Codimensional Non-Newtonian Fluids

Bo Zhu*
Stanford University

Minjae Lee*
Stanford University

Ed Quigley*
Stanford University

Ronald Fedkiw*
Stanford University
Industrial Light + Magic

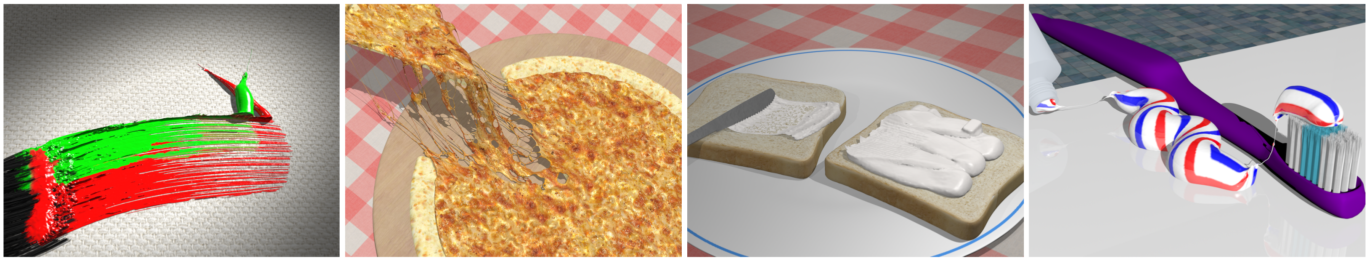

Figure 1: *Examples of codimensional non-Newtonian fluid phenomena. (Far Left) A brush is dragged across paint on a canvas, leaving small furrows in its wake. (Middle Left) Cheese is stretched into thin sheets and filaments as a slice of pizza is removed. (Middle Right) A knife leaves furrows on mayonnaise, then spreads a thin layer onto a slice of bread. (Far Right) Multicolored toothpaste is squeezed out of a tube in a twisting motion.*

Abstract

We present a novel method to simulate codimensional non-Newtonian fluids on simplicial complexes. Our method extends previous work for codimensional incompressible flow to various types of non-Newtonian fluids including both shear thinning and thickening, Bingham plastics, and elastoplastics. We propose a novel time integration scheme for semi-implicitly treating elasticity, which when combined with a semi-implicit method for variable viscosity alleviates the need for small time steps. Furthermore, we propose an improved treatment of viscosity on the rims of thin fluid sheets that allows us to capture their elusive, visually appealing twisting motion. In order to simulate complex phenomena such as the mixing of colored paint, we adopt a multiple level set framework and propose a discretization on simplicial complexes that facilitates the tracking of material interfaces across codimensions. We demonstrate the efficacy of our approach by simulating a wide variety of non-Newtonian fluid phenomena exhibiting various codimensional features.

CR Categories: I.3.3 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation

Keywords: non-Newtonian fluids, codimension

1 Introduction

Non-Newtonian fluids exhibit many different codimensional features that are visually interesting. The furrows made by a brush moving through paint, the thin sheet and filaments of cheese on a pizza, and the thin filaments of toothpaste are just some of the many examples of viscoelastic phenomena to which a codimensional representation is naturally amenable. The special material properties of non-Newtonian fluids make their codimensional motions even more interesting. For example, paint (a shear thinning fluid) has low viscosity at high shear rates making it easy to apply to surfaces using a brush; however, its high viscosity at low shear rates prevents it from running or dripping after being applied. Contrast this behavior with quicksand (a shear thickening fluid) which flows at low shear rates allowing one to sink, but becomes viscous at high shear rates preventing one from climbing out. While non-Newtonian fluids are frequently studied in both computer graphics and computational physics, most state-of-the-art methods focus on modeling them volumetrically, and few address their codimensional features such as thin sheets and filaments.

Motivated by the codimensional fluid simulation of [Zhu et al. 2014], the codimensional solid simulation of [Martin et al. 2010], and the unified framework for simulating fluids and solids in [Macklin et al. 2014], we aim to simulate a wide range of non-Newtonian fluid behaviors especially focusing on codimensional phenomena such as twisting thin films, viscoelastic filaments, and shear thinning or Bingham plastic furrows. We follow the work of [Zhu et al. 2014] for simulating codimensional incompressible flow on simplicial complexes, but provide various novel extensions

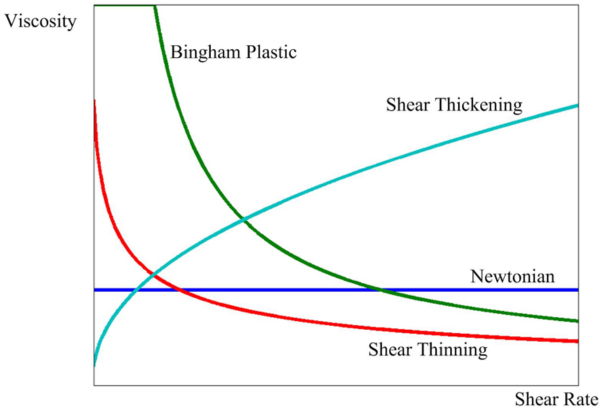

Figure 2: *Plots of viscosity vs. shear rate for various types of fluid.*

*e-mail: {boolzhu,mjgg,equigley,rfedkiw}@stanford.edu

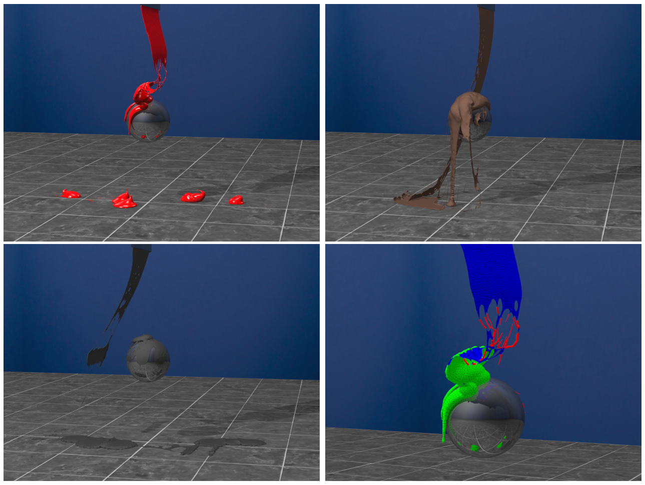

Figure 3: *Shear thinning fluid (top left), shear thickening fluid (top right), and Bingham plastic (bottom left) are poured onto a metal sphere by a source that changes from a volume to a thin sheet to a filament. The underlying simplicial complex (bottom right) is visualized with green tetrahedra, blue triangles, and red segments.*

in order to treat non-Newtonian fluids including shear thinning and thickening fluids, Bingham plastics, and elastoplastics. See Figure 2.

Unlike in [Zhu et al. 2014] where fluids with high Reynolds numbers had most of their codimensional features created by strong surface tension forces, our work focuses on non-Newtonian fluids with low Reynolds numbers where the codimensional features are mostly created by shear dependent viscosity and solid fluid contact. Furthermore, since non-Newtonian fluids can exhibit both solid and fluid behavior, we propose a more general framework that can simulate codimensional materials with properties that bridge the gap between solids and fluids. This includes a new semi-implicit time integration scheme for elasticity that uses a smoothing operator, a semi-implicit treatment for variable viscosity incompressible flows, and an improved treatment of viscosity on the rims of thin fluid sheets allowing for visually appealing twisting motions. In addition, a multiple material level set tracking technique is proposed for simplicial complexes.

2 Related Work

Researchers have simulated non-Newtonian flow using a variety of methods including grid based methods [Goktekin et al. 2004; Losasso et al. 2006; Batty and Bridson 2008], volumetric mesh-based methods [Bargteil et al. 2007; Wojtan and Turk 2008; Wojtan et al. 2009; Wicke et al. 2010], particle-based methods [Clavet et al. 2005; Paiva et al. 2009; Gerszewski et al. 2009; Zhou et al. 2013], and hybrid particle-grid methods [Stomakhin et al. 2014]. However, most of the results produced by these methods are restricted to volumetric phenomena. Few of the interesting codimensional features are captured, such as the furrows left by a brush in paint or filaments that extrude from a volume. Some work has been done on this front. [Wojtan and Turk 2008] proposed a method to embed a high resolution surface mesh in a tetrahedral FEM simulator to get thin features, and [Batty and Houston 2011] used an adaptive tetrahedral mesh to achieve realistic buckling and coiling of viscous fluids. However, a codimensional approach is more computationally efficient for capturing arbitrarily thin films and filaments, especially those that emanate from volumetric regions as material thins and breaks apart (as in Figure 1, middle left).

Many complex fluid and solid phenomena that exist in high codimensions have been studied in recent graphics research. These include codimension-2 materials such as paint jets [Lee et al. 2006] and viscous threads [Bergou et al. 2010], as well as codimension-1 materials such as viscous fluid sheets [Batty et al. 2012]. One major application area of fluid simulation in high codimensions is painting [Baxter et al. 2001; Baxter et al. 2004]. Much research has been devoted to creating systems that simulate paint phenomena that occur in high codimensions, such as paint jets [Lee et al. 2006], streaks [Chu et al. 2010; DiVerdi et al. 2013], and color dispersion [Chu and Tai 2005]. These systems produce realistic painting results and support real-time user interaction. The paint simulation in most of these systems relies on solving the shallow water equations on a surface [Curtis et al. 1997; Wang et al. 2005]. While this non-volumetric representation enables real-time simulations, it leaves many interesting effects unexplored. For example, dragging a brush through a glob of paint produces distinctive "furrows," which arise due to the shear thinning properties of the paint.

3 Material Models

Consider the Lagrangian form of the generalized incompressible Navier Stokes equations

$$\rho\frac{D\vec{u}}{Dt} + \nabla p = \nabla\cdot(\boldsymbol{\sigma_v} + \boldsymbol{\sigma_e}) + \vec{f}, \tag{1}$$

where ρ is density, $\vec{u}$ is velocity, t is time, p is pressure, $\boldsymbol{\sigma_v}$ is the viscous stress tensor, $\boldsymbol{\sigma_e}$ is the elastic stress tensor, and $\vec{f}$ represents other forces such as gravity, surface tension, and friction.

The viscous stress tensor for incompressible flow is $\boldsymbol{\sigma}_v = \mu\dot{\boldsymbol{\gamma}}$ where μ is the viscosity and $\dot{\boldsymbol{\gamma}} = \nabla\vec{u} + (\nabla\vec{u})^T$ is the strain rate tensor. Non-Newtonian fluids are typically modeled using an approximation to the shear rate $\dot{\gamma}$, and we use the Frobenius norm $\dot{\gamma} = \sqrt{\dot{\boldsymbol{\gamma}} : \dot{\boldsymbol{\gamma}}}$. Some of our examples also consider the dependence of viscosity on temperature, and in these cases we simply multiply μ by e^{-cT} where c is a constant [Reynolds 1886]. This requires the solution of an auxiliary heat equation $\rho DT/Dt = \nabla\cdot(\kappa\nabla T)$,

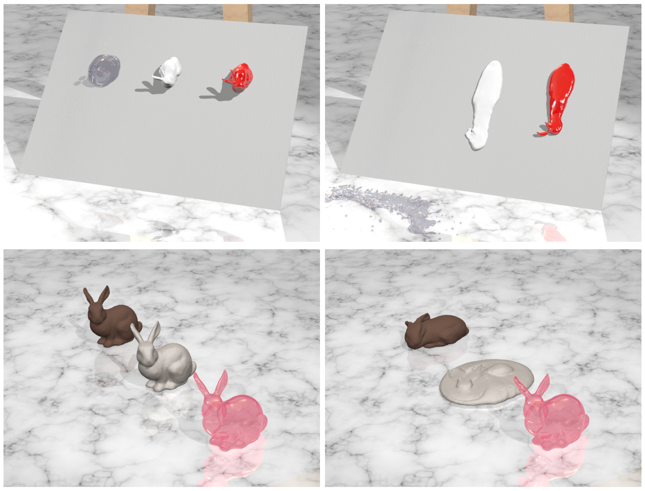

Figure 4: *(Top) A low viscosity Newtonian bunny, a high viscosity Newtonian bunny, and a shear thinning bunny drip down a slope. The shear thinning "paint" bunny stops flowing when its shear rate becomes small. (Bottom) A shear thickening bunny, a Bingham plastic bunny, and an elastic bunny fall onto the ground. The shear thickening fluid rigidifies upon impact, but then begins to flow as its shear rate decreases. The Bingham plastic "clay" bunny deforms as it hits the ground, but is rigid thereafter.*

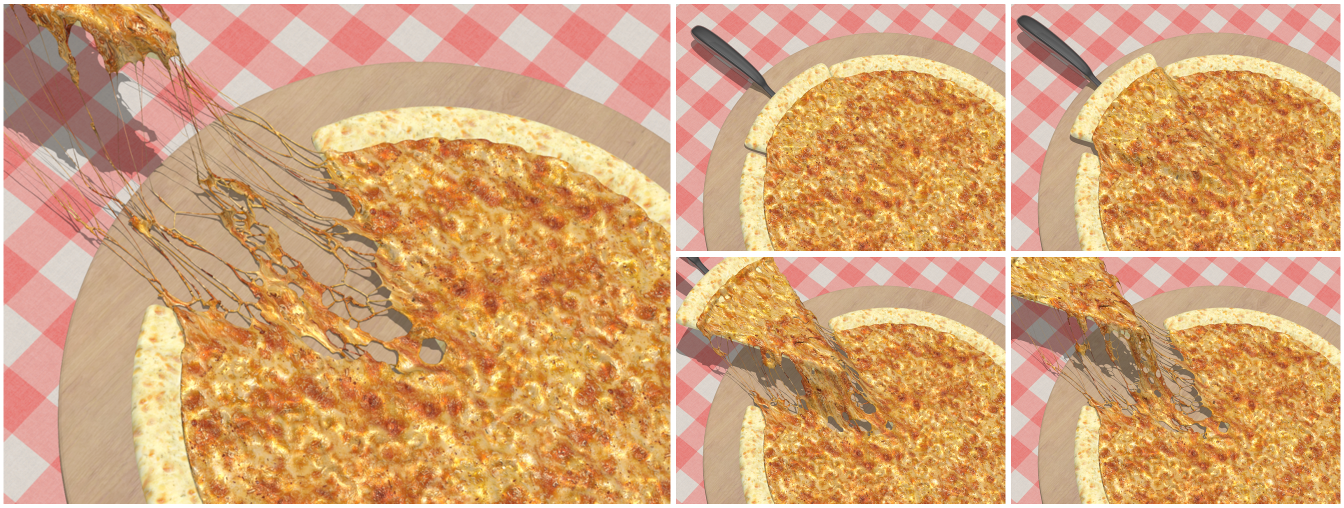

Figure 5: *Pulling out a slice of pizza causes the cheese to stretch along the boundaries of the slice, creating a web-like collection of viscoelastic films and filaments as it becomes very thin.*

where κ is the thermal conductivity. The heat equation is solved using a backward Euler implicit time discretization using the same codimensional simplical complex discretization scheme [Zhu et al. 2014] that is used for pressure in Equation 1 (see Section 4).

We use the Carreau-Yasuda model [Carreau 1972; Yasuda 1979] to determine the viscosity for both shear thinning and thickening fluids. The relationship between shear rate and viscosity can be written as

$$\mu(\dot{\gamma}) = \mu_\infty + (\mu_0 - \mu_\infty)(1 + (\Lambda\dot{\gamma})^\alpha)^{\frac{n-1}{\alpha}}, \tag{2}$$

where Λ scales the shear rate and α governs smoothness of transitions. When $n = 1$, we recover a Newtonian fluid with viscosity μ_0. When the shear rate is zero, we also recover $\mu = \mu_0$ illustrating that μ_0 governs the viscosity of both shear thickening and shear thinning fluids under low shear rate. As can be seen from looking at the left hand side of Figure 2, this a relatively high viscosity for shear thinning fluids stopping paint from dripping and a relatively low viscosity for shear thickening fluids allowing one to sink into quicksand. For shear thinning fluids, $n < 1$ and the second term approaches zero forcing the viscosity to asymptotically approach μ_∞ at high shear rates. As can be seen in Figure 2, setting $\mu_\infty < \mu_0$ for shear thinning fluids allows paint to be easily applied by a brush as brush strokes create shear. For shear thickening fluids, $n > 1$ and the viscosity increases as the shear rate increases with the amount of increase scaled by $\mu_0 - \mu_\infty$.

A Bingham plastic acts as a rigid body at low shear rates, but acts as a viscous fluid after its stress exceeds a critical threshold $\dot{\gamma_c}$. In practice, the rigid behavior can be modeled simply by using a large viscosity. Thus, we use the model of [Beverly and Tanner 1992],

$$\mu(\dot{\gamma}) = \begin{cases} \mu_0, & \dot{\gamma} \leq \dot{\gamma_c} \\ \mu_\infty + \tau_0/\dot{\gamma}^q, & \dot{\gamma} > \dot{\gamma_c} \end{cases} \tag{3}$$

where in the typical model $\tau_0 = \dot{\gamma_c}(\mu_0 - \mu_\infty)$ makes the viscosity continuous. However, we add the extra parameter q to provide more animator control.

We model the elastic stress tensor as $\boldsymbol{\sigma_e} = \mu_e \boldsymbol{\varepsilon}$ where μ_e is constant and $\boldsymbol{\varepsilon}$ is the elastic strain tensor [Goktekin et al. 2004; Losasso et al. 2006]. In order to capture rotation, we use the Green strain $\boldsymbol{\varepsilon}_G = \frac{1}{2}(\boldsymbol{F}_e^T \boldsymbol{F}_e - \boldsymbol{I})$, where $\boldsymbol{F}_e$ is the elastic component of the deformation gradient $\boldsymbol{F}$. We use the multiplicative plasticity model of [Irving et al. 2004; Stomakhin et al. 2013] to separate $\boldsymbol{F}$ into its elastic and plastic components, $\boldsymbol{F} = \boldsymbol{F}_e \boldsymbol{F}_p$, in which the part of $\boldsymbol{F}_e$ that exceeds a critical deformation threshold is absorbed into $\boldsymbol{F}_p$. $\boldsymbol{F}$ is evolved as $D\boldsymbol{F}/Dt = \nabla\vec{u}\boldsymbol{F}$ as in [Stomakhin et al. 2013].

4 Temporal Evolution

Following [Zhu et al. 2014], we use particles connected into simplicial complexes to represent non-Newtonian fluids in different codimensions. Tetrahedra, triangles, segments, and points are used to model fluid volumes, thin sheets, narrow filaments, and small droplets respectively. As before, particles store their mass, position, velocity, thickness, and connectivity, but now they also store their temperature, deformation gradient, and information for computing their viscosity.

Each time step, we advect the particles, update the mesh topology and thicknesses, compute the viscosity using the relevant model, and solve the heat equation for the temperature. Then Equation 1 is solved in a time split fashion. First the external forces such as gravity are integrated explicitly, e.g. see [Zhu et al. 2014] for the treatment of surface tension and Section 7 for the treatment of friction. Then we further update the velocity using a semi-implicit discretization of elasticity as outlined in Section 6 followed by a semi-implicit treatment of viscosity as outlined in Section 5. Finally we enforce incompressibility by solving a Poisson equation for pressure $\nabla \cdot (\Delta t/\rho \nabla p) = \nabla \cdot \vec{u}$, which is used to obtain our final velocity field $\vec{u}^{n+1} = \vec{u} - \Delta t/\rho \nabla p$.

We use the codimensional Poisson solver of [Zhu et al. 2014] to solve for temperature, pressure, and the implicit part of viscosity and elasticity in Equations 5 and 10 (see Sections 5 and 6). The degrees of freedom are placed on vertices of tetrahedra, barycenters of triangles, and centers of segments. We interpolate attributes carried on particles from the particles to the degrees of freedom to avoid issues with decoupling. The volume-weighted gradient is defined at each particle n as

$$W_n \nabla p = \left(V_n \nabla + \lambda_n A_n \nabla + \frac{\pi \lambda_n^2}{4} L_n \nabla \right) p, \tag{4}$$

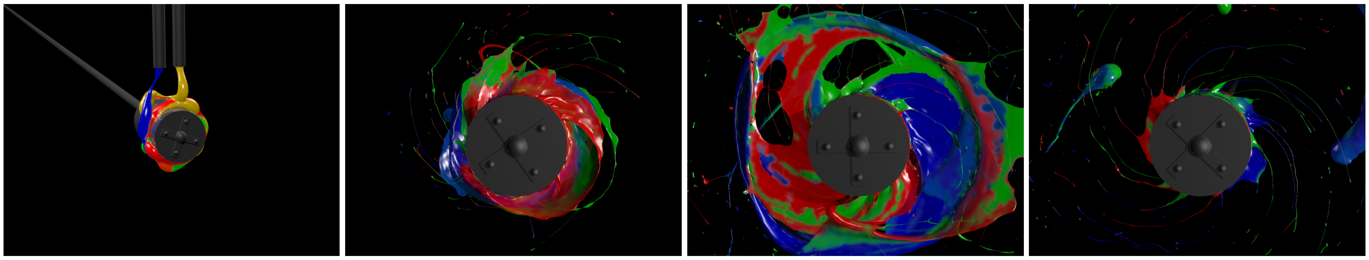

Figure 6: *Multiple colors of paint are poured onto a rotating cylinder. When the cylinder spins faster it creates a spray of paint. This spray consists of many filaments and droplets, and demonstrates color mixing through the evolution of multiple level sets across codimensions.*

where $V_n\nabla$ represents the volume-weighted gradient contribution from incident tetrahedra, $A_n\nabla$ represents the area-weighted gradient contribution from incident triangles, $L_n\nabla$ represents the length-weighted gradient contribution from incident segments, λ is the particle thickness, and $W_n = V_n + \lambda_n A_n + (\pi\lambda_n^2/4)L_n$ is the total control volume for particle n. After solving the sparse linear system, the results are interpolated back to the particles using the PIC-FLIP scheme from [Zhu and Bridson 2005].

5 Viscosity

Following [Rasmussen et al. 2004], we take a semi-implicit approach to viscosity splitting the shear rate tensor into an explicit part $\mu\nabla\vec{u}^T$ and an implicit part $\mu\nabla\vec{u}$. After adding in the explicit terms to obtain a new intermediate velocity $\vec{u}^*$, the implicit terms are integrated via

$$\vec{u}^{**} = \vec{u}^* + \frac{\Delta t}{\rho}\nabla\cdot\mu\nabla\vec{u}^{**}, \qquad (5)$$

where $\vec{u}^{**}$ includes the effects of viscosity.

The discretized matrix form of Equation 5 can be written as

$$(\boldsymbol{W} + \frac{\Delta t}{\rho}\boldsymbol{G}^T\hat{\boldsymbol{W}}^{-1}\boldsymbol{G})\vec{u}^{**} = \boldsymbol{W}\vec{u}^*, \qquad (6)$$

where $\boldsymbol{W}$ is a diagonal matrix of control volumes W_n for each particle defined in Section 4, $\hat{\boldsymbol{W}}^{-1}$ is a diagonal matrix with an entry μ/W_n for each particle, $\mathbf{G}$ is the volume weighted gradient matrix defined by Equation 4, and $\mathbf{G}^T$ is the negative volume weighted divergence matrix. Note that we have assumed a spatially constant density when applying the viscous forces.

Thin fluid sheets may exhibit complex twisting motions such as in [Oefner 2013]. This occurs when one section of the thin sheet wraps above/below another section creating a twisting spray of filaments and droplets. In order to simulate this twisting motion, we consider the forces on the rim of a thin sheet. The motion of the rim is dominated by the viscosity force, the surface tension force, and the incoming momentum flux from the thin sheet [Savva 2007]. Because the rim thickness is much greater than the thickness of the film interior, viscosity on the rim comes largely from fluid also on the rim, while the viscosity from the interior of the film is negligible. We incorporate these two observations into our discretization by adding extra degrees of freedom on the center of the rim segments and by removing the degrees of freedom on the triangles incident to rim segments when computing viscosity. These modifications achieve two ends. First, the rim segments accurately represent the geometry of the rim as a cylindrical tube, and therefore accurately compute the contribution of the viscosity force from fluid on the rim. Second, the degrees of freedom on the rim are decoupled from the degrees of freedom on the interior of the thin sheet, thus ignoring drag from the fluid on the thin sheet. With these modifications we are able to reproduce the effect of the twisting thin film (Figure 7).

6 Elasticity

Our semi-implicit time integration scheme for elasticity is motivated by [Smereka 2003; Xu and Zhao 2003], in which a diffusion term is introduced to convert an explicit scheme into a semi-implicit scheme for increased stability in simulating evolving interfaces using mean curvature flow. A similar idea is used by [Zheng et al. 2006] for semi-implicit surface tension.

First consider implicit time integration for elasticity using the Cauchy strain $\boldsymbol{\varepsilon}_c = \boldsymbol{F}_s - \boldsymbol{I}$ where $\boldsymbol{F}_s = \frac{1}{2}(\boldsymbol{F}_e + \boldsymbol{F}_e^T)$ is the symmetric part of $\boldsymbol{F}_e$:

$$\vec{u}^{**} = \vec{u}^* + \frac{\Delta t\mu_e}{\rho}\nabla\cdot(\boldsymbol{F}_s^{**} - \boldsymbol{I}). \qquad (7)$$

If we make the aggressive approximation that $\boldsymbol{F}_s$ obeys the same rule for time evolution as $\boldsymbol{F}_e$, i.e. $D\boldsymbol{F}_s/Dt \approx \nabla\vec{u}\boldsymbol{F}_s$, then we

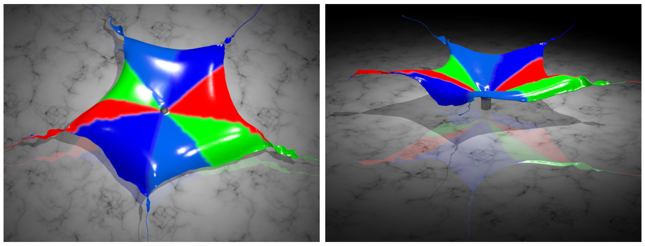

Figure 7: *A "paint fountain" sprays multicolored paint radially from a source. The combination of viscosity and surface tension forces create twisting effects along the rim.*

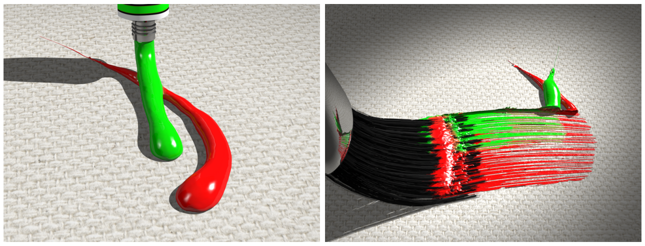

Figure 8: *(Left) Two different colors of paint are applied to a canvas. (Right) A brush drags the paint along the canvas, leaving furrows in its wake.*

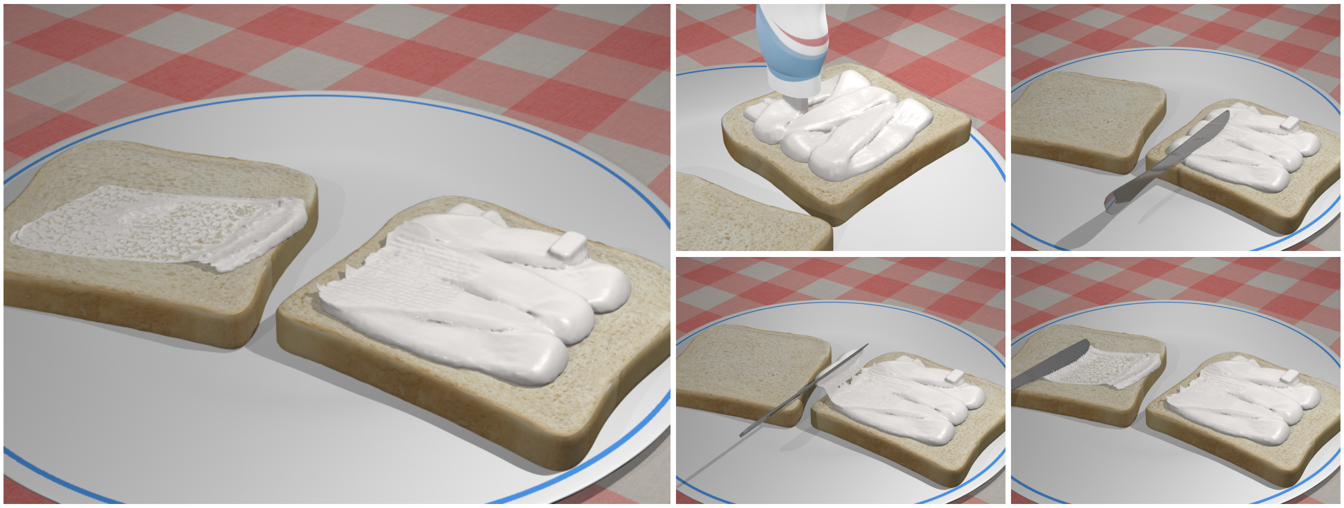

Figure 9: *Mayonnaise, modeled as a Bingham plastic, flows when it is initially poured onto a slice of bread but maintains its shape afterwards. A knife leaves furrows when it passes through the mayonnaise volume, then spreads a thin sheet onto the second piece of bread.*

may write

$$\vec{u}^{**} \approx \vec{u}^{*} + \frac{\Delta t \mu_e}{\rho} \nabla \cdot ((\boldsymbol{I} + \Delta t \nabla \vec{u}^{**}) \boldsymbol{F}_s^{*} - \boldsymbol{I}) \qquad (8)$$

where we have integrated $D\boldsymbol{F}_s/Dt \approx \nabla \vec{u} \boldsymbol{F}_s$ implicitly in $\vec{u}$ but explicitly in $\boldsymbol{F}_s$. Then from this one obtains

$$\vec{u}^{**} \approx \vec{u}^{*} + \frac{\Delta t \mu_e}{\rho} \nabla \cdot (\boldsymbol{\varepsilon}_c^{*} + \Delta t \nabla \vec{u}^{**} \boldsymbol{F}_s^{*}), \qquad (9)$$

where $\vec{u}^{*} + \frac{\Delta t \mu_e}{\rho} \nabla \cdot \boldsymbol{\varepsilon}_c^{*}$ is an explicit integration. This motivates the fact that a term of the form $\Delta t \nabla \vec{u} \boldsymbol{F}_s$ would be useful as a smoothing operator in implicitly integrating the Cauchy strain $\boldsymbol{\varepsilon}_c$. Thus we will use $S(\vec{u}) = \beta \Delta t \nabla \vec{u} \boldsymbol{F}_s$ as a smoothing operator for our Green strain, where β is a relaxation factor that controls the damping that the smoothing operator introduces into the system. In practice we use values of β between 0 and 1, where the integration is purely explicit when $\beta = 0$.

With this definition of $S(\vec{u})$, we can split the Green strain tensor into an explicit part $\boldsymbol{\varepsilon} - S(\vec{u})$ and an implicit part $S(\vec{u})$. Then after adding in the explicit terms to get a new intermediate velocity $\vec{u}^{*}$, the implicit terms are integrated via

$$\vec{u}^{**} = \vec{u}^{*} + \frac{\Delta t \mu_e}{\rho} \nabla \cdot S(\vec{u}^{**}), \qquad (10)$$

where $\vec{u}^{**}$ then includes the full effects of elasticity. Substituting the volume weighted divergence and gradient into Equation 10, we get the discretized matrix form

$$(\boldsymbol{W} + \frac{\mu_e \beta \Delta t^2}{\rho} \boldsymbol{G}^T \tilde{\boldsymbol{W}}^{-1} \boldsymbol{G}) \vec{u}^{**} = \boldsymbol{W} \vec{u}^{*}, \qquad (11)$$

where $\tilde{\boldsymbol{W}}^{-1}$ is a block tri-diagonal matrix with a block entry $\boldsymbol{F}_s / W_n$ for each particle. We solve Equation 11 for the three components of velocity independently by taking advantage of the codimensional Poisson solver already used for incompressibility and viscosity.

7 Solid Fluid Interactions

The distinctive behaviors of non-Newtonian fluids often arise due to shear induced by contact with a solid body. It is therefore important to pay special attention to the treatment of solid-fluid interactions. The friction force between fluids and kinematic solids is modeled using the technique of [Bridson et al. 2002; Zhu and Bridson 2005]. However, we note that friction and viscosity have the same underlying cause, namely the Brownian motion tangential to the macroscopic velocity of the particles in a continuum. Therefore, whereas [Bridson et al. 2002] uses a constant coefficient of friction, we scale the coefficient of friction based on the local viscosity of the fluid contacting the solid. This allows the less viscous bunny in Figure 4 (top) to slide off the incline plane, while the more viscous bunny sticks to the surface.

We model paint brush fibers as a collection of segments using a mass-spring model with linear springs and bending springs [Selle et al. 2008]. When fibers collide with elements of the fluid simplicial complex, we model the fiber-fluid interaction by applying explicit forces in an immersed boundary fashion, similar to the bubble-air interaction in [Zhu et al. 2014]. This interaction, together with friction from the canvas and our codimensional non-Newtonian fluid solver, naturally models the codimensional transitions from volumetric elements to thin furrows that occur when a brush is dragged through paint (see Figure 11). To produce highly detailed furrows, we also model the normal pressure force exerted by fibers on codimension-1 fluid elements that lay on the canvas. This is done by adding a target divergence term [Losasso et al. 2008] scaled by the fluid volume displaced by the fiber to the incompressible solver in order to push the fluid apart. Finally, we use a diffusion process to push any remaining volume of a particle still in contact with the fiber to its neighbors using Gauss-Seidel iterations to obtain an accurate furrow shape.

8 Adaptive Meshing

We extend the codimensional meshing algorithm of [Zhu et al. 2014] by adding a boundary vertex snap operation similar to [Da et al. 2014] to support the topological merging of elements across different codimensions. When the distance between a pair of non-neighboring vertices falls below a threshold, they are connected as a single vertex. To facilitate adaptive meshing, we store a meshing length scale on each particle. In each meshing operation, the length threshold to collapse or split an edge is defined by the average of meshing length scales from incident particles. We gain adaptivity by varying these per-particle length scales. This can be done using a variety of strategies in order to place degrees of freedom near visually interesting regions, e.g., by decreasing the length scale for particles on high codimension elements as in Figure 5 or for parti-

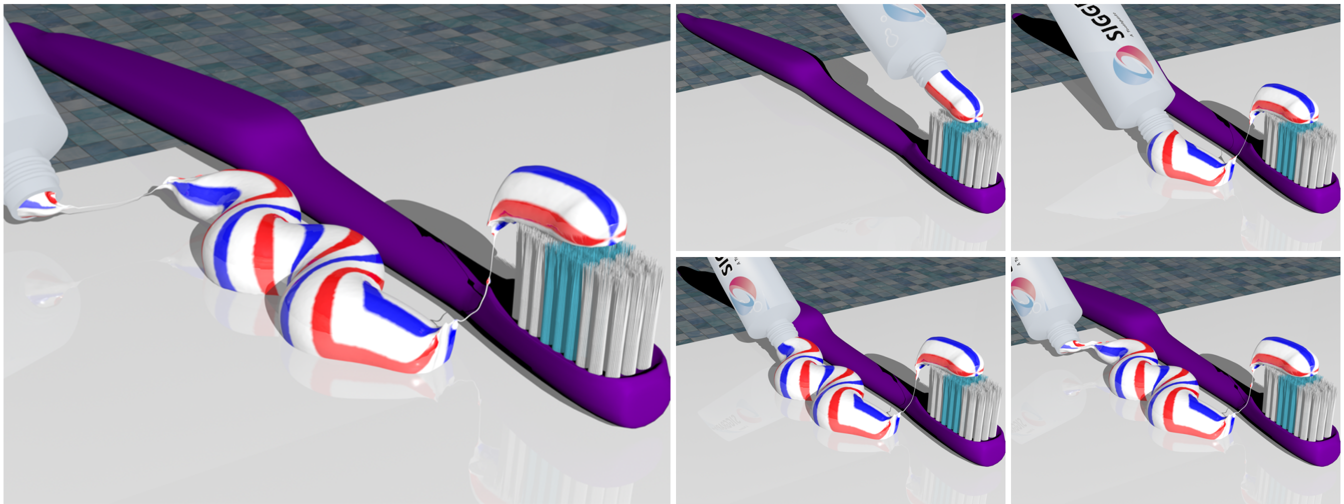

Figure 10: *Toothpaste, a Bingham plastic, is squeezed out in a twisting motion onto a toothbrush and a counter. Its colors are modeled using multiple level sets. The toothpaste exhibits codimensional features where the volumes are pulled apart.*

cles near solids as in Figure 9.

9 Material Interfaces

We follow the method of [Losasso et al. 2006] maintaining a vector of level set values $\vec{\phi}$ on each particle, where each component of the vector corresponds to a different material. Exactly one component of each particle's $\vec{\phi}$ is negative denoting that the particle is made of the corresponding material. After advection, we perform fast marching on the simplical complexes connecting the particles to update the signed distance values of $\vec{\phi}$. On thin sheets, we perform fast marching on triangles using the method of [Kimmel and Sethian 1998]. For volumetric elements, we approximate the interface using linear interpolation along edges to compute triangular faces on the interior of interface tetrahedra. After initializing the interface, we march outwards through particles on adjacent tetrahedra, roughly approximating signed distance as the smallest Euclidean distance to an interface triangle.

Although one could use $\vec{\phi}$ to determine various material properties, our examples rely on $\vec{\phi}$ merely to indicate the colors of various materials. Thus, we can increase efficiency and versatility by computing $\vec{\phi}$ in a postprocessing step allowing us to generate many different coloring combination options while running the primary simulation only once. This is facilitated by tracking the parents of each newly created particle during the simulation. A particle's parents are the set of one or more particles from the previous time step that correspond to the particle in the current time step. It is possible to have more than one parent in the case of mesh collapsing operations. A set of weights are also computed and stored to record the relative influence of each parent on its newly created child [Yu et al. 2012; Bojsen-Hansen et al. 2012].

As in [Zhu et al. 2014], we generate a skinned mesh in a postprocessing step using the simulation degrees of freedom. To render a broader range of materials, we copy $\vec{\phi}$ as well as texture coordinates from the simulation particles to their corresponding skinned mesh nodes. These values are interpolated when new skinned mesh nodes are created during subdivision, so we also correct $\vec{\phi}$ on the skinned mesh using the projection method of [Losasso et al. 2006].

In order to gain an accurate representation of paint for rendering, we use a shader based on the reflectance model for pigments proposed by [Haase and Meyer 1992] and used in painting applications by [Baxter et al. 2004]. This model describes the relationship $\sum_{i\in P}(K_iC_i)/(S_iC_i) = (1-R_\infty)^2/(2R_\infty)$, where P is the set of pigments in the material, K is pigment absorption, C is pigment concentration, S is pigment scattering, and R_∞ is reflectance. Thus we can derive the reflectance of any pigmented solution given its pigment concentrations, absorptions, and scattering parameters. We use the spectral data provided by [Okumura 2005] and [Smits 1999] for our paint renderings. This enables us to perform mixing between different paint colors, as seen in Figures 6 and 7.

10 Examples

We simulate the cheese on a pizza using a shear thinning viscosity model along with an elasticity term (see Figure 5), with parameters μ_0=2, μ_∞=4, Λ=1, α=1, n=.4, μ_e=1.2×10^3, and β=.2. The initial codimensional mesh used to represent the cheese is composed of tetrahedra and triangles. The cheese is stretched into viscoelastic thin sheets and a web of filaments as the slice is pulled away.

In Figure 9, we model mayonnaise as a Bingham plastic (μ_0=1, μ_∞=1×10^3, $\dot{\gamma}_c$=1, and q=2) in order to capture its rigid behavior at low shear rates. Mayonnaise flows from a source but holds its shape after settling. Later, a knife scoops up some mayonnaise volume. By adaptively remeshing volumetric elements near the knife, we are able to see furrows left in the mayonnaise by the knife's serrated edge. The knife then spreads mayonnaise as a thin sheet.

In Figure 10, a tube squeezes toothpaste onto a toothbrush, then continues squeezing toothpaste onto the counter. Multiple level sets are used to capture the interfaces between the toothpaste's different colors. The tube makes sharp motions to pull apart volumetric sections of the fluid, creating thin sheets and filaments. Because toothpaste is a Bingham plastic it is able to flow from the tube initially, then hold its shape after settling. The parameters for this example are μ_0=1, μ_∞=2×10^3, $\dot{\gamma}_c$=20, and q=1.

We simulate paint using a shear thinning model. In Figure 6, we pour four different colors of paint onto a rotating cylinder. We then accelerate the cylinder's rotation to create a spray consisting of various thin sheets, filaments, and droplets. Colors are mixed across codimensions through the evolution of multiple level sets. This example was run with parameters μ_0=.2, μ_∞=20, Λ=.05, α=.8, and n=.2. In Figure 8, we apply two globs of paint to a canvas, then

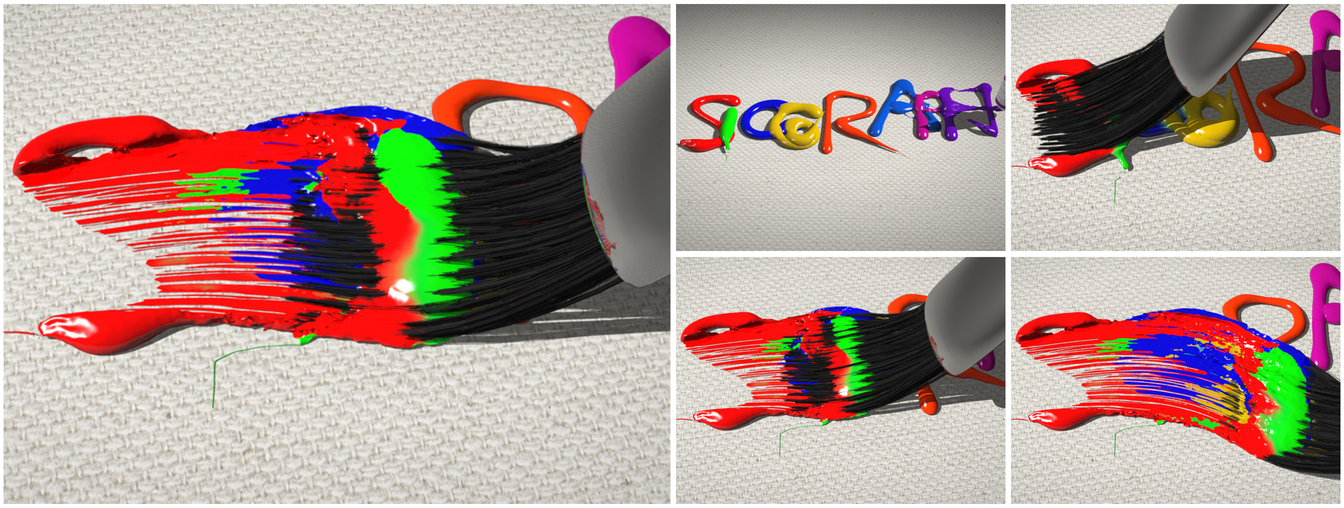

Figure 11: *A paint brush traces furrows through a set of letters made of paint.*

Example	No. Elements (thousands)				Time	Percentage	
	Par	Tet	Tri	Seg	(sec)	Mesh	Solve
Pizza	11	31	2.8	.1	67	6	94
Mayo	21	96	15	0	35	25	75
Toothpaste	37	205	.046	.022	208	23	77
Fountain	3.8	0	7.2	.07	1	41	58
Cylinder	22	23	17	3	20	26	74
Brush	207	89	88	33	1506	11	89
Letters	117	148	85	42	1523	32	68

Table 1: *This table gives the average number of particles, tetrahedra, triangles, and segments per frame for each example. The rightmost columns give the average time per frame, along with the percentage of time spent meshing and time spent on solving for forces and incompressibility.*

drag a brush through both. A series of furrows, modeled using thin sheets and segments, are left in the wake of the brush. In Figure 11, we use paints with eight different colors to spell a word on a canvas, then drag a brush through the letters. The interfaces between colors persist when different letters merge together, demonstrating our multiple level set method for tracking material interfaces. Both of these paint brush simulations were run with parameters μ_0=2, μ_∞=40, Λ=1, α=1, and n=.4.

We show the number of particles and elements in different codimensions and the runtimes of these examples in Table 1. The bottleneck of each simulation is the Poisson solver which is used for viscosity, elasticity (where applicable), and incompressibility. The fraction of time spent on meshing varies by examples. The cost of meshing is greater for examples with large changes in volumetric elements or with many codimensional transitions (e.g., the letter and cylinder examples).

We ran additional simulations to compare our semi-implicit elasticity scheme with fully implicit (see Appendix) and explicit schemes. We found that all three produced visually similar results for different values of μ_e using a CFL of 1. For a CFL of 10 the semi-implicit scheme produces smooth results for $\beta\geq$.2, while the semi-implicit scheme with β=.1 exhibits some artifacts and the explicit scheme does not converge.

11 Limitations and Future Work

One limitation of our method is that it does not accurately account for rotational motion in codimension-2 or bending motion in codimension-1. By taking these factors into consideration, one could begin to produce phenomena that exhibit viscous thin thread coiling and thin sheet bending. For example, pouring honey is an inherently codimensional effect involving thin filaments and thin sheets spontaneously coiling and bending, stacking on themselves, and sinking into a volume. Some interesting work in this vein can be seen in [Batty and Bridson 2008] and [Bergou et al. 2010]. Another limitation of our method is that the explicit, semi-implicit, and implicit time integration schemes may not produce exactly the same results since the schemes introduce different amounts of damping.

A useful extension to our method would be a temporally coherent treatment of the flows it models: since most of the interesting aspects of non-Newtonian fluid phenomena occur under low Reynolds numbers, even minor topology changes in a simplicial complex can create artifacts in rendering. This presents a challenging problem in meshing for simplicial complexes. Additionally, work remains to be done on the topic of two-way coupling between codimensional non-Newtonian fluids and thin brush fibers.

Acknowledgements

Research is supported in part by ONR N00014-13-1-0346, ONR N-00014-11-1-0027, ONR N00014-11-1-0707, ARL AHPCRC W911NF-07-0027, NSF CSR-1409847, and the Intel Science and Technology Center for Visual Computing. Computing resources were provided in part by ONR N00014-05-1-0479.

References

Bargteil, A. W., Wojtan, C., Hodgins, J. K., and Turk, G. 2007. A finite element method for animating large viscoplastic flow. *ACM Trans. Graph. (SIGGRAPH Proc.) 26*, 3.

Batty, C., and Bridson, R. 2008. Accurate viscous free surfaces for buckling, coiling, and rotating liquids. In *Proceedings of the 2008 ACM SIGGRAPH/Eurographics symposium on computer animation*, Eurographics Association, 219–228.

Batty, C., and Houston, B. 2011. A simple finite volume method for adaptive viscous liquids. In *Proceedings of the 2011*

ACM SIGGRAPH/Eurographics Symposium on Computer Animation, ACM, 111–118.

BATTY, C., URIBE, A., AUDOLY, B., AND GRINSPUN, E. 2012. Discrete viscous sheets. *ACM Trans. Graph. (SIGGRAPH Proc.) 31*, 4, 113.

BAXTER, B., SCHEIB, V., LIN, M. C., AND MANOCHA, D. 2001. Dab: interactive haptic painting with 3d virtual brushes. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, ACM, 461–468.

BAXTER, W., WENDT, J., AND LIN, M. C. 2004. Impasto: a realistic, interactive model for paint. In *Proceedings of the 3rd international symposium on Non-photorealistic animation and rendering*, ACM, 45–148.

BERGOU, M., AUDOLY, B., VOUGA, E., WARDETZKY, M., AND GRINSPUN, E. 2010. Discrete viscous threads. *ACM Trans. Graph. (SIGGRAPH Proc.) 29*, 4, 116.

BEVERLY, C., AND TANNER, R. 1992. Numerical analysis of three-dimensional bingham plastic flow. *Journal of non-newtonian fluid mechanics 42*, 1, 85–115.

BOJSEN-HANSEN, M., LI, H., AND WOJTAN, C. 2012. Tracking surfaces with evolving topology. *ACM Trans. Graph. 31*, 4, 53.

BRIDSON, R., FEDKIW, R., AND ANDERSON, J. 2002. Robust treatment of collisions, contact and friction for cloth animation. *ACM Trans. Graph. (SIGGRAPH Proc.) 21*, 3, 594–603.

CARREAU, P. J. 1972. Rheological equations from molecular network theories. *Transactions of The Society of Rheology (1957-1977) 16*, 1, 99–127.

CHU, N. S.-H., AND TAI, C.-L. 2005. Moxi: real-time ink dispersion in absorbent paper. In *ACM Trans. Graph. (SIGGRAPH Proc.)*, vol. 24, ACM, 504–511.

CHU, N., BAXTER, W., WEI, L.-Y., AND GOVINDARAJU, N. 2010. Detail-preserving paint modeling for 3d brushes. In *Proceedings of the 8th International Symposium on Non-Photorealistic Animation and Rendering*, ACM, 27–34.

CLAVET, S., BEAUDOIN, P., AND POULIN, P. 2005. Particle-based viscoelastic fluid simulation. In *Proc. of the 2004 ACM SIGGRAPH/Eurographics Symp. on Comput. Anim.*, ACM Press, 219–228.

CURTIS, C. J., ANDERSON, S. E., SEIMS, J. E., FLEISCHER, K. W., AND SALESIN, D. H. 1997. Computer-generated watercolor. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, 421–430.

DA, F., BATTY, C., AND GRINSPUN, E. 2014. Multimaterial mesh-based surface tracking. *ACM Trans. Graph. 33*, 4, 112:1–112:11.

DIVERDI, S., KRISHNASWAMY, A., MECH, R., AND ITO, D. 2013. Painting with polygons: A procedural watercolor engine. *Visualization and Computer Graphics, IEEE Transactions on 19*, 5, 723–735.

GERSZEWSKI, D., BHATTACHARYA, H., AND BARGTEIL, A. W. 2009. A point-based method for animating elastoplastic solids. In *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '09, 133–138.

GOKTEKIN, T. G., BARGTEIL, A. W., AND O'BRIEN, J. F. 2004. A method for animating viscoelastic fluids. *ACM Trans. Graph. (SIGGRAPH Proc.) 23*, 463–468.

HAASE, C. S., AND MEYER, G. W. 1992. Modeling pigmented materials for realistic image synthesis. *ACM Trans. Graph. 11*, 4, 305–335.

IRVING, G., TERAN, J., AND FEDKIW, R. 2004. Invertible finite elements for robust simulation of large deformation. In *Proc. of the ACM SIGGRAPH/Eurographics Symp. on Comput. Anim.*, 131–140.

KIMMEL, R., AND SETHIAN, J. A. 1998. Computing geodesic paths on manifolds. *Proceedings of the National Academy of Sciences 95*, 15, 8431–8435.

LEE, S., OLSEN, S. C., AND GOOCH, B. 2006. Interactive 3d fluid jet painting. In *Proceedings of the 4th international symposium on Non-photorealistic animation and rendering*, ACM, 97–104.

LOSASSO, F., SHINAR, T., SELLE, A., AND FEDKIW, R. 2006. Multiple interacting liquids. *ACM Trans. Graph. (SIGGRAPH Proc.) 25*, 3, 812–819.

LOSASSO, F., TALTON, J., KWATRA, N., AND FEDKIW, R. 2008. Two-way coupled SPH and particle level set fluid simulation. *IEEE TVCG 14*, 4, 797–804.

MACKLIN, M., MÜLLER, M., CHENTANEZ, N., AND KIM, T.-Y. 2014. Unified particle physics for real-time applications. *ACM Trans. Graph. (SIGGRAPH Proc.) 33*, 4, 153:1–153:12.

MARTIN, S., KAUFMANN, P., BOTSCH, M., GRINSPUN, E., AND GROSS, M. 2010. Unified simulation of elastic rods, shells, and solids. *ACM Trans. Graph. (SIGGRAPH Proc.) 29*, 4, 39:1–39:10.

OEFNER, F., 2013. Orchid. `http://fabianoefner.com/?portfolio=orchid`.

OKUMURA, Y. 2005. *Developing a spectral and colorimetric database of artist paint materials*. Master's thesis, Rochester Institute of Technology, NY.

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

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

REYNOLDS, O. 1886. On the theory of lubrication and its application to mr. beauchamp tower's experiments, including an experimental determination of the viscosity of olive oil. *Phil. Trans. Royal Soc. London 177*, 157–234.

SAVVA, N. 2007. *Viscous fluid sheets*. PhD thesis, Massachusetts Institute of Technology.

SELLE, A., LENTINE, M., AND FEDKIW, R. 2008. A mass spring model for hair simulation. *ACM Trans. Graph. (SIGGRAPH Proc.) 27*, 3 (Aug.), 64.1–64.11.

SMEREKA, P. 2003. Semi-implicit level set methods for curvature and surface diffusion motion. *J. Sci. Comput. 19*, 1, 439–456.

SMITS, B. 1999. An rgb-to-spectrum conversion for reflectances. *Journal of Graphics Tools 4*, 4, 11–22.

STOMAKHIN, A., SCHROEDER, C., CHAI, L., TERAN, J., AND SELLE, A. 2013. A material point method for snow simulation. *ACM Trans. Graph. (SIGGRAPH Proc.) 32*, 4, 102.

Stomakhin, A., Schroeder, C., Jiang, C., Chai, L., Teran, J., and Selle, A. 2014. Augmented mpm for phase-change and varied materials. *ACM Transactions on Graphics (TOG) 33*, 4, 138.

Wang, H., Mucha, P. J., and Turk, G. 2005. Water drops on surfaces. In *ACM Trans. Graph. (SIGGRAPH Proc.)*, vol. 24, ACM, 921–929.

Wicke, M., Ritchie, D., Klingner, B. M., Burke, S., Shewchuk, J. R., and O'Brien, J. F. 2010. Dynamic local remeshing for elastoplastic simulation. *ACM Trans. Graph. (SIGGRAPH Proc.) 29*, 4, 49:1–49:11.

Wojtan, C., and Turk, G. 2008. Fast viscoelastic behavior with thin features. *ACM Trans. Graph. (SIGGRAPH Proc.) 27*, 3, 47.

Wojtan, C., Thürey, N., Gross, M., and Turk, G. 2009. Deforming meshes that split and merge. In *ACM Trans. Graph. (SIGGRAPH Proc.)*, vol. 28, 76:1–76:10.

Xu, J., and Zhao, H. 2003. An Eulerian formulation for solving partial differential equations along a moving interface. *J. of Sci. Comput. 19*, 1, 573–594.

Yasuda, K. 1979. *Investigation of the analogies between viscometric and linear viscoelastic properties of polystyrene fluids.* PhD thesis, Massachusetts Institute of Technology.

Yu, J., Wojtan, C., Turk, G., and Yap, C. 2012. Explicit mesh surfaces for particle based fluids. *Comp. Graph. Forum (Eurographics Proc.) 31*, 815–824.

Zheng, W., Yong, J.-H., and Paul, J.-C. 2006. Simulation of bubbles. In *SCA '06: Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation*, 325–333.

Zhou, Y., Lun, Z., Kalogerakis, E., and Wang, R. 2013. Implicit integration for particle-based simulation of elasto-plastic solids. In *Computer Graphics Forum*, vol. 32, 215–223.

Zhu, Y., and Bridson, R. 2005. Animating sand as a fluid. *ACM Trans. Graph. (SIGGRAPH Proc.) 24*, 3, 965–972.

Zhu, B., Quigley, E., Cong, M., Solomon, J., and Fedkiw, R. 2014. Codimensional surface tension flow on simplicial complexes. *ACM Trans. Graph. (SIGGRAPH Proc.) 33*, 4, 111:1–111:11.

Appendix

We implemented a fully-implicit integration scheme for elasticity for comparison purposes. If we take $\dot{\boldsymbol{\varepsilon}}_G = \frac{1}{2}F^T\dot{\gamma}F$, we can write the fully-implicit formula as

$$\vec{u}^{**} = \vec{u}^* + \frac{\mu_e \Delta t}{\rho}\nabla \cdot (\boldsymbol{\varepsilon}^* + \frac{\Delta t}{2}\boldsymbol{F}^{*T}(\nabla\vec{u}^{**} + \nabla\vec{u}^{**T})\boldsymbol{F}^*).$$

We solve for the three components of velocity in one system, discretized as

$$[\bar{\boldsymbol{W}} + \frac{\mu_e \Delta t^2}{2\rho}\bar{\boldsymbol{G}}^T(\boldsymbol{P}\bar{\boldsymbol{F}}^T)^2\bar{\boldsymbol{W}}^{-1}(\boldsymbol{I}+\boldsymbol{P})\bar{\boldsymbol{G}}]\bar{u}^{**} = \bar{\boldsymbol{W}}\bar{u}^* - \frac{\mu_e}{\rho}\bar{\boldsymbol{G}}^T\bar{\varepsilon}^*,$$

where $\bar{u}$ is a vector containing velocity components in all three dimensions and $\bar{\varepsilon}$ is a vector of strains repeated three times, once for each dimension. $\bar{\boldsymbol{F}}$, $\bar{\boldsymbol{W}}$, and $\bar{\boldsymbol{G}}$ are matrices with three block entries ($\boldsymbol{F}$, $\boldsymbol{W}$, and $\boldsymbol{G}$, respectively) along the diagonal, one for each dimension. $\boldsymbol{P}$ is a permutation matrix used as a transpose operator. We use the Gauss-Seidel method to solve this non-symmetric system.