening), in which case $\mathcal{A}^E$ would also need to be updated.

6.3. Local reference frames in MPM

The global-to-local mapping requires a rotation matrix $\boldsymbol{R}$ for each particle. $\boldsymbol{R}$ is also needed for the visualization of the granular material as it represents the local orientation (or main orientation) of the grains. $\boldsymbol{R}$ can be computed in many different ways like polar [ACOL00] or singular value [Sol15] decomposition of the deformation gradient $\boldsymbol{F}$. We choose an approach based on shape matching [MHTG05]. Our approach begins with an orientation $\boldsymbol{R}_0$ given by the initial conditions, which we update over time using a shape matching algorithm on the MPM grid. Specifically, we create a weighted point distribution S_p^t consisting of particle p and its nearby grid nodes weighted by the MPM kernel w_{ip}. We then approximate the positions for the next time step by advecting each point in S_p^t to create another set S_p^{n+1}. We compute the optimal rotation $\boldsymbol{R}$ about particle p which takes S_p^n to S_p^{n+1} using shape matching [MHTG05]. This approach relies on the interpolation of the advected position of the grid nodes rather than their velocity, making it consistent with the *grid to particles* transfer approach used in our implementation based on [KGP*16]. Although other strategies for estimating $\boldsymbol{R}$ are possible, we found this approach reliable for simulation and useful for visualization post-processes (Section 7).

6.4. Changing Anisotropy over Time

As noted in Section 6.2, anisotropy in the material can change over time. If we approximate a granular material with an anisotropic elastoplastic constitutive model, then the anisotropy depends on the alignment of the grains. Aligned grains will exhibit strong anisotropy along the common direction, while a uniform mixture of grain orientations will exhibit roughly isotropic behavior.

Daviet & Bertails-Descoubes [DBD16] introduced a method for tracking grain orientations by evolving the distribution's second moment matrix $\boldsymbol{\Pi}_2$ [FT84]. They describe a partial differential equation for updating this matrix:

$$\frac{\partial \boldsymbol{\Pi}_2}{\partial t} = \boldsymbol{W}_p \boldsymbol{\Pi}_2 - \boldsymbol{\Pi}_2 \boldsymbol{W}_p + l(\boldsymbol{D}_p \boldsymbol{\Pi}_2 + \boldsymbol{\Pi}_2 \boldsymbol{D}_p - 2\boldsymbol{D}_p : \boldsymbol{\Pi}_4), \quad (8)$$

where l is a coefficient describing the elongation of the grains, $\boldsymbol{\Pi}_4$ is the fourth moment tensor approximated by $\boldsymbol{\Pi}_2 \otimes \boldsymbol{\Pi}_2$, and $\boldsymbol{D}_p$ and $\boldsymbol{W}_p$ are the symmetric and asymmetric parts of the velocity gradient at point p, respectively. We note that this equation models $\boldsymbol{\Pi}_2$'s changes in the *global* Eulerian reference frame, including not only the change in distribution, but also its rotation. However, our local frames $\boldsymbol{R}$ already compute rotations, so we simply extract the rightmost portion of this equation:

$$\frac{\partial \widetilde{\boldsymbol{\Pi}}_2}{\partial t} = l(\widetilde{\boldsymbol{D}}_p \widetilde{\boldsymbol{\Pi}}_2 + \widetilde{\boldsymbol{\Pi}}_2 \widetilde{\boldsymbol{D}}_p - 2\widetilde{\boldsymbol{D}}_p : \widetilde{\boldsymbol{\Pi}}_4), \quad (9)$$

where $\widetilde{\boldsymbol{D}}_p$ and $\widetilde{\boldsymbol{\Pi}}_2$ are defined in the *local* frame.

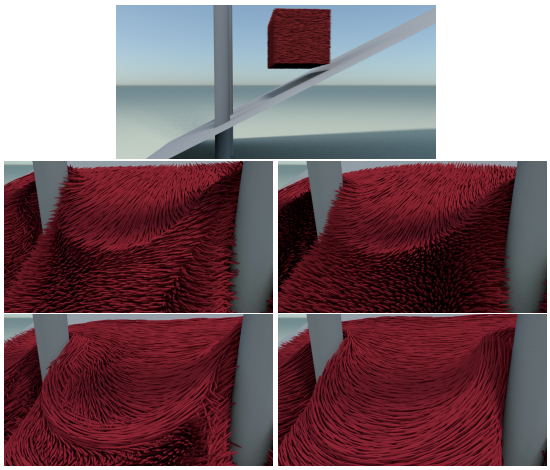


Figure 8: *Granular materials aligning with the flow: a block of particles is dropped on a slope (top). The particles are initially horizontal and oriented in the direction of the slope (middle) or the orthogonal direction (bottom). The simulations on the left update their local frame (Section 6.4), while the right do not.*

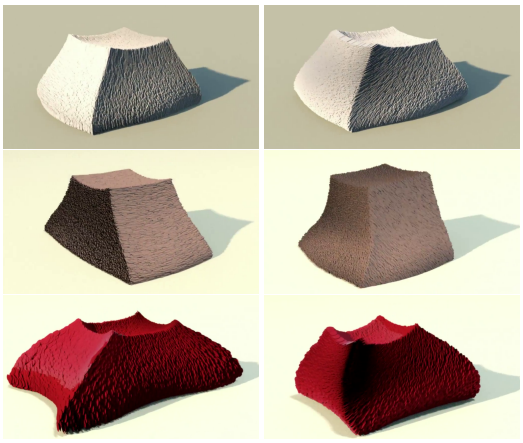
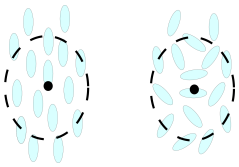


Figure 9: *Varying the shape of the material anisotropy between 2D discs and 1D fibers lets us animate materials consisting of large numbers of oriented cards (top), rods (middle), and a layered material (bottom). The particles are initially horizontal (left) or tilted (right)*



This equation enables us to keep track of the distribution of grain orientations associated with each MPM particle. If the orientations are uniformly distributed (inset right), the three eigenvalues of $\widetilde{\mathbf{\Pi}}_2$ equal $\frac{1}{3}$, and we prescribe isotropic behavior at this position by setting $E_1 = E_2 = E_3$, $\nu_{12} = \nu_{23} = \nu_{13}$, $a_1 = a_2 = a_3$, and recomputing sheer moduli ν from these new parameters. On the other hand, if all the grains are oriented in the same direction (inset left), the eigenvalues of $\widetilde{\mathbf{\Pi}}_2$ are (0,0,1), with the last eigenvector pointing in the same direction as the grains. Here, we prescribe maximally anisotropic behavior, given by $E_\parallel$, $\nu_\parallel$, and $a_\parallel$ in the main direction of orientation and $E_\perp, \nu_\perp, a_\perp$ in the orthogonal direction.

In most cases, $\widetilde{\mathbf{\Pi}}_2$ will describe behavior in between these two extremes, so we use its eigenvalues λ to linearly interpolate between $\{E_{\text{iso}}, \nu_{\text{iso}}, a_{\text{iso}}\}$ and $\{E_\perp, \nu_\perp, a_\perp\}$ when $\lambda < 1/3$, and interpolate between $\{E_{\text{iso}}, \nu_{\text{iso}}, a_{\text{iso}}\}$ and $\{E_\parallel, \nu_\parallel, a_\parallel\}$ when $\lambda \geq 1/3$.

An important note is that the eigenvectors of $\widetilde{\mathbf{\Pi}}_2$ function as the axes of a local material frame, with the largest eigenvector corresponding to the z-axis. Updating $\widetilde{\mathbf{\Pi}}_2$ with Equation 9 can change the eigenvectors and thus change the material frame. We label this change in frame $\Delta\boldsymbol{R}$ and use it to update our $\boldsymbol{R}$ matrix after each re-orientation.

7. Visualization

The materials simulated in our examples consist of large numbers of small oriented fibers, plates, or grains (Figure 9). Since MPM discretizes a continuous material into smaller control volumes, the MPM particles alone are not sampled finely enough to provide a rich visualization of the material. As a post-process after simulation, we choose to populate the material with many "sub-particles" ($\approx 10\times$ more than the number of MPM particles), each visualized as an oriented rigid body. In a post-process, we update the position of each sub-particle by advecting it through the velocity field, and we update its orientation using the shape-matching algorithm described in Section 6.3.

During this post-process, sub-particles may escape the influence of the MPM simulation (when the interpolated density at that point falls below a user-defined threshold of 0.1%-1% of the material density). We treat these escaped sub-particles as ballistic rigid bodies, instead of interpolating their motion from the MPM simulation. This ballistic motion is especially useful for fracture simulations which tend to shatter into small fragments (Figure 1). The velocity can get unstable on the grid nodes with very low density, this may cause spurious velocities for particles switching from MPM to a ballistic motion. To avoid this we set the velocity of a node to zero when its density is below a threshold.

To account for changing anisotropy (as described in Section 6.4), we also update the local frame of each sub-particle. We do this by interpolating the change in frame $\Delta\boldsymbol{R}$ to each sub-particle and applying the rotation adjustment. For a smoother visual effect, we prefer to first interpolate $\Delta\boldsymbol{R}$ from the MPM particles to the grid, and then interpolate from the grid to the sub-particles (rather than interpolating from particles to sub-particles directly).

Note that this subparticles visualization is only a starting point. It works well-enough for large simulations seen from rather far away. But further investigations would be needed for a more complex visualization scheme that would be required for closer range. Indeed at close range, individual interactions between grains –resulting in motions and changes of orientation more complex than MPM is capable to handle– become more visible. Also MPM artifacts –notably floating particles that follow the general motion without seeming in contact with anything– becomes more evident. Solving those visualization problems is a great area for future work.

8. Results

Our supplementary video shows several simulations created with our technique. Our implementation uses an orthotropic material, with anisotropy expressed as an independent scaling along three spatial axes. We approximate fibrous materials or elongated grains (rice, bricks, etc.) by amplifying one axis much more than the other two. We approximate layered materials or flattened grains (coins, cards, etc.) by shrinking one axis much more than the other two. We simulate nearly isotropic materials by using a more uniform scaling. The different types of anisotropy can be seen in Figure 9.

Our method can take any isotropic code and re-use it for anisotropic materials. We illustrate this benefit by using a multitude of different material models throughout our examples. For stress computation, we used isotropic co-rotated linear elasticity, a hyper-elastic model [SSC*13], and a granular elasticity model [KGP*16]. For plasticity, we used the isotropic snow model from Stomakhin *et. al.* [SSC*13], the Drucker-Prager model proposed by Klar *et. al.* [KGP*16], and a shear-clamping biological tissue model from Ionescu *et. al.* [IGB*]. (See Table 2.)

We also modify the orientation of the anisotropy for different visual effects, as shown in Figure 10. One convenient side-effect of MPM is that it approximates fracture behavior when particles become too sparsely sampled. We use this feature to animate fractured fibrous materials in Figures 1 and 3.

Compared to an existing isotropic MPM implementation, our method adds the calculation, application, and update of a two fourth-order tensors per particle, and an additional frame rotation per particle. The performance overhead of our method per time step amounts to 26% on average (See Table 1). The memory overhead is two additional tensors and an additional rotation matrix per particle. However, if we can approximate the stress as linear for the calculation of $\mathcal{A}^E$ –so the tangent elasticity $\mathcal{C}$ is constant through time, and if we do not allow the frames to change via mixing effects, then we only need to store global maps for the entire simulation, instead of storing maps per particle.

Note that multiplying 4th-order matrices is rather costly. If one already knows what constitutive law needs to be used, it may be more efficient to directly implement it. However this method becomes an interesting tool when it comes to experimenting and exploring different anisotropic constitutive laws.

The plastic projection step may induce a non-physical volume gain. Some articles, as [TGK*17], propose solutions to correct this artifact but we did not implement it in our solver yet.

Table 1: *Performance of our method: time in ms for each time step of isotropic MPM (t_{iso}), MPM with our anisotropic extension (t_{ani}), and for handling sub-particles/visualization (t_{viz}). Number of MPM particles (N_{MPM}) and sub-particles (N_{sub})*

Figure	t_{iso}	t_{ani}	t_{viz}	N_{MPM}	N_{sub}
Figure 10	83	97	136	10k	100k
Figure 1	83	102	137	10k	100k
Figure 8	242	325	270	50k	200k
Figure 2	127	164	270	50k	200k

8.1. Discussion and Outlook

Our method is the first technique in computer animation to simulate anisotropic elastoplastic materials by re-using isotropic models. We have shown how the method is capable of animating a wide range of materials by swapping different isotropic elasticity models, different plasticity models, different angles of anisotropy, and different anisotropy dimensions (1D fibers vs 2D discs). Our method also supports spatially varying anisotropy parameters and frame fields.

Although the strategy presented in Section 6.2 is simple and practical for the purpose of computer animation, it makes several assumption which limit the degrees of freedom. Automatically finding correct parameters for general anisotropy –even for orthotropy– is a very complicated task, and out of the scope of this paper. Our primary goal here is to provide artists with a small set of easily tunable control knobs for navigating this complicated design space.

Our current implementation uses forward Euler time integration, which is not ideal for very stiff materials. We would like to investigate implicit integration in the future.

Anisotropic simulators tend to have worse numerical conditioning than isotropic ones, because the ratio between eigenvalues is necessarily exaggerated when we scale the material stiffness along one axis more than another [She02]. However, we do not consider this a limitation of our method, because all methods for simulating anisotropic materials will have this same difficulty.

We have only begun to explore methods for visualizing our anisotropic material simulations. We believe our particle-based visualization is useful for animating fibrous and composite materials, but there is room for improvement. Large piles of objects could benefit from two-way coupling with a rigid body simulator (similar to [YSC*18]), in order to convincingly animate individual grains that escape from the continuum. Also, although standard techniques should work fine for extracting a surface from our simulations, we may wish to investigate techniques with anisotropic bias [YT13].

Our model assumes a local linear map between isotropic space and fictitious isotropic space. We could presumably support non-linear maps by using our algorithm within a Newton solver (which linearizes the non-linear problem each iteration), but we have not explored this yet.

Objects in a pile (particularly objects with a complex shape) can present very complex behavior: making clumping together, interlocking, jamming etc... Representing the material as a continuum may prevent the emergence of some of these phenomena. Efficiently modeling such intricate behaviors is a difficult problem and a very interesting one to look at in the future.

Acknowledgments: We wish to thank the anonymous reviewers and the members of the Visual Computing Group at IST Austria for their valuable feedback. This research was supported by the Scientific Service Units (SSU) of IST Austria through resources provided by Scientific Computing. We would also like to thank Joseph Teran and Chenfanfu Jiang for the helpful discussions.

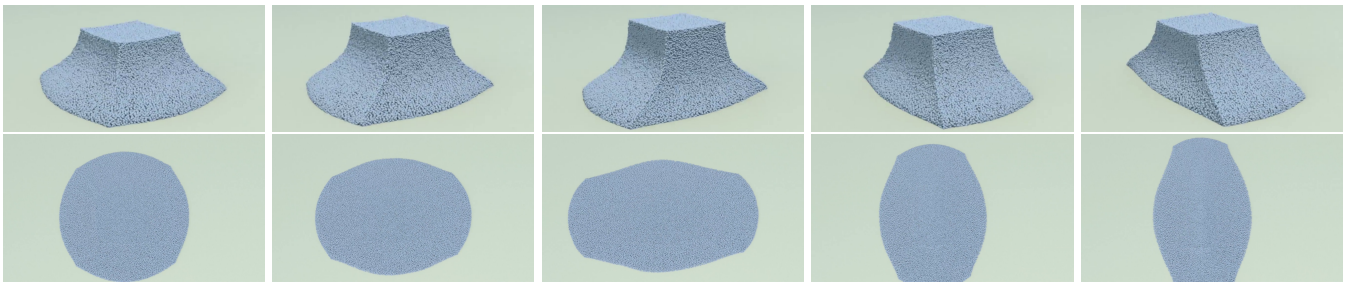


Figure 10: *A column collapsing: varying the direction of anisotropy leads to direction-dependent angles of repose.*

Table 2: *Parameters for the figures in this paper. E_i is the Young's modulus (Pa) for each direction, ν_{ij} the Poisson ratio for each symmetry plane.*

Figure	Elasticity	$E_1/E_2/E_3$	$\nu_{12}/\nu_{23}/\nu_{13}$	Plasticity	Plastic Anisotropy *x*/*y*/*z*	dt (s)
Figure 2	Hyperelastic	5e5/5e5/5e4	0.4/0.4/0.1	[KGP*16]	0.8/0.8/1.4	0.00025
Figure 1	Linear	1e5/1e5/5e4	0.1/0.1/0.4	[IGB*]	0.8/0.8/1.4	0.0002
Figure 8	Hyperelastic	5e5/5e5/5e4	0.4/0.4/0.1	[KGP*16]	0.8/0.8/1.4	0.001
Figure 9 (left)	Hyperelastic	5e4/5e4/1e5	0.2/0.2/0	[KGP*16]	1.1/1.1/0.8	0.001
Figure 9 (center)	Hyperelastic	5e5/5e5/5e5	0.0/0.0/0.4	[KGP*16]	0.8/0.8/1.4	0.001
Figure 9 (right)	Hyperelastic	1e4/1e4/1e4	0.2/0.2/0.1	[SSC*13]	1.1/1.1/0.8	0.00025
Figure 10 (left)	Hyperelastic	1e5/1e5/1e5	0.4/0.4/0.4	[KGP*16]	1/1/1	0.0002
Figure 10 (middle-left)	Hyperelastic	1e5/1e5/1e5	0.4/0.4/0.4	[KGP*16]	1/1/0.75	0.0002
Figure 10 (middle)	Hyperelastic	1e5/1e5/1e5	0.4/0.4/0.4	[KGP*16]	1/1/0.5	0.0002
Figure 10 (middle-right)	Hyperelastic	1e5/1e5/1e5	0.4/0.4/0.4	[KGP*16]	1/1/1.5	0.0002
Figure 10 (right)	Hyperelastic	5e4/5e4/5e4	0.4/0.4/0.4	[KGP*16]	1/1/2	0.0002

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme under grant agreement No. 638176.

References

[AA10] ANDERSEN S., ANDERSEN L.: Modelling of landslides with the material-point method. *Computational Geosciences 14*, 1 (Jan 2010), 137–147. 2

[ACOL00] ALEXA M., COHEN-OR D., LEVIN D.: As-rigid-as-possible shape interpolation. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques* (2000), ACM Press/Addison-Wesley Publishing Co., pp. 157–164. 7

[ATO09] ALDUÁN I., TENA A., OTADUY M. A.: Simulation of high-resolution granular media. In *CEIG* (2009), pp. 11–18. 3

[BWHT07] BARGTEIL A. W., WOJTAN C., HODGINS J. K., TURK G.: A finite element method for animating large viscoplastic flow. In *ACM transactions on graphics (TOG)* (2007), vol. 26, ACM, p. 16. 2, 4

[BYM05] BELL N., YU Y., MUCHA P. J.: Particle-based simulation of granular materials. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2005), ACM, pp. 77–86. 3

[CBP05] CLAVET S., BEAUDOIN P., POULIN P.: Particle-based viscoelastic fluid simulation. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (New York, NY, USA, 2005), SCA '05, ACM, pp. 219–228. 2

[CLS16] CAI J., LIN F., SEAH H. S.: *Graphical Simulation of Deformable Models*. Springer, 2016. 2

[COO00] CAR E., OLLER S., OÑATE E.: An anisotropic elastoplastic constitutive model for large strain analysis of fiber reinforced composite materials. *Computer Methods in Applied Mechanics and Engineering 185*, 2 (2000), 245–277. 2, 4, 5

[DBD16] DAVIET G., BERTAILS-DESCOUBES F.: A semi-implicit material point method for the continuum simulation of granular materials. *ACM Trans. Graph. 35*, 4 (July 2016), 102:1–102:13. 2, 3, 7

[FBGZ18] FEI Y. R., BATTY C., GRINSPUN E., ZHENG C.: A multi-scale model for simulating liquid-fabric interactions. *ACM Trans. Graph. 37*, 4 (July 2018), 51:1–51:16. 2

[FGG*17] FU C., GUO Q., GAST T., JIANG C., TERAN J.: A polynomial particle-in-cell method. *ACM Transactions on Graphics (TOG) 36*, 6 (2017), 222. 2

[FHHJ18] FANG Y., HU Y., HU S.-M., JIANG C.: A temporally adaptive material point method with regional time stepping. *Computer Graphics Forum 37*, 8 (2018), 195–204. 2

[FT84] FOLGAR F., TUCKER, III C. L.: Orientation behavior of fibers in concentrated suspensions. *Journal of Reinforced Plastics and Composites 3*, 2 (1984), 98–119. 7

[GBO04] GOKTEKIN T. G., BARGTEIL A. W., O'BRIEN J. F.: A method for animating viscoelastic fluids. *ACM Trans. Graph. 23*, 3 (Aug. 2004), 463–468. 2

[GPH*18] GAO M., PRADHANA A., HAN X., GUO Q., KOT G., SIFAKIS E., JIANG C.: Animating fluid sediment mixture in particle-laden flows. *ACM Trans. Graph. 37*, 4 (July 2018), 149:1–149:11. 2

[GTJS17] GAO M., TAMPUBOLON A. P., JIANG C., SIFAKIS E.: An adaptive generalized interpolation material point method for simulating elastoplastic materials. *ACM Transactions on Graphics (TOG) 36*, 6 (2017), 223. 2

[GZY10] GAO Z., ZHAO J., YAO Y.: A generalized anisotropic failure criterion for geomaterials. *International Journal of Solids and Structures 47*, 22 (2010), 3166 – 3185. 2

[HFG*18] HU Y., FANG Y., GE Z., QU Z., ZHU Y., PRADHANA A.,

JIANG C.: A moving least squares material point method with displacement discontinuity and two-way rigid body coupling. *ACM Trans. Graph. 37*, 4 (July 2018), 150:1–150:14. 2

[HHMK13] HAN D., HSU S.-W., MCNAMARA A., KEYSER J.: Believability in simplifications of large scale physically based simulation. In *Proceedings of the ACM Symposium on Applied Perception* (New York, NY, USA, 2013), SAP '13, ACM, pp. 99–106. 3

[HK10] HSU S.-W., KEYSER J.: Piles of objects. *ACM Trans. Graph. 29*, 6 (Dec. 2010), 155:1–155:6. 3

[Huo05] HUO B.: An inhomogeneous and anisotropic constitutive model of human dentin. *Journal of Biomechanics 38*, 3 (2005), 587 – 594. 2

[IGB*] IONESCU I., GUILKEY J. E., BERZINS M., KIRBY R. M., WEISS J. A.: Simulation of soft tissue failure using the material point method. *Journal of Biomechanical Engineering 128*, 6. 2, 9, 10

[JGT17] JIANG C., GAST T., TERAN J.: Anisotropic elastoplasticity for cloth, knit and hair frictional contact. *ACM Trans. Graph. 36*, 4 (July 2017), 152:1–152:14. 2

[JSS*15] JIANG C., SCHROEDER C., SELLE A., TERAN J., STOMAKHIN A.: The affine particle-in-cell method. *ACM Transactions on Graphics (TOG) 34*, 4 (2015), 51. 2

[JST*16] JIANG C., SCHROEDER C., TERAN J., STOMAKHIN A., SELLE A.: The material point method for simulating continuum materials. In *ACM SIGGRAPH 2016 Courses* (New York, NY, USA, 2016), SIGGRAPH '16, ACM, pp. 24:1–24:52. 5

[KDGI19] KIM T., DE GOES F., IBEN H.: Anisotropic elasticity for inversion-safety and element rehabilitation. *ACM Trans. Graph. 38*, 4 (July 2019), 69:1–69:15. 2

[KGP*16] KLÁR G., GAST T., PRADHANA A., FU C., SCHROEDER C., JIANG C., TERAN J.: Drucker-prager elastoplasticity for sand animation. *ACM Trans. Graph. 35*, 4 (July 2016). 2, 4, 5, 7, 9, 10

[KMOD09] KHAREVYCH L., MULLEN P., OWHADI H., DESBRUN M.: Numerical coarsening of inhomogeneous elastic materials. *ACM Transactions on Graphics (TOG) 28*, 3 (2009), 51. 2

[KMZ10] KOVACS D., MYLES A., ZORIN D.: Anisotropic quadrangulation. In *Proceedings - 14th ACM Symposium on Solid and Physical Modeling, SPM10* (2010), pp. 137–146. 3

[LB15] LI Y., BARBIČ J.: Stable anisotropic materials. *IEEE Trans. on Visualization and Computer Graphics 21*, 10 (2015), 1129–1137. 2, 6

[LST09] LEE S.-H., SIFAKIS E., TERZOPOULOS D.: Comprehensive biomechanical modeling and simulation of the upper body. *ACM Transactions on Graphics (TOG) 28*, 4 (2009), 99. 2

[MAMHM15] MAST C. M., ARDUINO P., MACKENZIE-HELNWEIN P., MILLER G. R.: Simulating granular column collapse using the material point method. *Acta Geotechnica 10*, 1 (Feb 2015), 101–116. 2

[MHTG05] MÜLLER M., HEIDELBERGER B., TESCHNER M., GROSS M.: Meshless deformations based on shape matching. *ACM Trans. Graph. 24*, 3 (July 2005), 471–478. 7

[MKN*04] MÜLLER M., KEISER R., NEALEN A., PAULY 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* (2004), Eurographics Association, pp. 141–151. 2

[MS*15] MITCHELL N., SIFAKIS E., ET AL.: Gridiron: An interactive authoring and cognitive training foundation for reconstructive plastic surgery procedures. *ACM Transactions on Graphics (TOG) 34*, 4 (2015), 43. 2

[NGL10] NARAIN R., GOLAS A., LIN M. C.: Free-flowing granular materials with two-way solid coupling. *ACM Transactions on Graphics (TOG) 29*, 6 (2010), 173. 3

[NMK*06] NEALEN A., MÜLLER M., KEISER R., BOXERMAN E., CARLSON M.: Physically based deformable models in computer graphics. In *Computer graphics forum* (2006), vol. 25, Wiley Online Library, pp. 809–836. 2

[OBMO95] OLLER S., BOTELLO S., MIQUEL J., OÃŚATE E.: An anisotropic elastoplastic model based on an isotropic formulation. *Engineering Computations 12*, 3 (1995), 245–262. 2, 4, 5

[PDA03] PICINBONO G., DELINGETTE H., AYACHE N.: Non-linear anisotropic elasticity for real-time surgery simulation. *Graphical models 65*, 5 (2003), 305–321. 2

[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

[PPTSH14] PANOZZO D., PUPPO E., TARINI M., SORKINE-HORNUNG O.: Frame fields: Anisotropic and non-orthogonal cross fields. *ACM Trans. Graph. 33*, 4 (July 2014), 134:1–134:11. 3

[RGJ*15] RAM D., GAST T., JIANG C., SCHROEDER C., STOMAKHIN A., TERAN J., KAVEHPOUR P.: A material point method for viscoelastic fluids, foams and sponges. In *Proceedings of the 14th ACM SIGGRAPH / Eurographics Symposium on Computer Animation* (New York, NY, USA, 2015), SCA '15, ACM, pp. 157–163. 2

[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* (2012), ACM, p. 20. 2

[SH06] SIMO J. C., HUGHES T. J.: *Computational inelasticity*, vol. 7. Springer Science & Business Media, 2006. 4

[She02] SHEWCHUK J.: What is a good linear finite element? interpolation, conditioning, anisotropy, and quality measures (preprint). *University of California at Berkeley 73* (2002). 9

[Sol15] SOLOMON J.: *Numerical algorithms: methods for computer vision, machine learning, and graphics*. CRC Press, 2015. 7

[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, 4, 9, 10

[SZS95] SULSKY D., ZHOU S.-J., SCHREYER H. L.: Application of a particle-in-cell method to solid mechanics. *Computer Physics Communications 87*, 1 (1995), 236 – 252. Particle Simulation Methods. 2

[TGK*17] TAMPUBOLON A. P., GAST T., KLÁR G., FU C., TERAN J., JIANG C., MUSETH K.: Multi-species simulation of porous sand and water mixtures. *ACM Trans. Graph. 36*, 4 (July 2017). 2, 9

[VFA06] VELARDI F., FRATERNALI F., ANGELILLO M.: Anisotropic constitutive equations and experimental tensile behavior of brain tissue. *Biomechanics and Modeling in Mechanobiology 5*, 1 (Mar 2006). 2

[WRK*10] WICKE M., RITCHIE D., KLINGNER B. M., BURKE S., SHEWCHUK J. R., O'BRIEN J. F.: Dynamic local remeshing for elastoplastic simulation. In *ACM Transactions on graphics (TOG)* (2010), vol. 29, ACM, p. 49. 2

[WT08] WOJTAN C., TURK G.: Fast viscoelastic behavior with thin features. *ACM transactions on graphics (TOG) 27*, 3 (2008), 47. 2

[XBP02] XIA Q. S., BOYCE M. C., PARKS D. M.: A constitutive model for the anisotropic elasticâĂŞplastic deformation of paper and paperboard. *International Journal of Solids and Structures 39*, 15 (2002), 4053 – 4071. 2

[YSB*15] YUE Y., SMITH B., BATTY C., ZHENG C., GRINSPUN E.: Continuum foam: A material point method for shear-dependent flows. *ACM Trans. Graph. 34* (2015), 160:1–160:20. 2

[YSC*18] YUE Y., SMITH B., CHEN P. Y., CHANTHARAYUKHONTHORN M., KAMRIN K., GRINSPUN E.: Hybrid grains: Adaptive coupling of discrete and continuum simulations of granular media. *ACM Trans. Graph. 37*, 6 (Dec. 2018), 283:1–283:19. 3, 9

[YT13] YU J., TURK G.: Reconstructing surfaces of particle-based fluids using anisotropic kernels. *ACM Transactions on Graphics (TOG) 32*, 1 (2013), 5. 9

[ZGW*13] ZHONG Z., GUO X., WANG W., LÉVY B., SUN F., LIU Y., MAO W.: Particle-based Anisotropic Surface Meshing. *ACM Transactions on Graphics 32*, 4 (2013), 99:1–99:14. 3