A Material Point Method for Viscoelastic Fluids, Foams and Sponges

Daniel Ram[†] Theodore Gast[†] Chenfanfu Jiang[†] Craig Schroeder[†] Alexey Stomakhin[⋆]
Joseph Teran[⋆†] Pirouz Kavehpour[†]

[†]University of California Los Angeles [⋆]Walt Disney Animation Studios

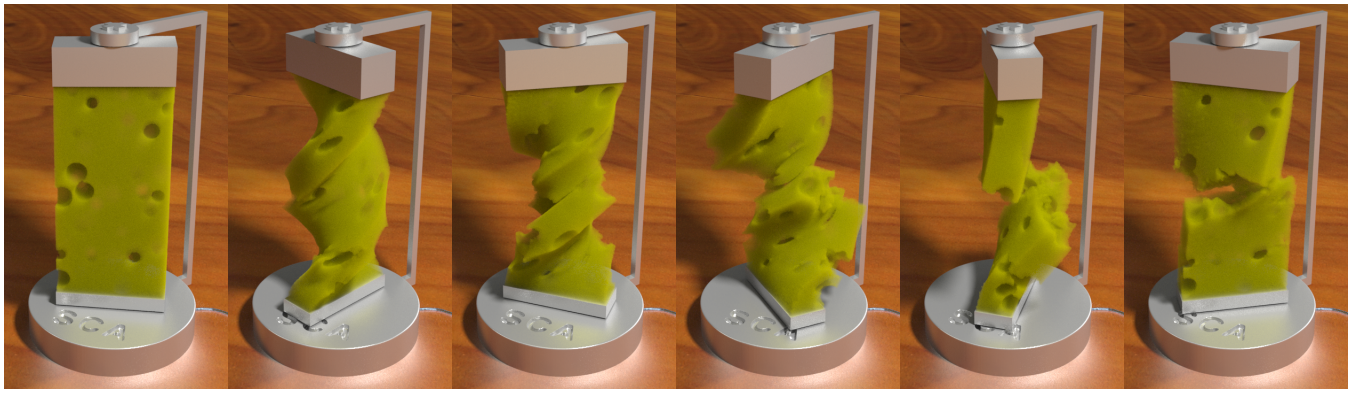

Figure 1: *A soft sponge is twisted. It fractures and collides with itself. The failure and contact phenomena are resolved automatically by the MPM approach.*

Abstract

We present a new Material Point Method (MPM) for simulating viscoelastic fluids, foams and sponges. We design our discretization from the upper convected derivative terms in the evolution of the left Cauchy-Green elastic strain tensor. We combine this with an Oldroyd-B model for plastic flow in a complex viscoelastic fluid. While the Oldroyd-B model is traditionally used for viscoelastic fluids, we show that its interpretation as a plastic flow naturally allows us to simulate a wide range of complex material behaviors. In order to do this, we provide a modification to the traditional Oldroyd-B model that guarantees volume preserving plastic flows. Our plasticity model is remarkably simple (foregoing the need for the singular value decomposition (SVD) of stresses or strains). Lastly, we show that implicit time stepping can be achieved in a manner similar to [Stomakhin et al. 2013] and that this allows for high resolution simulations at practical simulation times.

CR Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation I.6.8 [Simulation and Modeling]: Types of Simulation—Animation;

Keywords: MPM, complex fluids, elastoplastic, physically-based modeling

1 Introduction

Non-Newtonian fluid behavior is exhibited by a wide range of everyday materials including paint, gels, sponges, foams and various food components like ketchup and custard [Larson 1999]. These materials are often special kinds of colloidal systems (a type of mixture in which one substance is dispersed evenly throughout another), where dimensions exceed those usually associated with colloids (up to 1μm for the dispersed phase) [Hiemenz and Rajagopalan 1997; Larson 1999]. For example, when a gas and a liquid are shaken together, the gas phase becomes a collection of bubbles dispersed in the liquid: this is the most common observation of foams. While a standard Newtonian viscous stress is a component of the mechanical response of these materials, they are non-Newtonian in the sense that there are other, often elastoplastic, aspects of the stress response to flow rate and deformation. Comprehensive reviews are given in [Morrison and Ross 2002; Prudhomme and Kahn 1996; Schramm 1994; Larson 1999].

Discretization of these materials is challenging because of the wide range of behaviors exhibited and by the nonlinear governing equations. These materials can behave with elastic resistance to deformation but can also undergo very large strains and complex topological changes characteristic of fluids. While Lagrangian approaches are best for resolving the solid-like behavior and Eulerian approaches most easily resolve the fluid-like behavior, these materials are in the middle ground and this makes discretization difficult. The Material Point Method is naturally suited for this class of materials because it uses a Cartesian grid to resolve topology changes and self-collisions combined with Lagrangian tracking of mass, momentum and deformation on particles. In practice, the particle-wise deformation information can be used to represent elastoplastic stresses arising from changes in shape, while an Eulerian background grid is used for implicit solves.

We show that the MPM approach in [Stomakhin et al. 2013] can be generalized to achieve a wide range of viscoelastic, complex fluid effects. As in [Stomakhin et al. 2013], we show that implicit time stepping can easily be used to improve efficiency and allow for simulation at high spatial resolution. With our Oldroyd-inspired approach, we avoid the need for the SVD of either elastic or plastic responses. While SVD computation is not a bottleneck for MPM when done efficiently (see e.g [McAdams et al. 2011]), it is not a

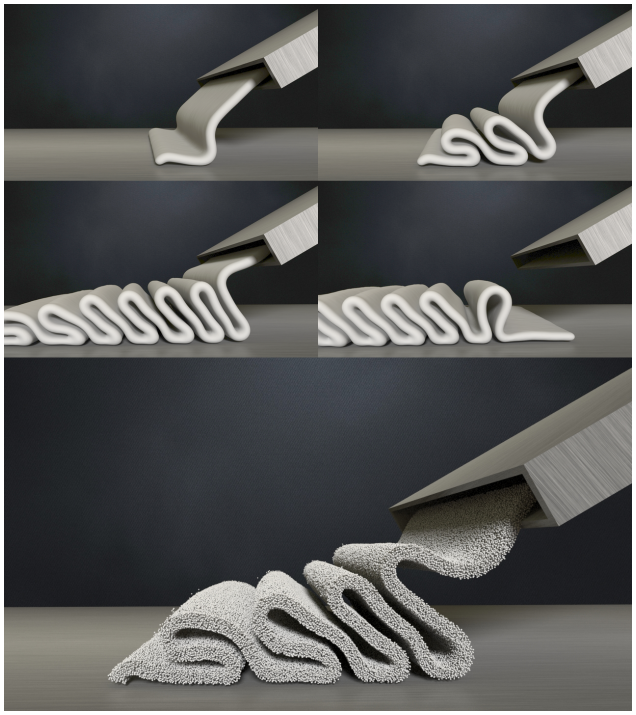

Figure 2: *Viennetta ice cream is poured onto a conveyor belt and forms characteristic folds. A particle view is shown on the bottom.*

straightforward implementation. More standard SVD implementations can have a dramatic impact on performance (see e.g. [Chao et al. 2010]). Thus although it is not essential for performance to avoid the SVD, it is preferable to avoid the need to implement them when, as with our model, they are not necessary for achieving desired behaviors.

We summarize our specific contributions as

- A new volume-preserving Oldroyd-B rate-based description of plasticity
- Semi-implicit MPM discretization of viscoelasticity and viscoplasticity, allowing for high spatial resolution simulations
- Rate-based plasticity that does not require an SVD

2 Related work

Terzopoulos and Fleischer were the first in computer graphics to show the effects possible with simulated elastoplastic materials [Terzopoulos and Fleischer 1988a; Terzopoulos and Fleischer 1988b]. Since those seminal works, many researchers have developed novel methods capable of replicating a wide range of material behaviors. Generally, these fall into one of three categories: Eulerian grid, Lagrangian mesh or particle based techniques. In addition to the following discussion, we summarize some aspects of our approach relative to a few representative approaches in Table 3.

Eulerian grid based approaches: Goktekin et al. [Goktekin et al. 2004] showed that the addition of an Eulerian elastic stress with Von Mises criteria plasticity to the standard level set based simulation of free surface Navier Stokes flows can capture a wide range of viscoelastic behaviors. Losasso et al. also use an Eulerian approach [Losasso et al. 2006]. Rasmussen et al. experiment with a range of viscous effects for level set based free surface melting flows in [Rasmussen et al. 2004]. Batty et al. use Eulerian approaches to efficiently simulate spatially varying viscous coiling and buckling [Batty and Bridson 2008; Batty and Houston 2011]. Carlson et al. also achieve a range of viscous effects in [Carlson et al. 2002].

Lagrangian mesh based approaches: Lagrangian methods naturally resolve deformation needed for elastoplasticity; however, large strains can lead to mesh tangling for practical flow scenarios and remeshing is required. Bargteil et al. show that this can achieve impressive results in [Bargteil et al. 2007]. This was later extended to embedded meshes in [Wojtan and Turk 2008] and further treatment of splitting and merging was achieved in [Wojtan et al. 2009]. Batty et al. used a reduced dimension approach to simulate thin viscous sheets with adaptively remeshed triangle meshes in [Batty et al. 2012].

Particle Methods: Ever since Desbrun and Gascuel [Desbrun and Gascuel 1996] showed that SPH can be used for a range of viscous behavior, particle methods have been popular for achieving complex fluid effects. Like Goktekin et al., Chang et al. [Chang et al. 2009] also use an Eulerian update of the strain for elastoplasic SPH simulations. Solenthaler et al. show that SPH can be used to compute strain and use this to get a range of elastoplastic effects [Solenthaler et al. 2007]. Becker et al. show that this can be generalized to large rotational motion in [Becker et al. 2009]. Gerszewski et al. also update deformation directly on particles [Gerszewski et al. 2009]. [Keiser et al. 2005] and [Müller et al. 2004] also add elastic effects into SPH formulations. Paiva et al. use a non-Newtonian model for fluid viscosity in [Paiva et al. 2006] and [Paiva et al. 2009].

Although MPM is a hybrid grid/particle method, particles are arguably the primary material representation. MPM has recently been used to simulate elastoplastic flows to capture snow in [Stomakhin et al. 2013] and varied, melting materials in [Stomakhin et al. 2014]. Yue et al. use MPM to simulate Herschel-Bulkley plastic flows for foam in [Yue et al. 2015]. Their approach is very similar to ours, however their treatment of plasticity is much more accurate and can handle a wider range of phenomena (notably, shear thickening). They also provide a novel particle splitting technique useful for resolving shearing flows that are problematic for a wide range of MPM simulations. However, their plastic flow update is more complicated and this is likely why they resort to explicit time stepping. With our comparatively simple plastic flow model, we show that semi-implicit time stepping as in [Stomakhin et al. 2013] can be achieved.

3 Governing equations

The governing equations arise from basic conservation of mass and momentum as

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

where ρ is the mass density, $\mathbf{v}$ is the velocity, $\boldsymbol{\sigma}$ is the Cauchy stress and $\mathbf{g}$ is gravitational acceleration. As is commonly done with viscoelastic complex fluids, we write the Cauchy stress as $\boldsymbol{\sigma} = \boldsymbol{\sigma}^N + \boldsymbol{\sigma}^E$ where $\boldsymbol{\sigma}^N = \frac{\mu^N}{2}\left(\frac{\partial\mathbf{v}}{\partial\mathbf{x}} + \frac{\partial\mathbf{v}}{\partial\mathbf{x}}^T\right)$ is the viscous Newtonian component and $\boldsymbol{\sigma}^E$ is the elastic component. We express the constitutive behavior through the elastic component of the left Cauchy Green strain. Specifically, the deformation gradient of the flow $\mathbf{F}$ can be decomposed as a product of elastic and plastic deformation as $\mathbf{F} = \mathbf{F}^E\mathbf{F}^P$ and the elastic left Cauchy Green strain is $\mathbf{b}^E = \mathbf{F}^E(\mathbf{F}^E)^T$ [Bonet and Wood 1997]. With this convention, we can define the elastic portion of the Cauchy stress via the stored elastic potential $\psi(\mathbf{b}^E)$ as $\boldsymbol{\sigma}^E = \frac{2}{J}\frac{\partial\psi}{\partial\mathbf{b}^E}\mathbf{b}^E$.

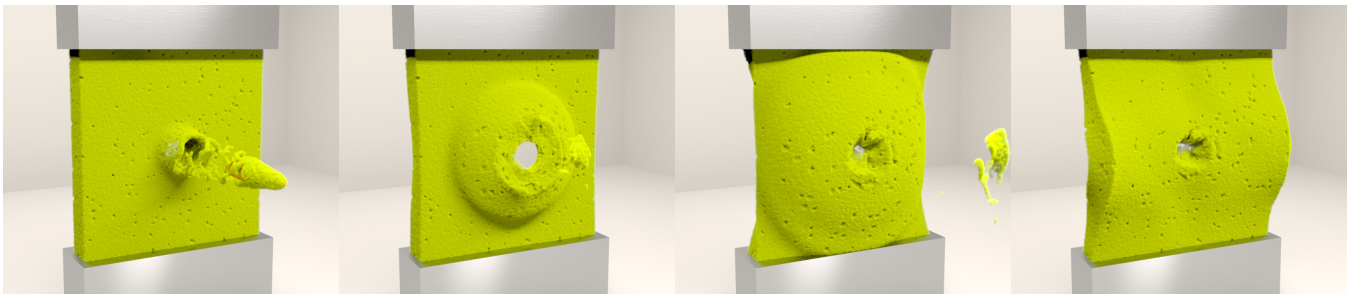

Figure 3: *A kinematic bullet is fired at a sponge, resulting in significant deformation and fracture.*

3.1 Left Cauchy-Green strain plasticity and the upper convected derivative

We can define the plastic flow using the temporal evolution of the elastic right Cauchy Green strain as in [Bonet and Wood 1997]. Rewriting $\mathbf{F}^E = \mathbf{F}(\mathbf{F}^P)^{-1}$, $\mathbf{b}^E = \mathbf{F}(\mathbf{C}^P)^{-1}\mathbf{F}^T$ where $\mathbf{C}^P = (\mathbf{F}^P)^T\mathbf{F}^P$ is the right plastic Cauchy Green strain. The Eulerian form of the temporal evolution is then obtained by taking the material derivative of $\mathbf{b}^E$ to get

$$\frac{D\mathbf{b}^E}{Dt} = \frac{D\mathbf{F}}{Dt}(\mathbf{C}^P)^{-1}\mathbf{F}^T + \mathbf{F}(\mathbf{C}^P)^{-1}\frac{D\mathbf{F}}{Dt}^T + \mathbf{F}\frac{D}{Dt}[(\mathbf{C}^P)^{-1}]\mathbf{F}^T. \tag{2}$$

With this view, the plastic flow is defined via $\frac{D}{Dt}[(\mathbf{C}^P)^{-1}]$. Combining this with $\frac{D}{Dt}\mathbf{F} = \frac{\partial \mathbf{v}}{\partial \mathbf{x}}\mathbf{F}$ (see e.g. [Bonet and Wood 1997]), the previous equation can be rewritten as

$$\frac{D\mathbf{b}^E}{Dt} = \frac{\partial \mathbf{v}}{\partial \mathbf{x}}\mathbf{b}^E + \mathbf{b}^E\frac{\partial \mathbf{v}}{\partial \mathbf{x}}^T + \mathbf{g}(\mathbf{b}^E) \tag{3}$$

where $\mathbf{g}(\mathbf{b}^E) = \mathbf{F}\frac{D}{Dt}[(\mathbf{C}^P)^{-1}]\mathbf{F}^T$ is used to describe the plastic flow rate. This equation is often abbreviated as

$$\overset{\triangledown}{\mathbf{b}}^E = \mathbf{g}(\mathbf{b}^E). \tag{4}$$

Here, the operator $\overset{\triangledown}{\mathbf{b}}^E$ (often referred to as the upper convected derivative) is defined to be $\overset{\triangledown}{\mathbf{b}}^E \equiv \frac{D}{Dt}\mathbf{b}^E - \frac{\partial \mathbf{v}}{\partial \mathbf{x}}\mathbf{b}^E - \mathbf{b}^E\frac{\partial \mathbf{v}}{\partial \mathbf{x}}^T$ (see e.g. [Larson 1999]).

3.2 Von Mises plasticity

The Von Mises model [Bonet and Wood 1997] achieves plasticity through the rate $\mathbf{g}(\mathbf{b}^E) = -2\dot{\gamma}\delta\frac{\partial f(\tau)}{\partial \tau}\mathbf{b}^E$, where τ is the Kirchhoff stress, $\dot{\gamma}$ is the plastic multiplier, $f(\tau)$ is the Von Mises yield condition, and $\delta = 1$ if $f(\tau) \geq 0$, $\delta = 0$ otherwise. However, this is relatively difficult to discretize given the conditional nature of the function. It is often more straightforward to just work directly with $\mathbf{F}^E$ and $\mathbf{F}^P$ in that case (see e.g [Stomakhin et al. 2013]), however Yue et al [Yue et al. 2015] do discretize this directly.

3.3 Oldroyd-B plasticity

The Oldroyd-B model [Larson 1999; Teran et al. 2008] can be see as an alternative definition of $\mathbf{g}(\mathbf{b}^E) = \frac{1}{Wi}(\mathbf{I} - \mathbf{b}^E)$. Combining this with $\mathbf{g}(\mathbf{b}^E) = \mathbf{F}\frac{D}{Dt}[(\mathbf{C}^P)^{-1}]\mathbf{F}^T$ shows that the plastic flow of this model is $\frac{D}{Dt}[(\mathbf{C}^P)^{-1}] = \frac{1}{Wi}(\mathbf{C}^{-1} - (\mathbf{C}^P)^{-1})$ where $\mathbf{C} = \mathbf{F}^T\mathbf{F}$ is the right Cauchy Green strain. This expression for $\mathbf{g}(\mathbf{b}^E)$ is very simple in comparison with that of Von Mises. This simplicity allows for a much easier treatment of temporal discretization needed for implicit time stepping. Specifically, we show in Section 4 that this simple definition of $\mathbf{g}(\mathbf{b}^E)$ facilitates the implicit description of the plastic flow in terms of discrete grid node velocities. We can see, both from the $\frac{1}{Wi}(\mathbf{I} - \mathbf{b}^E)$ and $\frac{D}{Dt}[(\mathbf{C}^P)^{-1}] = \frac{1}{Wi}(\mathbf{C}^{-1} - (\mathbf{C}^P)^{-1})$ terms that the plasticity achieves a strong damping of the elastic component of the stress. The severity of this damping is inversely proportionate to the Weissenberg number Wi. That is, the smaller the Weissenberg number, the faster the elastic strain is damped to the identity, thus releasing elastic potential and associated resistance to deformation. Thus, the Weissenberg number directly controls the amount of the plasticity.

3.4 Volume preserving plasticity

The plastic flow in the Oldroyd model will not generally be volume preserving. Since many plastic flows, including those of foams, exhibit this behavior we provide a modification to the standard Oldroyd model that will satisfy this. If we define $\mathbf{b}^E_{OB}$ to obey $\overset{\triangledown}{\mathbf{b}}^E_{OB} = \frac{1}{Wi}(\mathbf{I} - \mathbf{b}^E_{OB})$, then we define a new elastic left Cauchy Green strain as

$$\mathbf{b}^E \equiv \left(\frac{J}{J_{OB}}\right)^{\frac{2}{3}}\mathbf{b}^E_{OB}, \tag{5}$$

where $J = \det(\mathbf{F})$ and $J_{OB} = \sqrt{\det(\mathbf{b}^E_{OB})}$. Using this definition, $\det(\mathbf{b}^E) = J^2$ and since by definition $\det(\mathbf{b}^E) = \det(\mathbf{F}^E)^2$ and $J = \det(\mathbf{F}^E)\det(\mathbf{F}^P)$ we see that it must be true that $\det(\mathbf{F}^P) = 1$, and thus the plastic flow is volume preserving.

3.5 Modified plastic flow

This modification to the Oldroyd plasticity obeys

$$\overset{\triangledown}{\mathbf{b}}^E = \frac{D}{Dt}\left(\left(\frac{J}{J_{OB}}\right)^{\frac{2}{3}}\right)\mathbf{b}^E_{OB} + \frac{1}{Wi}\left(\frac{J}{J_{OB}}\right)^{\frac{2}{3}}(\mathbf{I} - \mathbf{b}^E_{OB}) \tag{6}$$

which has the plastic flow

$$\begin{aligned}\frac{D}{Dt}[(\mathbf{C}^P)^{-1}] = \frac{D}{Dt}\left(\left(\frac{J}{J_{OB}}\right)^{\frac{2}{3}}\right)(\mathbf{C}^P_{OB})^{-1} + \\ \frac{1}{Wi}\left(\frac{J}{J_{OB}}\right)^{\frac{2}{3}}(\mathbf{C}^{-1} - (\mathbf{C}^P_{OB})^{-1}).\end{aligned} \tag{7}$$

We do not need to solve for $\mathbf{b}^E$ using the definition of its plastic flow. In practice, we solve for the comparatively simple $\mathbf{b}^E_{OB}$ and then obtain the elastic stress as $\mathbf{b}^E = \left(\frac{J}{J_{OB}}\right)^{\frac{2}{3}}\mathbf{b}^E_{OB}$. We only provide this derivation here to show that there is a plastic flow associated with this definition of the elastic strain.

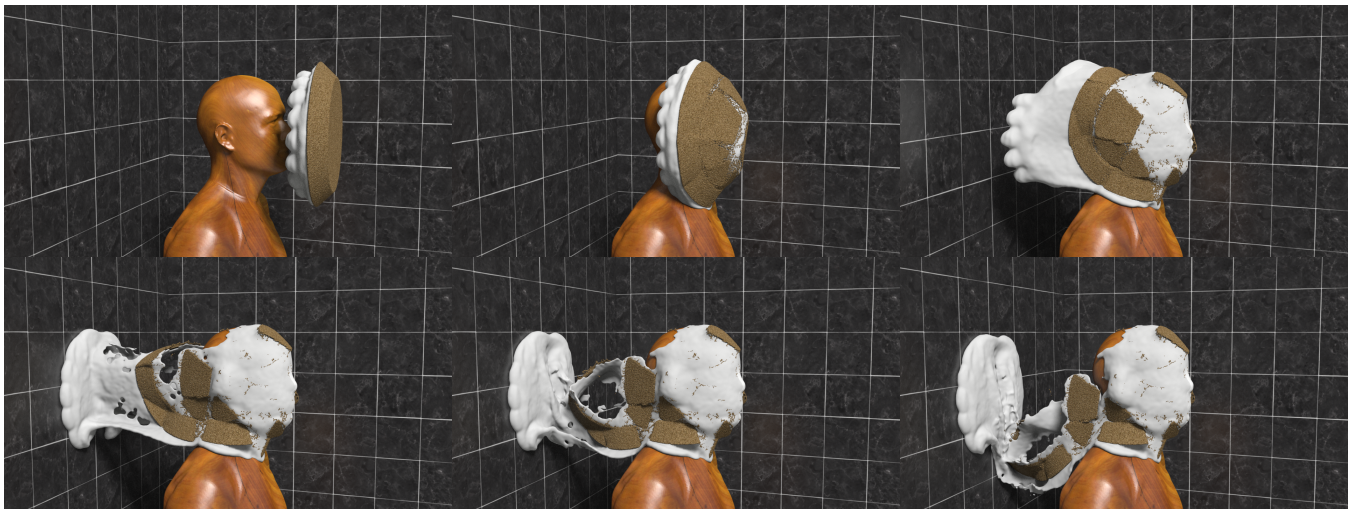

Figure 4: *A pie with a stiff crust and soft whipped cream is thrown at a mannequin.*

3.6 Elasticity

We define constitutive behavior through the compressible Neo-Hookean elastic potential energy density as

$$\psi(\mathbf{b}^E) = \frac{\mu}{2}(\text{tr}(\mathbf{b}^E) - 3) - \mu \ln(J) + \frac{\lambda}{2}(J-1)^2 \tag{8}$$

with associated Cauchy stress

$$\boldsymbol{\sigma}^E = \frac{\mu}{J}(\mathbf{b}^E - \mathbf{I}) + \lambda(J-1)\mathbf{I}. \tag{9}$$

4 Material point method

We closely follow the algorithm from [Stomakhin et al. 2013]. The only difference is in the discrete Eulerian grid node forces and force derivatives. All steps in the algorithm not related to the update of grid node velocities are the same; we simply change the nature of stress-based forces. In this section, we describe how to modify the potential-based definition of these forces to discretize our new governing equations. We refer the reader to [Stomakhin et al. 2013] for all other steps in the MPM time stepping algorithm.

Using the notation from [Stomakhin et al. 2013], we denote position, velocity and deformation gradient of particle p at time t^n as $\mathbf{x}_p^n$, $\mathbf{v}_p^n$ and $\mathbf{F}_p^n$ respectively. Eulerian grid node locations are denoted as $\mathbf{x_i}$ where $\mathbf{i} = (i, j, k)$ is the grid node index. The weights at time t^n are $w_{\mathbf{i}p}^n = N_{\mathbf{i}}(\mathbf{x}_p^n)$, where $N_{\mathbf{i}}(\mathbf{x})$ is the interpolation function associated with grid node $\mathbf{i}$ and the weight gradients are $\nabla w_{\mathbf{i}p}^n = \nabla N_{\mathbf{i}}(\mathbf{x}_p^n)$. As in [Stomakhin et al. 2013], we define the forces on the Eulerian grid nodes as the derivative of an energy with respect to grid node locations. We do not actually move grid nodes, but we consider their movement to define grid node velocities $\mathbf{v_i}$ as $\hat{\mathbf{x}}_{\mathbf{i}} = \mathbf{x_i} + \Delta t \mathbf{v_i}$. Using $\hat{\mathbf{x}}$ to denote the vector of all grid nodes, we define the potential

$$\Phi(\hat{\mathbf{x}}) = \sum_p \left(\Phi^E(\hat{\mathbf{x}})V_p^0 + \Phi^N(\hat{\mathbf{x}})V_p^n\right) \tag{10}$$

where $\Phi^E(\hat{\mathbf{x}})$ is the elastoplastic component of the potential energy density $\Phi^E(\hat{\mathbf{x}}) = \psi\left(\hat{\mathbf{b}}^E(\hat{\mathbf{x}})\right)$ and $\Phi^N(\hat{\mathbf{x}})$ is the Newtonian viscous potential energy density

$$\Phi^N(\hat{\mathbf{x}}) = \mu^N \hat{\boldsymbol{\epsilon}}_p(\hat{\mathbf{x}}) : \hat{\boldsymbol{\epsilon}}_p(\hat{\mathbf{x}}) = \sum_{i,j} \mu^N \hat{\epsilon}_{p_{ij}}(\hat{\mathbf{x}})\hat{\epsilon}_{p_{ij}}(\hat{\mathbf{x}}). \tag{11}$$

Here $\hat{\boldsymbol{\epsilon}}_p(\hat{\mathbf{x}}) = \frac{1}{2}\left(\nabla\hat{\mathbf{v}}(\hat{\mathbf{x}}) + (\nabla\hat{\mathbf{v}}(\hat{\mathbf{x}}))^T\right)$ is the strain rate at $\mathbf{x}_p^n$ induced by the grid node motion defined by $\hat{\mathbf{x}}$ over the time step and $\nabla\hat{\mathbf{v}}(\hat{\mathbf{x}}) = \sum_{\mathbf{i}} \frac{\hat{\mathbf{x}}_{\mathbf{i}} - \mathbf{x}_{\mathbf{i}}}{\Delta t}(\nabla w_{\mathbf{i}p}^n)^T$. As in [Stomakhin et al. 2013], V_p^0 is the volume of the material originally occupied by the particle p. However, for the viscous Newtonian potential, we are approximating an integral over the time t^n configuration of the material so we have $V_p^n = \det(\mathbf{F}_p^n)V_p^0$.

As in [Stomakhin et al. 2013], we store a deformation gradient $\mathbf{F}_p^n$ on each particle and update it using

$$\hat{\mathbf{F}}(\hat{\mathbf{x}}) = (\mathbf{I} + \Delta t \nabla\hat{\mathbf{v}}(\hat{\mathbf{x}}))\mathbf{F}_p^n. \tag{12}$$

We use this to define $\hat{J}_p(\hat{\mathbf{x}}) = \det(\hat{\mathbf{F}}(\hat{\mathbf{x}}))$ in the definition of

$$\hat{\mathbf{b}}^E(\hat{\mathbf{x}}) = \left(\frac{\hat{J}_p(\hat{\mathbf{x}})^2}{\det(\hat{\mathbf{b}}_{OB_p}^E(\hat{\mathbf{x}}))}\right)^{\frac{1}{3}} \hat{\mathbf{b}}_{OB_p}^E(\hat{\mathbf{x}}). \tag{13}$$

Similar to the treatment in Equation 12, we store $\mathbf{b}_{OB_p}^{E^n}$ on each particle and discretize the upper convected derivative terms in the evolution equation for $\mathbf{b}_{OB}^E$ to get

$$\begin{aligned}\hat{\mathbf{b}}_{OB_p}^E(\hat{\mathbf{x}}) =& \Delta t \nabla\hat{\mathbf{v}}(\hat{\mathbf{x}})\mathbf{b}_{OB_p}^{E^n} + \Delta t \mathbf{b}_{OB_p}^{E^n}(\nabla\hat{\mathbf{v}}(\hat{\mathbf{x}}))^T \\ &+ \frac{\Delta t}{Wi}\mathbf{I} + \left(1 - \frac{\Delta t}{Wi}\right)\mathbf{b}_{OB_p}^{E^n}\end{aligned} \tag{14}$$

The force on the grid nodes is defined as $\mathbf{f}(\hat{\mathbf{x}}) = -\frac{\partial\Phi}{\partial\hat{\mathbf{x}}}(\hat{\mathbf{x}})$ and it is used in the implicit update of grid velocities $\mathbf{v}_{\mathbf{i}}^{n+1}$ exactly as in [Stomakhin et al. 2013]. We work out these derivatives as well as the $\frac{\partial\mathbf{f}}{\partial\hat{\mathbf{x}}}(\hat{\mathbf{x}})$ in the appendix.

5 Results

In Figure 1, a sponge is twisted with top and bottom fixed by Dirichlet boundary conditions. Dynamic fracture and self collision are naturally handled. In Figure 3, the top and bottom of a sponge are held in place as we shoot it with a kinematic bullet. The animation is in slow motion to show the detailed material response after the impact. In Figure 6, we simulate a stream of shaving foam hitting the ground, and compare it with real world footage. Our method captures the S-shaped buckling and merging behaviors. It also exhibits similar elasto-plastic responses. In Figure 5,

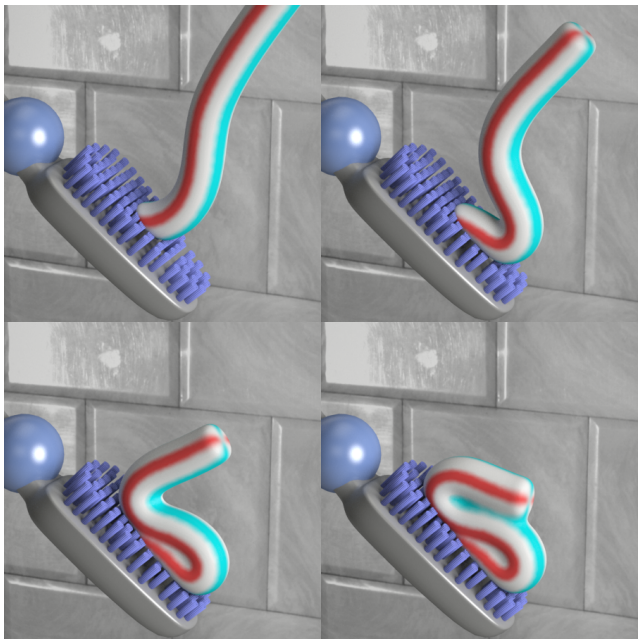

Figure 5: *A simulation of toothpaste. Unlike the shaving foam, Newtonian viscosity dominates material behavior.*

we simulate toothpaste falling onto a toothbrush. Unlike the shaving foam, Newtonian viscosity dominates material behavior. Figure 2 shows a simulation of manufacturing Viennetta ice cream. It captures the characteristic folding behavior. In Figure 4, we model a pie and throw it at a mannequin. The fracture pattern of the crust is prescored with weak MPM particles. The cream exhibits detailed splitting and merging behavior. For the particle-grid transfers, we used the affine Particle-In-Cell (APIC) method from [Jiang et al. 2015] . We found that using APIC greatly reduced positional artifacts of the pie particles. We do not perform any explicit particle resampling because self-collision and topology change are naturally handled by MPM.

The material parameters used in our examples are given in Table 1. The simulation times are shown in Table 2. All simulations were performed on Intel Xeon machines. All renderings were done with Mantra in Houdini. For foam, toothpaste, and Viennetta ice cream, surfaces were reconstructed with OpenVDB [Museth 2014] and rendered with subsurface scattering. The sponges were rendered as a density field.

6 Discussions

We found that using a Jacobi preconditioner greatly reduced simulation run times. For example, in the shooting sponge test (Figure 3), the Jacobi preconditioner reduces the number of CG iterations by a factor of 6.

While we have used our method successfully in simulating a variety of materials, it has some limitations. Many of these are related to the Oldroyd-B model. For example, unlike the approach in [Yue et al. 2015], our approach cannot handle shear thickening. Therefore, the model cannot be applied to materials such as oobleck. Our method also does not handle material softening or hardening.

Our update rule of $\mathbf{b}_{OB}^E$ allows for inversion which the constitutive model cannot handle. While $\mathbf{b}_{OB}^E$ should remain positive definite, we have found this to be only partially required. In particular, (8) involves the quantity $\text{tr}(\mathbf{b}^E)$, which we must ensure is bounded from below. If $\mathbf{b}^E$ is positive definite, then $\text{tr}(\mathbf{b}^E) > 0$. We also compute $\det(\mathbf{b}_{OB}^E)^{-\frac{1}{3}}$, which is problematic if $\mathbf{b}_{OB}^E$ may become singular. We avoid these problems in practice by taking advantage of the optimization-based integrator from [Gast et al. 2015]. We add a large penalty to our objective when the determinant or trace of $\mathbf{b}_{OB}^E$ becomes infeasible; the line search in our optimizer then discards these configurations. While bounding the trace and determinant does not enforce definiteness in 3D, this strategy worked well in practice. Not enforcing these produces popping artifacts.

	ρ	μ	λ	μ^N	Wi
Twisting sponge	2	3.6×10^2	1.4×10^3	0	50
Shooting sponge	1	3.6×10^2	1.4×10^3	0	50
Shaving foam	0.2	5	50	1×10^{-4}	0.5
Toothpaste	1	0.839	8.39	1×10^{-1}	0.4
Viennetta ice cream	1	1	10	5×10^{-5}	0.1
Pie cream	0.2	5	50	1×10^{-7}	1×10^{-4}
Pie crust	0.5	5×10^5	4×10^6	1×10^{-8}	1×10^{30}
Pie crust scored	0.5	5	10	1×10^{-5}	1

Table 1: *Material parameters.*

Acknowledgements

UCLA authors were partially supported by NSF CCF-1422795, ONR (N000141110719, N000141210834), Intel STC-Visual Computing Grant (20112360) as well as a gift from Disney Research. We also thank Matthew Wang for providing his 3D head model for the pie example.

References

BARGTEIL, A., WOJTAN, C., HODGINS, J., AND TURK, G. 2007. A finite element method for animating large viscoplastic flow. *ACM Trans. Graph. 26*, 3.

BATTY, C., AND BRIDSON, R. 2008. Accurate viscous free surfaces for buckling, coiling, and rotating liquids. In *Proc 2008 ACM/Eurographics Symp Comp Anim*, 219–228.

BATTY, C., AND HOUSTON, B. 2011. A simple finite volume method for adaptive viscous liquids. In *Proc 2011 ACM SIGGRAPH/Eurograp Symp Comp Anim*, 111–118.

BATTY, C., URIBE, A., AUDOLY, B., AND GRINSPUN, E. 2012. Discrete viscous sheets. 113:1–113:7.

BECKER, M., IHMSEN, M., AND TESCHNER, M. 2009. Corotated sph for deformable solids. In *Eurographics Conf. Nat. Phen.*, 27–34.

BONET, J., AND WOOD, R. 1997. *Nonlinear Continuum Mechanics for Finite Element Analysis*. Cambridge University Press.

CARLSON, M., MUCHA, P., HORN, R. V., AND TURK, G. 2002. Melting and flowing. In *ACM SIGGRAPH/Eurographics Symp. Comp. Anim.*, 167–174.

CHANG, Y., BAO, K., LIU, Y., ZHU, J., AND WU, E. 2009. A particle-based method for viscoelastic fluids animation. In *ACM Symp. Virt. Real. Soft. Tech.*, 111–117.

CHAO, I., PINKALL, U., SANAN, P., AND SCHRÖDER, P. 2010. A simple geometric model for elastic deformations. In *ACM Transactions on Graphics (TOG)*, vol. 29, ACM, 38.

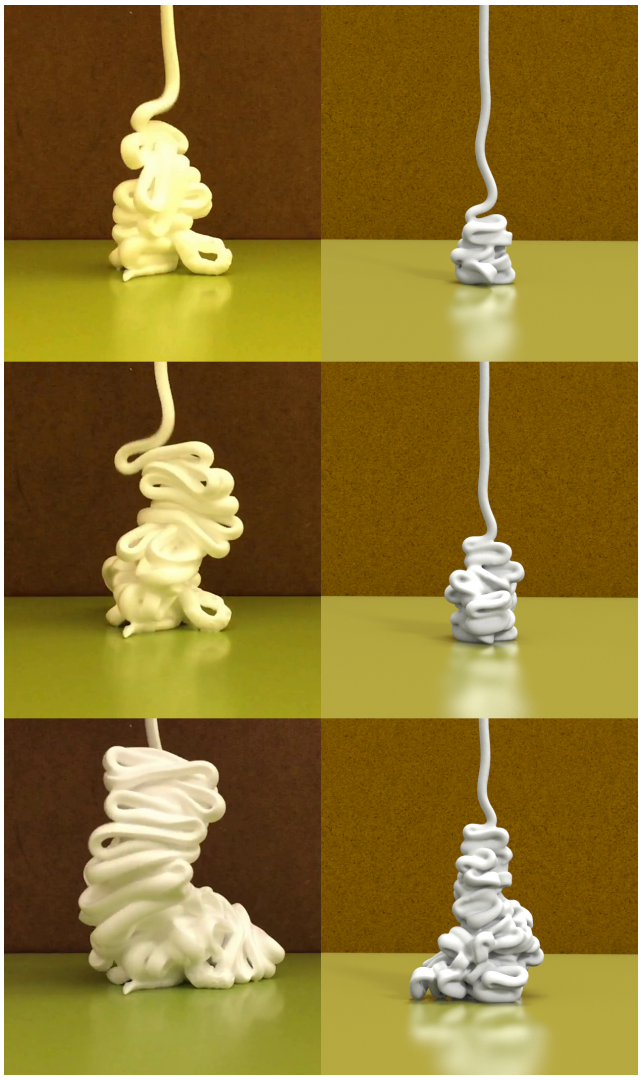

Figure 6: *Simulated shaving foam (right) is compared with real world footage (left). The simulation captures the characteristic S-shaped buckling and elastic behavior.*

DESBRUN, M., AND GASCUEL, M. 1996. Smoothed particles: A new paradigm for animating highly deformable bodies. In *Eurographics Workshop Comp. Anim. Sim.*, 61–76.

GAST, T., SCHROEDER, C., STOMAKHIN, A., JIANG, C., AND TERAN, J. 2015. Optimization integrator for large time steps. *IEEE Trans. Vis. Comp. Graph. (in review)*.

GERSZEWSKI, D., BHATTACHARYA, H., AND BARGTEIL, A. 2009. A point-based method for animating elastoplastic solids. In *Proc ACM SIGGRAPH/Eurograph Symp Comp Anim*, 133–138.

GOKTEKIN, T., BARGTEIL, A., AND O'BRIEN, J. 2004. A method for animating viscoelastic fluids. *ACM Trans. Graph. 23*, 3, 463–468.

HIEMENZ, P., AND RAJAGOPALAN, R. 1997. *Principles of Colloid and Surface Chemistry*. Marcel Dekker.

JIANG, C., SCHROEDER, C., SELLE, A., TERAN, J., AND STOMAKHIN, A. 2015. The affine particle-in-cell method. *ACM Trans. Graph. (to appear)*.

KEISER, R., ADAMS, B., GASSER, D., BAZZI, P., DUTRÉ, P., AND GROSS, M. 2005. A unified lagrangian approach to solid-fluid animation. In *Eurographics/IEEE VGTC Conf. Point-Based Graph.*, 125–133.

LARSON, R. G. 1999. *The Structure and Rheology of Complex Fluids*. Oxford University Press: New York.

LOSASSO, F., IRVING, G., GUENDELMAN, E., AND FEDKIW, R. 2006. Melting and burning solids into liquids and gases. *IEEE Trans. Vis. Comp. Graph. 12*, 343–352.

MCADAMS, A., ZHU, Y., SELLE, A., EMPEY, M., TAMSTORF, R., TERAN, J., AND SIFAKIS, E. 2011. Efficient elasticity for character skinning with contact and collisions. In *ACM Transactions on Graphics (TOG)*, vol. 30, ACM, 37.

MORRISON, I., AND ROSS, S. 2002. *Colloidal Dispersions: Suspensions, Emulsions and Foams*. Wiley Interscience.

MÜLLER, M., KEISER, R., NEALEN, A., PAULY, M., GROSS, M., AND ALEXA, M. 2004. Point based animation of elastic, plastic and melting objects. In *ACM SIGGRAPH/Eurographics Symp. Comp. Anim.*, 141–151.

MUSETH, K. 2014. A flexible image processing approach to the surfacing of particle-based fluid animation (invited talk). In *Mathematical Progress in Expressive Image Synthesis I*, vol. 4 of *Mathematics for Industry*. 81–84.

PAIVA, A., PETRONETTO, F., LEWINER, T., AND TAVARES, G. 2006. Particle-based non-newtonian fluid animation for melting objects. In *Conf. Graph. Patt. Images*, 78–85.

PAIVA, A., PETRONETTO, F., LEWINER, T., AND TAVARES, G. 2009. Particle-based viscoplastic fluid/solid simulation. *Comp. Aided Des. 41*, 4, 306–314.

PRUDHOMME, R., AND KAHN, S. 1996. *Foams: Theory, Measurements, and Applications*. Marcel Dekker.

RASMUSSEN, N., ENRIGHT, D., NGUYEN, D., MARINO, S., SUMNER, N., GEIGER, W., HOON, S., AND FEDKIW, R. 2004. Directable photorealistic liquids. In *ACM SIGGRAPH/Eurographics Symp. Comp. Anim.*, 193–202.

SCHRAMM, L. 1994. *Foams: Fundamentals and Applications in the Petroleum Industry*. ACS.

SOLENTHALER, B., SCHLÄFLI, J., AND PAJAROLA, R. 2007. A unified particle model for fluid-solid interactions. *Comp. Anim. Virt. Worlds 18*, 1, 69–82.

STOMAKHIN, A., SCHROEDER, C., CHAI, L., TERAN, J., AND SELLE, A. 2013. A material point method for snow simulation. *ACM Trans. Graph. 32*, 4, 102:1–102:10.

STOMAKHIN, A., SCHROEDER, C., JIANG, C., CHAI, L., TERAN, J., AND SELLE, A. 2014. Augmented mpm for phase-change and varied materials. *ACM Trans. Graph. 33*, 4, 138:1–138:11.

TERAN, J., FAUCI, L., AND SHELLEY, M. 2008. Peristaltic pumping and irreversibility of a stokesian viscoelastic fluid. *Phys Fl 20*, 7.

TERZOPOULOS, D., AND FLEISCHER, K. 1988. Deformable models. *Vis Comp 4*, 6, 306–331.

	Min/Frame	Particle #	Threads	CPU	Δx	Grid Resolution
Twisting sponge	5.3	9.1×10^5	20	3.00GHz	0.0366	245^3
Shooting sponge	2.0	7.2×10^5	16	2.90GHz	0.0402	175^3
Shaving foam	0.93	1.1×10^6	12	3.47GHz	0.0019	257^3
Toothpaste	0.28	2.8×10^5	16	2.90GHz	0.0082	$244 \times 487 \times 244$
Viennetta ice cream	1.11	1.2×10^6	12	2.67GHz	0.0026	$385 \times 96 \times 64$
Pie	23.6	1.3×10^6	12	3.07GHz	0.0024	333^3

Table 2: *Simulation performance.*

TERZOPOULOS, D., AND FLEISCHER, K. 1988. Modeling inelastic deformation: Viscolelasticity, plasticity, fracture. *SIGGRAPH Comp Graph 22*, 4, 269–278.

WOJTAN, C., AND TURK, G. 2008. Fast viscoelastic behavior with thin features. *ACM Trans. Graph. 27*, 3, 47:1–47:8.

WOJTAN, C., THÜREY, N., GROSS, M., AND TURK, G. 2009. Deforming meshes that split and merge. *ACM Trans. Graph. 28*, 3, 76:1–76:10.

YUE, Y., SMITH, B., BATTY, C., ZHENG, C., AND GRINSPUN, E. 2015. Continuum foam: A material point method for shear-dependent flows. *ACM Trans. Graph.*.

Appendix: Derivatives

While the potential energy contains many elements and computing its second derivatives seems like a hopeless task, this is not the case. Breaking the potential energy into small pieces makes the implementation straightforward to implement and debug. In this appendix, we present pseudocode that may be used to compute the potential energy $\Phi = \sum_p \Phi_p$ along with its derivatives, $\frac{\partial \Phi}{\partial \mathbf{x_i}} = \sum_p \Phi_{p,\mathbf{i}}$ and $\frac{\partial^2 \Phi}{\partial \mathbf{x_i} \partial \mathbf{x_j}} = \sum_p \Phi_{p,\mathbf{ij}}$. The following computational steps may be used to compute the potential energy contribution of a particle Φ_p. Note that all of the quantities computed below, except for the final result Φ_p, are intermediate quantities used to break the computation into many parts. Most of them have no particular physical significance, and most have no particular relationship to similarly named quantities elsewhere in this manuscript. The bold capitalized quantities are matrices, and the rest are scalars.

$$\mathbf{A}_p \leftarrow \sum_{\mathbf{i}} (\hat{\mathbf{x}}_\mathbf{i} - \mathbf{x}^n_\mathbf{i})(\nabla w^n_{\mathbf{i}p})^T \qquad \mathbf{B}_p \leftarrow \mathbf{A}_p \mathbf{b}^{E^n}_{OBp}$$

$$\mathbf{G}_p \leftarrow \mathbf{b}^{E^n}_{OBp} + \frac{\Delta t}{Wi}\left(\mathbf{I} - \mathbf{b}^{E^n}_{OBp}\right) \qquad \hat{\mathbf{F}}_p \leftarrow (\mathbf{I} + \mathbf{A}_p)\mathbf{F}^n_p$$

$$\mathbf{S}_p \leftarrow \mathbf{G}_p + \mathbf{B}_p + \mathbf{B}_p^T \qquad \mathbf{H}_p \leftarrow \hat{\mathbf{F}}_p^{-1}$$

$$J_p \leftarrow \det(\hat{\mathbf{F}}_p) \qquad a_p \leftarrow \frac{\lambda}{2}(J_p - 1)^2$$

$$q_p \leftarrow \frac{1}{2\Delta t^2}\left(\|\mathbf{A}_p\|_F^2 + \mathbf{A}_p^T : \mathbf{A}_p\right) \qquad b_p \leftarrow \mu \ln(J_p)$$

$$c_p \leftarrow \mu^N_P q_p \det(\mathbf{F}^n_p) \qquad g_p \leftarrow \mathrm{tr}(\mathbf{S}_p)$$

$$\mathbf{K}_p \leftarrow \mathbf{S}_p^{-1} \qquad h_p \leftarrow \det(\mathbf{S}_p)$$

$$k_p \leftarrow h_p^{-\frac{1}{d}} \qquad m_p \leftarrow J_p^{\frac{2}{d}}$$

$$n_p \leftarrow k_p g_p \qquad p_p \leftarrow \frac{\mu}{2} m_p n_p$$

$$\Phi_p \leftarrow V_p(p_p - b_p + a_p + c_p)$$

The next set of routines are for the first derivatives of the quantities above, with the final result being the potential energy derivative for a particle, $\Phi_{p,\mathbf{i}}$. Note that these routines use the quantities computed above. Intermediate quantities of the form $c_{p,\mathbf{i}}$ are related to the intermediates above by $c_{p,\mathbf{i}} = \frac{\partial c_p}{\partial \mathbf{x_i}}$, which allows for incremental testing. All quantities computed below are vectors.

Method	Elastoplastic	Viscosity	No SVD	Implicit	No Remeshing
[Batty and Bridson 2008]	✗	✓	✓	✓	✓
[Wojtan et al. 2009]	✓	✓	✗	○†	✗
[Batty and Houston 2011]	✗	✓	✓	✓	○‡
[Batty et al. 2012]	✗	✓	✓	✓	✗
[Stomakhin et al. 2013]	✓	✗	✗	✓	✓
[Stomakhin et al. 2014]	✓	✗	✗	✓	✓
[Yue et al. 2015]	✓	✓	✓	✗	✓
Our method	✓	✓	✓	✓	✓

Table 3: *Feature comparison with some existing methods. †This method is not implicit in elasticity. ‡This method requires adaptive refinement of a BCC lattice.*

$$\bar{b}_{p\mathbf{i}} \leftarrow \mathbf{b}^{E^n}_{OBp} \nabla w^n_{\mathbf{i}p} \qquad \bar{f}_{p\mathbf{i}} \leftarrow (\mathbf{F}^n_p)^T \nabla w^n_{\mathbf{i}p}$$

$$\bar{h}_{p\mathbf{i}} \leftarrow \mathbf{H}_p^T \bar{f}_{p\mathbf{i}} \qquad \bar{k}_{p\mathbf{i}} \leftarrow \mathbf{K}_p^T \bar{b}_{p\mathbf{i}}$$

$$J_{p,\mathbf{i}} \leftarrow J_p \bar{h}_{p\mathbf{i}} \qquad a_{p,\mathbf{i}} \leftarrow \lambda(J_p - 1)J_{p,\mathbf{i}}$$

$$q_{p,\mathbf{i}} \leftarrow \frac{1}{\Delta t^2}(\mathbf{A}_p \nabla w^n_{\mathbf{i}p} + \mathbf{A}_p^T \nabla w^n_{\mathbf{i}p}) \qquad b_{p,\mathbf{i}} \leftarrow \mu \bar{h}_{p\mathbf{i}}$$

$$c_{p,\mathbf{i}} \leftarrow \mu^N_P \det(\mathbf{F}^n_p) q_{p,\mathbf{i}} \qquad g_{p,\mathbf{i}} \leftarrow 2\bar{b}_{p\mathbf{i}}$$

$$k_{p,\mathbf{i}} \leftarrow -\frac{2k_p}{d}\bar{k}_{p\mathbf{i}} \qquad m_{p,\mathbf{i}} \leftarrow \frac{2m_p}{d}\bar{h}_{p\mathbf{i}}$$

$$p_{p,\mathbf{i}} \leftarrow \frac{\mu}{2}(m_{p,\mathbf{i}} n_p + m_p n_{p,\mathbf{i}}) \qquad n_{p,\mathbf{i}} \leftarrow k_{p,\mathbf{i}} g_p + k_p g_{p,\mathbf{i}}$$

$$\Phi_{p,\mathbf{i}} \leftarrow V_p(p_{p,\mathbf{i}} - b_{p,\mathbf{i}} + a_{p,\mathbf{i}} + c_{p,\mathbf{i}})$$

The final set of routines are for second derivatives, with the final result being the potential energy Hessian for a particle, $\Phi_{p,\mathbf{ij}}$. Intermediate quantities of the form $c_{p,\mathbf{ij}}$ are related to the intermediates above by $c_{p,\mathbf{ij}} = \frac{\partial c_{p,\mathbf{i}}}{\partial \mathbf{x_j}}$. All quantities computed below are matrices.

$$J_{p,\mathbf{ij}} \leftarrow J_p \bar{h}_{p,\mathbf{i}} \bar{h}^T_{p,\mathbf{j}} - J_p \bar{h}_{p,\mathbf{j}} \bar{h}^T_{p,\mathbf{i}}$$

$$a_{p,\mathbf{ij}} \leftarrow \lambda J_{p,\mathbf{i}} J^T_{p,\mathbf{j}} + \lambda(J_p - 1)J_{p,\mathbf{ij}}$$

$$b_{p,\mathbf{ij}} \leftarrow -\mu \bar{h}_{p,\mathbf{j}} \bar{h}^T_{p,\mathbf{i}}$$

$$q_{p,\mathbf{ij}} \leftarrow \frac{1}{\Delta t^2}((\nabla w^n_{\mathbf{i}p})^T \nabla w_{,\mathbf{j}} \mathbf{I} + \nabla w_{,\mathbf{j}} (\nabla w^n_{\mathbf{i}p})^T)$$

$$c_{p,\mathbf{ij}} \leftarrow \mu^N_P \det(\mathbf{F}^n_p) q_{p,\mathbf{ij}}$$

$$k_{p,\mathbf{ij}} \leftarrow \frac{4k_p}{d^2}\bar{k}_{p,\mathbf{i}} \bar{k}^T_{p,\mathbf{j}} + \frac{2k_p}{d}\bar{k}_{p,\mathbf{j}} \bar{k}^T_{p,\mathbf{i}} + \frac{2k_p}{d}\bar{b}^T_{p,\mathbf{i}} \bar{k}_{p,\mathbf{j}} \mathbf{K}_p$$

$$m_{p,\mathbf{ij}} \leftarrow \frac{4m_p}{d^2}\bar{h}_{p,\mathbf{i}} \bar{h}^T_{p,\mathbf{j}} - \frac{2m_p}{d}\bar{h}_{p,\mathbf{j}} \bar{h}^T_{p,\mathbf{i}}$$

$$n_{p,\mathbf{ij}} \leftarrow k_{p,\mathbf{ij}} g_p + k_{p,\mathbf{i}} g^T_{,\mathbf{j}} + g_{p,\mathbf{i}} k^T_{,\mathbf{j}}$$

$$p_{p,\mathbf{ij}} \leftarrow \frac{\mu}{2}(m_{p,\mathbf{ij}} n_p + m_{p,\mathbf{i}} n^T_{,\mathbf{j}} + n_{p,\mathbf{i}} m^T_{,\mathbf{j}} + n_{p,\mathbf{ij}} m_p)$$

$$\Phi_{p,\mathbf{ij}} \leftarrow V_p(p_{p,\mathbf{ij}} - b_{p,\mathbf{ij}} + a_{p,\mathbf{ij}} + c_{p,\mathbf{ij}})$$