

# A Thermomechanical Material Point Method for Baking and Cooking

MENGYUAN DING, University of California, Los Angeles  
XUCHEN HAN, University of California, Los Angeles  
STEPHANIE WANG, University of California, Los Angeles  
THEODORE F. GAST, Jixie Effects Inc.  
JOSEPH M. TERAN, University of California, Los Angeles



Fig. 1. We demonstrate our method with various cooking and baking examples. **Top left** dehydrated apple slices. **Bottom left** pouring pancake batter. **Center** tearing a loaf of bread. **Top right** baking cookies. **Bottom right** baking loaves of bread.

We present a Material Point Method for visual simulation of baking breads, cookies, pancakes and similar materials that consist of dough or batter (mixtures of water, flour, eggs, fat, sugar and leavening agents). We develop a novel thermomechanical model using mixture theory to resolve interactions between individual water, gas and dough species. Heat transfer with thermal expansion is used to model thermal variations in material properties. Water-based mass transfer is resolved through the porous mixture, gas represents carbon dioxide produced by leavening agents in the baking process and dough is modeled as a viscoelastoplastic solid to represent its varied and complex rheological properties. Water content in the mixture reduces during the baking process according to Fick's Law which contributes to drying and cracking of crust at the material boundary. Carbon dioxide gas produced by leavening agents during baking creates internal pressure that causes rising. The viscoelastoplastic model for the dough is temperature dependent and is used to model melting and solidification. We discretize the governing

Authors' addresses: Mengyuan Ding, University of California, Los Angeles, Los Angeles, mengyuanding@ucla.edu; Xuchen Han, University of California, Los Angeles, Los Angeles, xhan0619@ucla.edu; Stephanie Wang, University of California, Los Angeles, Los Angeles, evast@ucla.edu; Theodore F. Gast, Jixie Effects Inc. Los Angeles, tgast@jixiefx.com; Joseph M. Teran, University of California, Los Angeles, Los Angeles, jteran@math.ucla.edu.

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

© 2019 Copyright held by the owner/author(s). Publication rights licensed to ACM.  
0730-0301/2019/11-ART192 \$15.00  
<https://doi.org/10.1145/3355089.3356537>

equations using a novel Material Point Method designed to track the solid phase of the mixture.

CCS Concepts: • **Computing methodologies** → **Physical simulation**.

Additional Key Words and Phrases: Material Point Method, Physical Simulation, Baking

## ACM Reference Format:

Mengyuan Ding, Xuchen Han, Stephanie Wang, Theodore F. Gast, and Joseph M. Teran. 2019. A Thermomechanical Material Point Method for Baking and Cooking. *ACM Trans. Graph.* 38, 6, Article 192 (November 2019), 14 pages. <https://doi.org/10.1145/3355089.3356537>

## 1 INTRODUCTION

Whether it is bread rising in the oven, cookies oozing with melting chocolate chips, or a pancake sizzling in a pan, baking and cooking are integral parts of our everyday lives. It is therefore important and yet challenging to model these phenomena accurately when creating compelling virtual scenes for computer graphics applications. Famously, Shah [2007] demonstrated the compelling nature of food dynamics in Pixar's *Ratatouille*. Surprisingly, given our everyday familiarity, these processes involve a wide range of complex physical phenomena including heat and mass transfer [Broyart and Trystram 2002; Nicolas et al. 2014], viscous and elastic rheology [Faridi and Faubion 2012], dynamics of porous mixtures [Debaste et al. 2010; Sakin-Yilmazer et al. 2013] and many more. We develop a model and numerical methods that can capture some of the most characteristic visual aspects of the baking or general cooking process, such

as melting, dehydrating, rising, and gelatinization. Furthermore, our approach allows for realistic simulation of user interactions like breaking and folding of the materials at various stages in the cooking process.

We propose a porous thermo-viscoelastoplastic mixture model. Melting effects are captured by a temperature-based change in viscoelastoplastic constitutive laws. During the cooking process water diffuses through the surface of the material according to Fick's law. This allows for effects such as wrinkling and curling of dehydrated fruit, as well as cracking of the top of baked goods [Thorvaldsson and Skjöldebrand 1998]. Leavening agents are often a predominant source of rising in baked goods, and we focus on the effects of chemical leaveners like baking powder and baking soda. With these agents, carbon dioxide ( $CO_2$ ) is created through a chemical reaction, which then expands under heat to help the dough to rise [Nicolas et al. 2014; Zhang and Datta 2006]; this reaction is activated at the critical temperature, peaks as temperature rises and finally gets deactivated when it goes beyond a threshold. Our model keeps track of the  $CO_2$  creation to capture the rising effect in baking. At the final stage of the baking process, flour gelatinization takes place and the baked goods become much more elastic and less viscous than the initial dough or batter [Vanin et al. 2009]. This is achieved in our model with temperature-dependent plasticity. We demonstrate these abilities with baking, tearing and dehydrating examples.

While dough and batter are mixtures of constituents that include water, leavening agents, flour, eggs, fat, sugar and others, we model the non-water or non- $CO_2$  agents as a single phase in a three species mixture model.  $CO_2$  from leavening agents and water make up the other two species in the mixture. We refer to the mixture of non-water and non- $CO_2$  species as the solid phase and model it as viscoelastoplastic with parameters that vary with temperature. This allows us to address rheological changes induced by cooking that do not arise from the effects of the leavening agents or water-based mass transfer. It is a simplification, but we confidently make it as it reduces the modeling complexity without precluding important features like melting and solidification.

We discretize our model with a novel Material Point Method [Sulsky et al. 1994] designed to treat thermomechanical porous mixtures and temperature dependent chemical production of gas from leavening agents. Our approach is designed to track the solid phase (everything but the water and  $CO_2$ ) since it is most apparent for visual rendering. Various MPM approaches have treated porous mixtures of water, gas and solid species [Abe et al. 2014; Bandara et al. 2016; Bandara and Soga 2015; Gao et al. 2018a; Jassim et al. 2013; Tampubolon et al. 2017; Yerro et al. 2015; Zhang et al. 2009]. Tampubolon et al. [2018a; 2017] use two sets of material points to track distinct water and sand phases. This allows for detailed visual resolution of each phase, but it is costly since each phase must be treated with a separate grid and their interactions are resolved via a stiff interaction term. Our approach is similar to [Abe et al. 2014; Bandara et al. 2016; Yerro et al. 2015] in that they use one set of material points and track relative motion of the other species. However, these works only considered mixtures of soil, water and air, largely for landslide applications. While there are some similarities in the treatment of porosity, none have considered thermal effects



Fig. 2. **Muffin.** The **left column** depicts baking of a tray of muffins, resulting in a classic dome shape on top. The **right column** shows a muffin torn open to reveal a fully cooked interior and melted chocolate chips. Surface meshes of the fractured muffin are generated through the mesh post-process of [Wang et al. 2019] and are used for rendering.

in the solid species as we do. Also, none have modeled gas pressures arising from temperature dependent chemical reactions. We summarize our contributions as

- A thermo-viscoelastoplastic model of dough and batter that approximates mixtures of non-water or  $CO_2$  contents, i.e. ingredients like flour, egg, fat, sugar, etc.
- A three species porous mixture model of the water,  $CO_2$  and remaining dough and batter contents.
- A model for the thermomechanical production of  $CO_2$  from leavening agents and its influence on the viscoelastoplastic rheology of the remaining materials in the solid phase.
- A novel MPM discretization of three species thermomechanical mixtures of solid, gas and water including the effects of chemical production of  $CO_2$  from leavening agents.

## 2 PREVIOUS WORK

Many efforts have been made to study one or a few aspects of complicated food processes. Vanin et al. [2009] focus on the formation and distinct texture of bread crust due to significantly lower water content than the crumb caused by rapid drying on the surface from the heat of the oven. Guillard et al. [2004] model the moisture content evolution in dry biscuit based on Fick's second law [1855]. De Cindio and Corraera [1995] view a leavened dough as a viscoelastic homogeneous macrosystem but also a gas-paste microsystem to model the interactions between the two phases during mixing, leavening and baking. Zhang and Datta [2006] develop a coupled system of solid, liquid water, water vapor and carbon dioxide phases to model the bread baking process. They model the material as viscoelastic and track the evolution of temperature, moisture, volume and surface coloration during bread baking. Yang et al. [2017] develop a unified particle framework for simulating various viscoelastic soft-matter with phase changes using conservative Cahn-Hilliard advection. They produce compelling simulations of phase transitions for cooking eggs and melting butter. Liu et al. [2015] use elastoplasticity

to simulate porous dehydration of fruits and other foods with the Finite Element Method (FEM). Chen et al. [2019] proposed a data-driven model that predictively generates RGB images of dough surface browning. Blutinger et al. [2019] studied the application of laser-heating technology to browning dough.

Graphics applications have used mixture theory and multi-species simulations to model porous water, sand and air mixtures [Gao et al. 2018a; Liu et al. 2008; Tampubolon et al. 2017]. Many other graphics applications have made compelling use of multi-species simulations for water, sprays and foams [Losasso et al. 2008; Nielsen and Osterby 2013; Takahashi et al. 2003; Yang et al. 2014], liquids with bubbles [Mihalef et al. 2009; Ren et al. 2015; Thürey et al. 2007] as well as for mixing of fluids [Bao et al. 2010; He et al. 2015; Kang et al. 2010; Ren et al. 2014; Yang et al. 2015]. Various models and numerical methods have been used to simulate realistic viscoelastic and plastic material behaviors [Bargteil et al. 2007; Barreiro et al. 2017; Gerszewski et al. 2009; Goktekin et al. 2004; Müller et al. 2005; Teschner et al. 2004; Wicke et al. 2010; Wojtan and Turk 2008]. Heat transfer and viscoelastic melting have also been used in many graphics applications to simulate visual effects of phase change and melting [Carlson et al. 2002; Keiser et al. 2005; Losasso et al. 2006; Maréchal et al. 2010; Müller et al. 2004; Terzopoulou et al. 1991; Wei et al. 2003; Zhao et al. 2006].

The Material Point Method (MPM) [1994] has been used in many computer graphics applications, including snow [Stomakhin et al. 2013], non-Newtonian fluids and foams [Ram et al. 2015; Yue et al. 2015], heat transfer and phase change [Gao et al. 2018b; Stomakhin et al. 2014], elastic and porous materials [Fei et al. 2018; Guo et al. 2018; Jiang et al. 2017a], and granular materials [Daviet and Bertails-Descoubes 2016; Klár et al. 2016; Yue et al. 2018]. Various MPM approaches have treated porous mixtures of different liquid, gas and solid species. Bandara et al. [2014; 2016; 2015] simulate mixtures of air, water and soil for landslide applications. In [Bandara and Soga 2015], they use two sets of Lagrangian marker particles for water and soil respectively. Tampubolon et al. [2018a; 2017] use a similar approach. In [Abe et al. 2014; Bandara et al. 2016], they use a single set of material points for the soil and track the motion of the water relative to the soil, but they neglect the water acceleration. Zhang et al. [2009] also neglect the water acceleration terms. Yerro et al. [2015] build on the two-phase porous MPM approach of Jassim et al. [2013] to simulate three species mixtures of air, water and soil. Bandara et al. [2015] use Biot's [1941] phenomenological model for their governing equations.

A few aspects of our approach are closely related to existing MPM techniques. We treat thermal effects similar to Stomakhin et al. [2014] and Gao et al. [2018b] and our treatment of multi-species mixtures is similar to the one-species-centric formulations in [Abe et al. 2014; Bandara et al. 2016; Bandara and Soga 2015; Yerro et al. 2015; Zhang et al. 2009]. Stomakhin et al. [2014] discretize the heat equation for temperature using a finite difference approach and use a Particle-In-Cell (PIC) [1964] transfer of thermal quantities between particles and grid. Gao et al. [2018b] improved on this by using an MPM style weak form discretization of the heat equation where particles are used as quadrature points to avoid voxelization associated with finite difference discretization. Furthermore, Gao et al. use higher order PIC ([Jiang et al. 2017b]) transfers. We use a

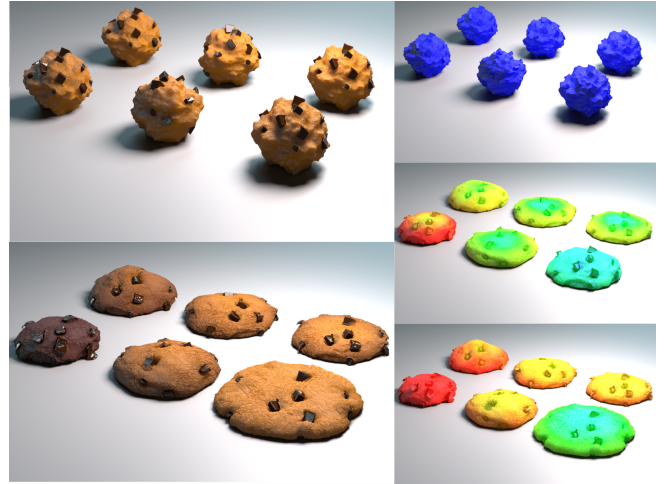


Fig. 3. **Cookies.** The cookies are initialized as dough balls (**top left**). The **bottom left** shows the end results. The top row from left to right varies with decreasing amounts of leavener, and the bottom row from left to right with decreasing oven temperature. The **right column** depicts the heat transfer progress in the cookies during baking with color varying from blue to green then to red with increasing temperature.

particle-based weak form derivation of the heat equation as in Gao et al., however we use Fluid-Implicit-Particle (FLIP) [1986] transfers rather than PIC transfers. Although FLIP transfers can suffer from noise [Jiang et al. 2015], we found that it was not an issue for our examples. Lastly, we note that we include the effects of thermal expansion which neither Stomakhin et al. nor Gao et al. address.

Our approach, as with many other MPM approaches, use the notion of effective stress [Atkin and Craine 1976] with multi-species mixtures. These approaches naturally admit MPM discretizations where particles track each species, however stiff drag interaction terms can require small time steps [Bandara and Soga 2015; Gao et al. 2018a; Tampubolon et al. 2017]. Alternatively, we track the motion of species relative to one set of particles. This is similar to many formulations for porous soil in geomechanical applications [Abe et al. 2014; Bandara et al. 2016; Bandara and Soga 2015; Yerro et al. 2015; Zhang et al. 2009]. However, none of these formulations account for thermal effects that occur during baking. Furthermore, most only account for water and soil mixtures. Only Yerro et al. [2015] consider three species mixtures of soil, water and air. While they discretize momentum balance relative to the mixture configuration, however, we do so relative to the solid phase which is easier to track numerically.

### 3 GOVERNING EQUATIONS

We model our materials as a mixture of water, gas and solid constituents and use a multi-species continuum model to derive the equations of motion from the governing physics [Atkin and Craine 1976]. With this assumption we use a different flow map for each constituent which can then be combined to describe the kinematics of the mixture. Formally, we use  $\phi^\alpha : B_0 \times [0, T] \rightarrow \mathbb{R}^3$  to define the motion of species  $\alpha$  where  $\alpha = w$  for water,  $\alpha = s$  for solid

and  $\alpha = g$  for gas. Here  $B_0 \subset \mathbb{R}^3$  represents the configuration of the material at time  $t = 0$ . We refer to points  $\mathbf{X} \in B_0$  as material points and  $\mathbf{x} \in B_t^\alpha = \{\mathbf{x} \in \mathbb{R}^3 | \exists \mathbf{X} \in B_0 \text{ with } \mathbf{x} = \phi^\alpha(\mathbf{X}, t)\}$  as world-space points where  $B_t^\alpha$  is the time  $t$  configuration of species  $\alpha$ . Furthermore, we use

$$\Omega_t^\alpha = \{\mathbf{x} \in \mathbb{R}^3 | \exists \mathbf{X} \in \Omega_0 \text{ with } \mathbf{x} = \phi^\alpha(\mathbf{X}, t)\} \quad (1)$$

for  $\Omega_0 \subset B_0$  to denote subsets of the domains. Note that we do not distinguish between different species in  $B_0$  since we assume they are all present in the initial mixture. We further assume that the motion of the gas relative to the solid is negligible, and that therefore  $\phi^s = \phi^g$  (and  $B_t^s = B_t^g$ ). We also assume that water may diffuse out of the mixture, but that our mixture will never be completely dry. With this assumption  $B_t^s \subset B_t^w$ .

The Lagrangian velocity of each constituent is defined from the flow map kinematics as  $\mathbf{V}^\alpha(\mathbf{X}, t) = \frac{\partial \phi^\alpha}{\partial t}(\mathbf{X}, t)$  and the Eulerian (or world-space) velocity is defined as  $\mathbf{v}^\alpha(\mathbf{x}, t) = \mathbf{V}^\alpha(\phi^{\alpha^{-1}}(\mathbf{x}, t), t)$  where  $\phi^{\alpha^{-1}}(\mathbf{x}, t)$  is the inverse flow map of the species  $\alpha$ . This is the standard relationship between Lagrangian and Eulerian representations, but done with respect to the individual species flow maps. We use  $\mathbf{F}^\alpha(\mathbf{X}, t) = \frac{\partial \phi^\alpha}{\partial \mathbf{X}}(\mathbf{X}, t)$  to denote the Jacobian of the constituent mappings (or deformation gradients) and  $J^\alpha = \det(\mathbf{F}^\alpha)$  to denote their determinants. Intuitively, the deviation of  $\mathbf{F}^\alpha(\mathbf{X}, t)$  from orthogonality indicates how non-rigid the motion is local to material point  $\mathbf{X}$  and  $J^\alpha$  expresses the local volume gain ( $J^\alpha > 1$ ) or loss ( $J^\alpha < 1$ ). Furthermore, we use  $\phi : B_0 \times [0, T] \rightarrow \mathbb{R}^3$  to denote the flow map of the mixture. It is related to the Eulerian velocity of the mixture  $\mathbf{v}$  as  $\frac{\partial \phi}{\partial t}(\mathbf{X}, t) = \mathbf{v}(\phi(\mathbf{X}, t), t)$ . The deformation gradient of the mixture and its determinant are denoted as  $\mathbf{F}$  and  $J$  respectively.

Following [Atkin and Craine 1976], the mass and momentum densities of the species are denoted as  $\rho^\alpha : B_t^\alpha \rightarrow \mathbb{R}$  and  $\rho^\alpha \mathbf{v}^\alpha : B_t^\alpha \rightarrow \mathbb{R}$  respectively. The mass and momentum densities of the mixture are defined as the respective sums  $\rho = \sum_\alpha \rho^\alpha$  and  $\rho \mathbf{v} = \sum_\alpha \rho^\alpha \mathbf{v}^\alpha$ . With this convention, the velocity of the mixture is defined via mass average  $\mathbf{v} = \frac{\sum_\alpha \rho^\alpha \mathbf{v}^\alpha}{\sum_\alpha \rho^\alpha}$ . This mass averaged notion of velocity defines the motion of the mixture from the individual species, and we emphasize that it follows directly from conservation of mass and momentum considerations. Furthermore, in our formulation we assume that the mass density of the gas  $\rho^g$  is negligible compared to that of the water and solid and that the density of water is initially spatially constant and equal to  $\rho_0^w$ . With these assumptions the velocity of the mixture follows as  $\mathbf{v} \approx \frac{\rho^w \mathbf{v}^w + \rho^s \mathbf{v}^s}{\rho^w + \rho^s}$ .

Lastly, we use  $\Theta : B_0 \times [0, T] \rightarrow \mathbb{R}$  to refer to the material-space temperature of the solid and  $\theta : B_t \times [0, T] \rightarrow \mathbb{R}$  as the world-space temperature. They are related through  $\theta(\phi(\mathbf{X}, t), t) = \Theta(\mathbf{X}, t)$ .

### 3.1 Conservation of mass and momentum

Following [Atkin and Craine 1976], we assume each constituent obeys conservation of mass with respect to its own motion  $\frac{D^\alpha \rho^\alpha}{Dt} + \rho^\alpha \nabla \cdot \mathbf{v}^\alpha = 0$  where  $\frac{D^\alpha f}{Dt} = \frac{\partial f}{\partial t} + \frac{\partial f}{\partial \mathbf{x}} \mathbf{v}^\alpha$  is the material derivative with respect to motion  $\alpha$ . By considering the mass in an arbitrary

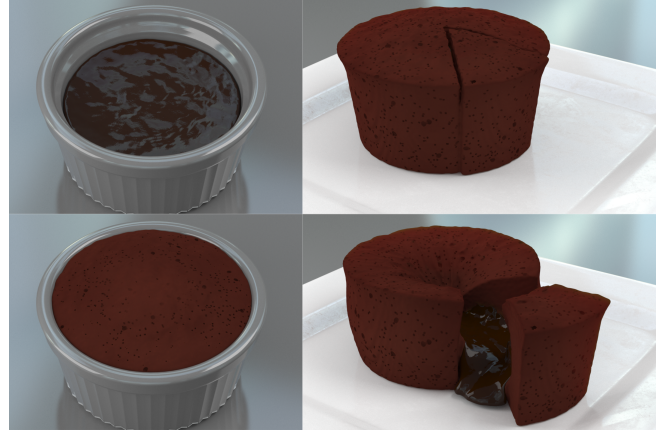


Fig. 4. **Lava Cake.** A lava cake is initialized as a homogeneous batter and baked in a ramekin, then plated and cut open, the molten center flows out.

subset of  $\Omega_t^\alpha \subset B_t^\alpha$  this can equivalently be expressed as

$$\frac{d}{dt} \int_{\Omega_t^\alpha} \rho^\alpha dx = \int_{\Omega_t^\alpha} \frac{d\rho^\alpha}{dt} dx + \int_{\partial\Omega_t^\alpha} \rho^\alpha \mathbf{v}^\alpha \cdot \mathbf{n} ds(\mathbf{x}) = 0 \quad (2)$$

Momentum balance for each constituent can similarly be expressed as

$$\frac{d}{dt} \int_{\Omega_t^\alpha} \rho^\alpha \mathbf{v}^\alpha dx = \int_{\Omega_t^\alpha} (\mathbf{f}^\alpha + \mathbf{p}^\alpha) dx + \int_{\partial\Omega_t^\alpha} \mathbf{t}^\alpha ds(\mathbf{x})$$

where  $\mathbf{f}^\alpha$  and  $\mathbf{t}^\alpha$  are the external body force and traction on phase  $\alpha$ , and the  $\mathbf{p}^\alpha$  term stands for the transfer of momentum between the constituents. We refer the reader to [Atkin and Craine 1976] for a more detailed derivation and motivation of these balances and the momentum exchanges between them. However, we note the mathematical relation

$$\frac{d}{dt} \int_{\Omega_t^\alpha} \rho^\alpha \mathbf{v}^\alpha dx = \int_{\Omega_t^\alpha} \frac{d}{dt} (\rho^\alpha \mathbf{v}^\alpha) dx + \int_{\partial\Omega_t^\alpha} \rho^\alpha \mathbf{v}^\alpha \otimes \mathbf{v}^\alpha \cdot \mathbf{n} ds(\mathbf{x}) \quad (3)$$

which states that the rate of change of the momentum of species  $\alpha$  in the region  $\Omega_t^\alpha$  varies based on the rate of change of the Eulerian momentum as well as the effect of the change of  $\Omega_t^\alpha$  under the motion of the species, as this will be useful for later derivations.

In contrast to other multi-species MPM approaches, e.g. Yerro et al. [2015], we consider conservation relative to the motion of the solid constituent. While the balance laws of the mixture are most naturally expressed in terms of its own motion, this can complicate the numerical method since it would typically require explicitly tracking the mixture motion. Instead, we follow the convention of Zhao et al. [Zhao and Papadopoulos 2013] to express mass and momentum conservation of the mixture relative to the solid constituent since it does not require explicit resolution of the mixture flow map.

First, we note that the rate of change of the mass of water in  $\Omega_t^s$  due to motion of the solid is

$$\frac{d}{dt} \int_{\Omega_t^s} \rho^w dx = \int_{\Omega_t^s} \frac{d\rho^w}{dt} dx + \int_{\partial\Omega_t^s} \rho^w \mathbf{v}^s \cdot \mathbf{n} ds(\mathbf{x}). \quad (4)$$

Here the time derivative takes into account the change in the set  $\Omega_t^s$  and its effect as the domain of integration. Consider the case when

$\Omega_t^w = \Omega_t^s$  at a given time  $t$ . In this case the sets are equal, but may not be so for earlier times  $\hat{t} < t$ , e.g.  $\Omega_{\hat{t}}^w \neq \Omega_{\hat{t}}^s$ , since they evolve under the respective constituent motions via Equation (1). We can subtract Equation (2) (in the case of  $\alpha = w$ ) from Equation (4) to yield an expression of conservation of water mass with respect to motion of the solid

$$\frac{d}{dt} \int_{\Omega_t^s} \rho^w dx + \int_{\partial\Omega_t^s} \rho^w (\mathbf{v}^w - \mathbf{v}^s) \cdot \mathbf{n} ds(\mathbf{x}) = 0.$$

We henceforth use  $\mathbf{q}^w = \rho^w (\mathbf{v}^w - \mathbf{v}^s)$  to denote the density weighted velocity of the water relative to the solid. Since the set  $\Omega_t^s$  is arbitrary, conservation of mass of the water with respect to the motion of the solid can be expressed as

$$\frac{D^s \rho^w}{Dt} + \rho^w \nabla \cdot \mathbf{v}^s + \nabla \cdot \mathbf{q}^w = 0, \mathbf{x} \in B_t^s. \quad (5)$$

As in the case of water mass, the rate of change of water momentum with respect to the motion of the solid can be expressed as

$$\frac{d}{dt} \int_{\Omega_t^s} \rho^w \mathbf{v}^w dx = \int_{\Omega_t^s} \frac{d}{dt} (\rho^w \mathbf{v}^w) dx + \int_{\partial\Omega_t^s} \rho^w \mathbf{v}^w \otimes \mathbf{v}^s \cdot \mathbf{n} ds(\mathbf{x}). \quad (6)$$

Again as with water mass, by considering coincident sets at time  $t$ ,  $\Omega_t^s = \Omega_t^w$  and subtracting Equation (3) (in the case of  $\alpha = w$ ) from Equation (6), we can conclude

$$\frac{d}{dt} \int_{\Omega_t^s} \rho^w \mathbf{v}^w dx = \frac{d}{dt} \int_{\Omega_t^w} \rho^w \mathbf{v}^w dx - \int_{\partial\Omega_t^s} \mathbf{v}^w \otimes \mathbf{q}^w \cdot \mathbf{n} ds(\mathbf{x}).$$

We can use this equality to express conservation of momentum of the mixture relative to the motion of the solid constituent as

$$\begin{aligned} & \frac{d}{dt} \int_{\Omega_t^s} \rho v dx \\ &= \frac{d}{dt} \int_{\Omega_t^s} (\rho^s \mathbf{v}^s + \rho^w \mathbf{v}^w) dx \\ &= \frac{d}{dt} \int_{\Omega_t^s} \rho^s \mathbf{v}^s dx + \frac{d}{dt} \int_{\Omega_t^w} \rho^w \mathbf{v}^w dx - \int_{\partial\Omega_t^s} \mathbf{v}^w \otimes \mathbf{q}^w \cdot \mathbf{n} ds(\mathbf{x}) \\ &= \int_{\Omega_t^s} (\mathbf{f}^s + \mathbf{p}^s) dx + \int_{\partial\Omega_t^s} \mathbf{t}^s ds(\mathbf{x}) + \int_{\Omega_t^w} (\mathbf{f}^w + \mathbf{p}^w) dx \\ &+ \int_{\partial\Omega_t^s} \mathbf{t}^w ds(\mathbf{x}) - \int_{\partial\Omega_t^s} \mathbf{v}^w \otimes \mathbf{q}^w \cdot \mathbf{n} ds(\mathbf{x}) \\ &= \int_{\Omega_t^s} \mathbf{f} dx + \int_{\partial\Omega_t^s} \mathbf{t} ds(\mathbf{x}) - \int_{\partial\Omega_t^s} \mathbf{v}^w \otimes \mathbf{q}^w \cdot \mathbf{n} ds(\mathbf{x}) \end{aligned}$$

where  $\mathbf{f} = \mathbf{f}^s + \mathbf{f}^w$  is the total body force on the mixture, and  $\mathbf{t} = \mathbf{t}^s + \mathbf{t}^w$  is the total traction and the sum of the momentum exchange terms  $\mathbf{p}^w + \mathbf{p}^s = \mathbf{0}$  is assumed to be zero [Atkin and Craine 1976]. Therefore, since the set  $\Omega_t^s \subset B_t^s$  is arbitrary, we can conclude that conservation of momentum of the mixture can be expressed with respect to motion of the solid constituent as

$$\frac{D^s \rho v}{Dt} + \rho v \nabla \cdot \mathbf{v}^s = \mathbf{f} + \nabla \cdot \boldsymbol{\sigma} - \nabla \cdot (\mathbf{v}^w \otimes \mathbf{q}^w), \mathbf{x} \in B_t^s. \quad (7)$$

Here  $\boldsymbol{\sigma}$  is the Cauchy stress in the mixture and it is related to the total traction of the mixture as  $\mathbf{t} = \boldsymbol{\sigma} \mathbf{n}$ .

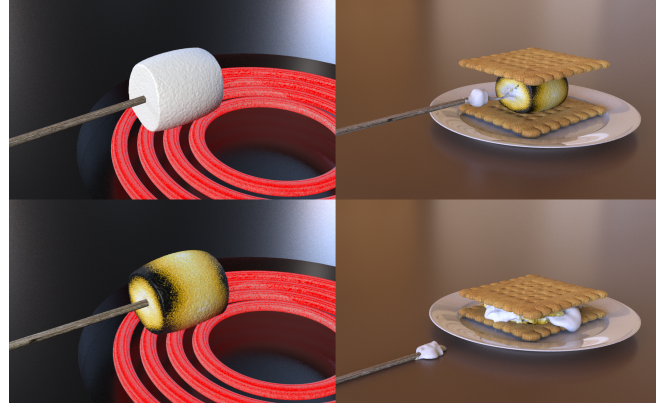


Fig. 5. S'more. The **left column** depicts a marshmallow roasting on the stovetop. Coloring is based on temperature. We press the marshmallow between two crackers (**right column**) and the melted interior flows out.

### 3.2 Fick's Law

We assume that the motion of the liquid phase (relative to the solid phase)  $\mathbf{q}^w$  is restricted to diffusion in the mixture; Fick's law [1855] states that

$$\mathbf{q}^w = -K \theta \rho^w \nabla s$$

where  $K > 0$  is a material diffusivity constant, and  $s = \frac{V^w}{V}$  stands for the saturation level of the dough, which is in the form of a volume fraction where  $V$  is a local volume of the mixture and  $V^w$  is the local volume of water in the mixture. Intuitively, this means that water content in the mixture flows from regions of higher saturation to regions of lower saturation at a rate that depends both on the material diffusivity  $K$  as well as the dough temperature  $\theta$ .

We note that the density of water  $\rho^w$  can be expressed in terms of the saturation  $s$  as

$$\rho^w = \frac{m^w}{V} = \frac{m^w}{V^w} \frac{V^w}{V} = \rho_0^w s. \quad (8)$$

Furthermore, we use  $\mathbf{q}^w \cdot \mathbf{n} = \beta^w \theta s$  for the boundary conditions in Equation (5) where  $\beta^w$  is a material constant that represents the rate of water mass loss due to exterior conditions.

## 4 HEAT TRANSFER

Heat flow through the material obeys the first and second law of thermodynamics, and following [Gonzalez and Stuart 2008] we summarize our heat energy balance equation as

$$\rho \alpha \frac{D\theta}{Dt} = \nabla \cdot (\kappa \nabla \theta) - \theta c \lambda \nabla \cdot \mathbf{v}, \mathbf{x} \in B_t. \quad (9)$$

Here,  $\alpha$  is the specific heat at constant volume which controls the rate at which temperature changes for a given amount of heating; and  $\kappa$  is the thermal conductivity which controls the speed with which heat diffuses through the body.  $c\lambda$  controls rate of temperature decrease resulting from thermal expansion and  $\lambda$  is a Lamé coefficient. Intuitively, since temperature measures the kinetic energy density at small scales, increases in material volume will tend to cool the material. We note that this effect was not considered in previous MPM work on thermal effects. We refer readers to the

supplemental technical document for the derivation [Ding et al. 2019]. Lastly, the heat flow boundary condition is

$$\kappa \nabla \theta \cdot \mathbf{n} = \beta(\theta - \theta_{out}), \mathbf{x} \in \partial B_t$$

where  $\mathbf{n}$  is the outward normal,  $\theta_{out}$  is the ambient temperature outside of the material and  $\beta$  controls the rate of temperature change that results from heat transfer with the ambient space.

## 5 CONSTITUTIVE MODEL

Again following [Atkin and Craine 1976], the stress in the mixture is a sum of the stresses in the solid, gas and water constituents

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}^s + \boldsymbol{\sigma}^g + \boldsymbol{\sigma}^w.$$

We note that although this is the Cauchy stress with respect to the configuration of the mixture, it can be expressed in the domain of the solid constituent in Equation (7). Here we discuss the constitutive models we use for each of the constituent species.

### 5.1 Solid stress

We assume the solid phase is itself a complex mixture of constituents which may include flour, fat, leavening agent, salt, sugar, or egg. These ingredients create a mixture material with a wide range of complex, non-Newtonian rheological behaviors whose properties, depending on temperature, vary from nearly liquid to nearly solid. Ideally each phase would be included in the mixture model, however to reduce complexity we instead adopt a temperature dependent viscoelastoplasticity model for the solid phase. This allows us to efficiently reproduce a wide range of complex rheological behaviors without the need to track each of the many species.

### 5.2 Elastic stress

We use a multiplicative decomposition of the deformation in the solid phase  $\mathbf{F}^s = \mathbf{F}^{s,E} \mathbf{F}^{s,P}$  where  $\mathbf{F}^{s,E}$  is the elastic component of the motion that will be penalized by the potential energy density and  $\mathbf{F}^{s,P}$  is the permanent deformation associated with the plasticity [Bonet and Wood 2008]. The stress in the solid phase arises from the potential energy density and thermal expansion as

$$\boldsymbol{\sigma}^s = \frac{\boldsymbol{\tau}^{s,E}}{J^s} - c\lambda J^s \theta. \quad (10)$$

where  $\boldsymbol{\tau}^{s,E}$  is the elastic Kirchhoff stress defined in terms of the elastic potential as  $\boldsymbol{\tau}^{s,E} = \frac{\partial \Psi}{\partial \mathbf{F}^{s,E}} \mathbf{F}^{s,E-T}$  and  $c$  is the same parameter that appears in Equation (9). The last term is due to thermal expansion and we refer the reader to [Ding et al. 2019; Gonzalez and Stuart 2008] for its derivation. Intuitively, as the material is heated, this term contributes an expansional pressure proportionate to temperature. We note that the elastic potential will have further temperature dependence from the thermal effects of plasticity (see Section 5.3). Our choice of potential is

$$\Psi(\mathbf{F}^{s,E}) = \frac{1}{2} \lambda (\text{tr}(\boldsymbol{\epsilon}))^2 + \mu \boldsymbol{\epsilon} : \boldsymbol{\epsilon}$$

where  $\boldsymbol{\epsilon} = \frac{1}{2} \log(\mathbf{F}^{s,E} \mathbf{F}^{s,E,T})$  is the elastic Hencky strain in the solid. Here  $\mu$  and  $\lambda$  are the Lamé coefficients. Note that the  $c\lambda$  term gives rise to a temperature varying positive (expansional) pressure. We also note that our primary motivation for adopting this potential

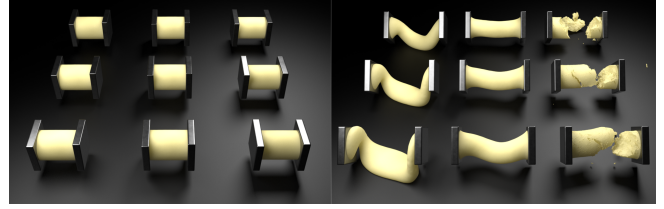


Fig. 6. **Dough Pull and Twist.** We apply the same pull and twist motion to dough cylinders of varying yield stress and viscosity parameters. The viscosity increases from bottom to top, and yield stress increases from left to right.

is to simplify discrete plastic integration as in [Gaume et al. 2018; Klár et al. 2016]. This simplicity is a consequence of its property  $\boldsymbol{\tau}^{s,E} = \mathbf{C} \boldsymbol{\epsilon}$  where  $\mathbf{C} = 2\mu \mathcal{I} + \lambda \mathbf{I} \otimes \mathbf{I}$  is the isotropic fourth order elastic stiffness tensor and  $\mathcal{I}$  is the fourth order identity tensor. Furthermore, it is for this reason that we introduce the notion of the Kirchhoff stress in Equation (10) despite being generally less intuitive than the Cauchy stress.

### 5.3 Viscoplasticity

While the potential energy varies with the elastic portion of the decomposition, the evolution of the plastic portion is defined in terms of a yield condition that identifies states of stress consistent with observed material behavior. We represent the yield condition in terms of the Kirchhoff stress  $\boldsymbol{\tau} = J\boldsymbol{\sigma}$  and a signed distance function  $f(\boldsymbol{\tau})$  where  $f(\boldsymbol{\tau}) \leq 0$  indicates a state of physically meaningful stress. We express this evolution of the plastic flow and its relation to the yield condition in terms of the left Cauchy-Green strain  $\mathbf{b}^E = \mathbf{F}^{s,E} \mathbf{F}^{s,E,T}$

$$\frac{D^s \mathbf{b}^E}{Dt} = \frac{\partial \mathbf{v}^s}{\partial \mathbf{x}} \mathbf{b}^E + \mathbf{b}^E \frac{\partial \mathbf{v}^{s,T}}{\partial \mathbf{x}} + \mathcal{L}_{\mathbf{v}^s} \mathbf{b}^E. \quad (11)$$

Here  $\mathcal{L}_{\mathbf{v}^s}^s \mathbf{b}^E$  is the Lie derivative with respect to the motion of the solid and is defined in terms of the yield condition. We use an associative flow rule  $\mathcal{L}_{\mathbf{v}^s} \mathbf{b}^E = -2\gamma \frac{\partial f}{\partial \boldsymbol{\tau}} \mathbf{b}^E$  where  $\gamma$  is the flow rate, and  $\frac{\partial f}{\partial \boldsymbol{\tau}} \mathbf{b}^E$  specifies the direction of maximum energy dissipation [Bonet and Wood 2008]. For rate independent plasticity  $\gamma$  is chosen so that the stress satisfies the yield condition  $f(\boldsymbol{\tau}) \leq 0$ . In the case of viscoplasticity we do not restrict the stress to always satisfy the yield condition; instead we choose the rate as

$$\gamma = \frac{1}{\eta} \frac{\partial g}{\partial f}(f(\boldsymbol{\tau}))$$

where  $\eta = \eta(\theta)$  is a viscosity penalty parameter,  $g(f)$  is a monotonic function for positive  $f$  and zero otherwise. It is chosen to penalize states of stress outside the yield surface without making adherence to the condition of a hard constraint. For simplicity, we use  $g(f) = \frac{1}{2} f^2$  whenever  $f$  is positive, and zero otherwise. Note that there is no plastic flow when the stress is inside the yield surface since in this case  $f(\boldsymbol{\tau}) \leq 0$  and therefore  $\gamma = 0$  since the argument to  $\frac{\partial g}{\partial f}$  is non-positive. Lastly, we also note that as  $\eta \rightarrow 0$ ,  $\gamma$  is chosen as in the rate independent case. We demonstrate the effect of different plasticity parameters with a dough pull and twist example, see Figure 6.

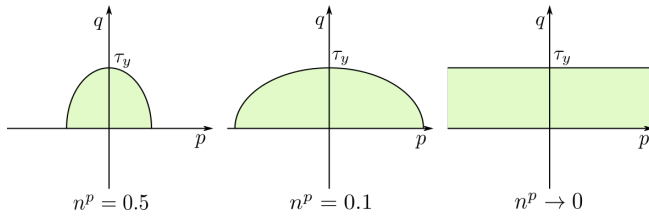


Fig. 7. **Yield Surface.** We visualize the yield surface with different choices of porosity. The yield surface is an ellipsoidal shape in the upper half-plane. In the limit  $n^g \rightarrow 0$ , the yield conditions becomes equivalent to a von Mises type criterion.

**5.3.1 Yield condition.** We use a modified temperature and porosity dependent Cam clay yield condition [Roscoe and Burland 1968]. This model is typically used for porous viscoelastoplastic materials and gives rise to a wide range of behaviors similar to those exhibited by the solid phase mixture for baking materials. This yield condition is ellipsoidal in principle stress space (Figure 7) and is given by

$$f(\boldsymbol{\tau}) = \frac{9}{4 \ln^2 \left( \frac{1}{n^p} \right) \left( \frac{7}{2} n^p + 1 \right)} p^2 + q^2 - \tau_y(\theta)^2 \leq 0 \quad (12)$$

where  $p = -\frac{1}{3} \text{tr}(\boldsymbol{\tau})$  is the mean normal stress,  $q = \sqrt{\frac{3}{2}} \|\boldsymbol{\tau} + p\mathbf{I}\|_F$  is the effective stress (proportional to magnitude of deviatoric stress) and  $\tau_y(\theta)$  is the yield stress.  $n^p \in (0, 1)$  is the gas porosity of the solid-gas mixture (see Section 5.4). We adopt the model of [Haghi and Anand 1992] where the coefficient of  $p$  is a monotonic increasing function of  $n^p$  and vice versa. Note that in the limit  $n^p \rightarrow 0$ , the yield function becomes  $f(\boldsymbol{\tau}) = q^2 - \tau_y^2 \leq 0$ , which is a von Mises type yield criterion equivalent to the viscoplastic formulation of [Yue et al. 2015]. Intuitively, a more porous solid mixture ( $n^p > 0$ ) will have limits on the degree to which it can achieve cohesion. This cohesion limit is expressed in terms of the tip of the ellipse on the negative portion of the  $p$  axis.

**5.3.2 Temperature dependence.** We model the viscosity parameter  $\eta$  and yield stress  $\tau_y$  as piecewise linear functions of temperature in order to track the melting and gelatinization of the dough that are often observed during baking. In the initial stage of heating, the fat in the dough, if present, would soften and melt, causing the mixture to appear less viscous and more inelastic. Flour gelatinization takes place later on which increases the elastic strength of the mixture, and this process cannot be reversed during baking. Based on this intuition we set  $\eta$  to be a non-negative monotonic non-increasing function of  $\theta$ , and  $\tau_y$  decreasing initially then increasing with respect to temperature, as shown in Figure 8. The parameters are fixed once the material reaches the gelatinization temperature. We follow this progression of plasticity parameters with respect to temperature during baking. The actual values vary among the different examples for optimal visual effects.

## 5.4 Gas stress

We model the effect of the leavening agents in the mixture with a chemical reaction that creates carbon dioxide ( $CO_2$ ). The  $CO_2$  then expands under heating to drive the rising of the mixture during the baking process. We model the chemical reaction with the differential

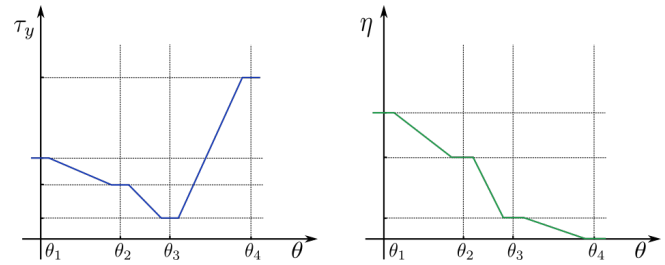


Fig. 8. **Temperature Dependent Plasticity.** We plot the change in yield stress (**left**) and viscosity (**right**) with temperature. The marked temperatures  $\theta_1, \theta_2, \theta_3, \theta_4$  stand for typical temperature ranges for fridge, room environment, fat melting point, and starch gelatinization respectively.

equation [Nicolas et al. 2014; Zhang and Datta 2006]

$$\frac{dn^g}{dt} = \alpha^g R_0 \exp \left( - \left( \frac{\theta - \theta_r}{\Delta\theta} \right)^2 \right).$$

Here  $n^g$  stands for the molar count of  $CO_2$  per unit volume,  $\alpha^g$  is a material constant,  $R_0$  is initial mass density of the mixture,  $\theta_r$  is a reference temperature at which the chemical reaction reaches its peak rate (Figure 12), and  $\Delta\theta$  is another constant scaling parameter. We define the gas porosity of the solid material as  $n^p = \frac{V^g}{V^s}$ , where  $V^g$  stands for a local representative gas volume, and  $V^s$  is the local representative volume of the solid/gas mixture. We assume that porosity changes as the solid mixture flows plastically as

$$n^p = J^{s,P} - 1 + n_0^p$$

where  $n_0^p$  is the initial gas porosity. Intuitively, as the material is kneaded it will gain volume plastically, allowing more room for gas. With these conventions,  $\frac{n^g}{J^s n^p}$  measures the amount of gas in the pore volume. Therefore, using the ideal gas law to model the pressure of  $CO_2$ , the stress in the gas constituent is

$$\boldsymbol{\sigma}^g = - \frac{n^g R \theta}{J^s n^p} \mathbf{I} \quad (13)$$

where  $R$  is the ideal gas constant.

## 5.5 Water stress

We adopt the saturation-based pressure model of [Zhao and Papadopoulos 2013] for the stress in the water phase. As the water leaves a region of the material, the mixture experiences a negative pressure and will tend to contract inwards as the elastic stress is allowed to compress the mixture in the absence of liquid. When water enters a region, the material will experience a positive pressure as the liquid pushes on the mixture. This is modeled with the linear relation

$$\boldsymbol{\sigma}^w = -\hat{c}(s - s_0)\mathbf{I} \quad (14)$$

where  $s_0$  is the initial saturation.

## 6 DISCRETIZATION

Our discretization is designed to track the solid constituents. We store  $\mathbf{x}_p^n = \boldsymbol{\phi}^s(\mathbf{X}_p, t^n)$  to denote the time  $t^n$  location of discrete material particle  $\mathbf{X}_p$  under the solid motion and an initial representative

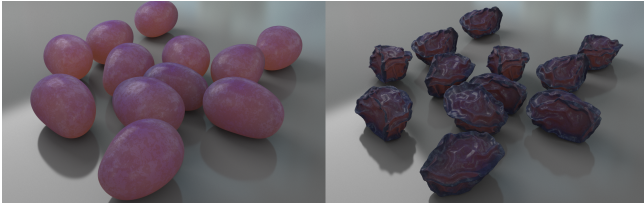


Fig. 9. **Dehydrating Grapes.** Grapes are dried at constant temperature. The smooth exterior (left) wrinkles up due to water loss of the interior flesh(right).

volume for each particle  $V_p^{s,0}$  that partitions the initial domain of the solid. Additionally, we store the translational  $\mathbf{v}_p^n = \frac{\partial \phi^s}{\partial t}(\mathbf{X}_p, t^n)$ , and affine  $\mathbf{A}_p^n \approx \frac{\partial \mathbf{v}^s}{\partial \mathbf{x}}(\mathbf{x}_p^n, t^n)$  velocities and elastic/plastic decomposition of the deformation gradient  $\mathbf{F}_p^{s,En} \mathbf{F}_p^{s,Pn} = \mathbf{F}_p^{s,n} = \frac{\partial \phi^s}{\partial \mathbf{X}}(\mathbf{X}_p, t^n)$  of the solid phase. The water  $m_p^{w,n}$  and solid  $m_p^s$  masses are also stored. We note that the solid mass does not change with time since we track the solid motion in a Lagrangian manner. Furthermore the water mass satisfies  $m_p^{w,n} = \rho^w(\mathbf{x}_p^n, t^n) V_p^{s,n}$  where  $V_p^{s,n} = J_p^{s,n} V_p^{s,0}$  is a representative volume around  $\mathbf{x}_p^n$  in the time  $t^n$  configuration of the solid and  $J_p^{s,n} = \det(\mathbf{F}_p^{s,n})$ . We store the water saturation  $s_p^n$  and the density weighted velocity of the water relative to the solid  $\mathbf{q}_p^{w,n}$ . We also store the temperature  $\theta_p^n$  and molar count per volume  $n_p^g$ .

### 6.1 Time step

We update the discrete state from time  $t^n$  to  $t^{n+1}$  by discretizing the governing equations with MPM. This requires transferring various data from particles to grid using grid based interpolating functions. We use quadratic B-splines as in [Jiang et al. 2017a]. We let  $N_i(\mathbf{x})$  denote the grid based interpolating function associated with grid node  $\mathbf{x}_i$  and  $w_{ip} = N_i(\mathbf{x}_p^n)$ ,  $\nabla w_{ip} = \nabla N_i(\mathbf{x}_p^n)$  for short. We transfer the mass, velocity, temperature and particle representative volume to the grid. The momentum is transferred using APIC [Jiang et al. 2015]

$$m_i^{s,n} = \sum_p m_p^s w_{ip}, \quad m_i^{w,n} = \sum_p m_p^{w,n} w_{ip} \quad (15)$$

$$\mathbf{v}_i^{s,n} = \frac{1}{m_i^{s,n}} \sum_p m_p^s w_{ip} \left( \mathbf{v}_p^{s,n} + \mathbf{A}_p^n (\mathbf{x}_i - \mathbf{x}_p^n) \right) \quad (16)$$

$$m_i^{s,n} \theta_i^n = \sum_p m_p^s w_{ip} \theta_p^n \quad (17)$$

$$V_i^{s,n} = \sum_p J_p^{s,n} V_p^{s,0} w_{ip} \quad (18)$$

Here  $m_i^{\alpha,n}$  are grid masses,  $\mathbf{v}_i^{s,n}$  is grid solid velocity and  $\theta_i^n$  is grid temperature.  $V_i^{s,n}$  is the representative volume of grid node  $i$  which we use to update grid saturation (see Section 6.2). We update the grid water mass and temperature from Equations (5) and (9) respectively to obtain  $\tilde{m}_i^{w,n+1}$  and  $\theta_i^{n+1}$ . Details for these updates are provided in Sections 6.2 and 6.3 respectively. Once these are updated, we update the particle mass, saturation, temperature and

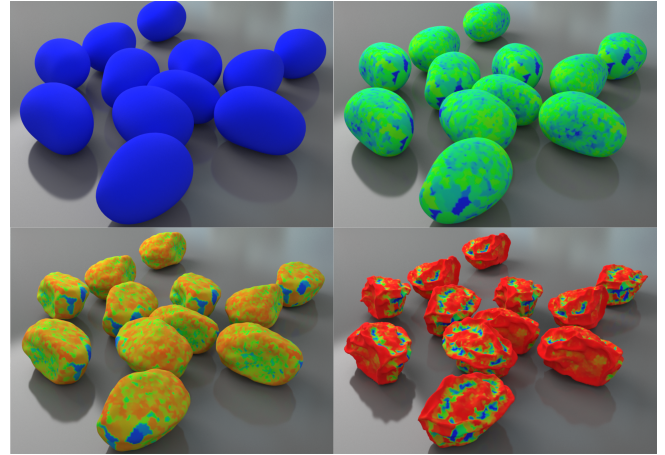


Fig. 10. **Grape Saturation View.** We visualize the saturation evolution of the grapes. The color changes from blue to green then to red with decreasing saturation.

density weighted velocity of the water relative to the solid from

$$m_p^{w,n+1} = m_p^{w,n} \left( 1 + \sum_i \frac{\Delta m_i^{w,n}}{m_i^{w,n}} w_{ip} \right) \quad (19)$$

$$s_p^{n+1} = \frac{m_p^{w,n+1}}{V_p^{s,n} \rho_0^w} \quad (20)$$

$$\theta_p^{n+1} = \theta_p^{n+1} + \sum_i \left( \theta_i^{n+1} - \theta_i^n \right) w_{ip} \quad (21)$$

$$\mathbf{q}_p^{w,n+1} = -k \theta_p^{n+1} \sum_i s_i^{n+1} \nabla w_{ip} \rho_0^w s_p^{n+1} \quad (22)$$

Here  $\Delta m_i^{w,n}$  is from Equation (25) and represents the change in the grid water mass;  $s_i^{n+1}$  in Equation (22) is the time  $t^{n+1}$  saturation of grid node  $i$  and is used to compute saturation gradient needed for the update of the density weighted velocity of the water relative to the solid. The weighting of the transfer in Equation (20) is chosen so that the total change in particle water mass is equal to the total change in grid water mass. Also, the update in Equation (21) follows from Equation (8) since  $\rho_p^{w,n+1} = \frac{m_p^{w,n+1}}{V_p^{s,n}}$ ,  $\rho_p^{w,n+1} = s_p^{n+1} \rho_0^w$  where  $\rho_0^w$  is the initial, spatially constant water mass density.

The last step in our update is to compute the updated grid solid velocity  $\tilde{\mathbf{v}}_i^{s,n+1}$  from the discrete conservation of momentum which is outlined in Section 6.4. We then compute the updated particle constant  $\mathbf{v}_p^{n+1}$  and linear  $\mathbf{A}_p^{n+1}$  velocities from  $\tilde{\mathbf{v}}_i^{s,n+1}$  using APIC [Jiang et al. 2015] and the particle positions as  $\mathbf{x}_p^{n+1} = \mathbf{x}_p^{n+1} + \Delta t \mathbf{v}_p^{n+1}$ . Lastly, a trial state of elastic stress is computed assuming no plastic flow over the time step as

$$\mathbf{F}_p^{s, Etr} = \left( \mathbf{I} + \Delta t \sum_i \tilde{\mathbf{v}}_i^{s,n+1} \nabla w_{ip} \right) \mathbf{F}_p^{E,n}$$

which is then finally projected to  $\mathbf{F}_p^{E,sn+1}$  according to plastic flow using the details in Section 6.5.



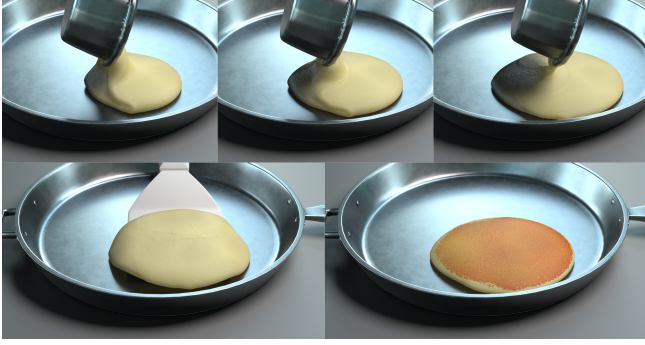


Fig. 11. **Pancake.** Batter is poured into the pan (**top row**) and the pancake is flipped after the bottom is cooked (**bottom row**). The coloring is based on temperature.

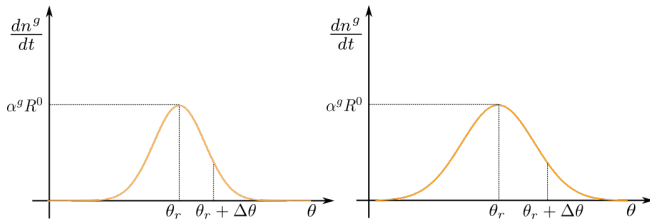


Fig. 12. **CO<sub>2</sub> Creation Rate.** The carbon dioxide creation rate is a bell-shaped curve with the peak at  $\theta_r$ . Bigger  $\Delta\theta$  results in a flatter curve (**right**).

## 6.2 Water mass update

Our discretization is based on the weak form of Equation (5). Given an arbitrary function  $u : B_{t^n}^s \rightarrow \mathbb{R}$  we can conclude

$$\begin{aligned} 0 &= \int_{B_{t^n}^s} \frac{D^s \rho^w}{Dt} u + \rho^w \nabla \cdot \mathbf{v}^s u + u \nabla \cdot \mathbf{q}^w dx \\ &= \int_{B_{t^n}^s} \frac{D^s \rho^w}{Dt} u + \rho^w \nabla \cdot \mathbf{v}^s u - \nabla u \cdot \mathbf{q}^w dx + \int_{\partial B_{t^n}^s} u \beta^w \theta s ds(x) \end{aligned}$$

by integrating by parts and using the water flux boundary condition in Section 3.2. Furthermore, we note that

$$\begin{aligned} &\int_{B_{t^n}^s} \frac{D^s \rho^w}{Dt} (\mathbf{x}, t) u(\mathbf{x}) dx \quad (23) \\ &= \int_{B_0^s} \frac{d}{dt} \rho^w(\phi^s(\mathbf{X}, t^n), t^n) u(\phi^s(\mathbf{X}, t^n)) J^s(\mathbf{X}, t^n) d\mathbf{X} \\ &\approx \int_{B_0^s} \frac{\rho^w(\phi^s(\mathbf{X}, t^{n+1}), t^{n+1}) - \rho^w(\phi^s(\mathbf{X}, t^n), t^n)}{\Delta t} u(\phi^s(\mathbf{X}, t^n)) J^s(\mathbf{X}, t^n) d\mathbf{X} \\ &= \int_{B_{t^n}^s} \frac{\tilde{\rho}^w(\mathbf{x}, t^{n+1}) - \rho^w(\mathbf{x}, t^n)}{\Delta t} u(\mathbf{x}) dx \end{aligned}$$

where  $\tilde{\rho}^w(\mathbf{x}, t^{n+1}) = \rho^w(\phi^s(\phi^{s^{-1}}(\mathbf{x}, t^n), t^{n+1}), t^{n+1})$  and  $\phi^{s^{-1}}(\mathbf{x}, t^n)$  is the inverse flow map of the solid constituents. Intuitively,  $\tilde{\rho}^w(\mathbf{x}, t^{n+1})$  is the time  $t^{n+1}$  mass density but pulled back to the time  $t^n$  spatial configuration of the solid constituents.

Letting  $u(\mathbf{x}) = \sum_j u_j N_j(\mathbf{x})$  be defined from interpolation over the grid, we can then conclude

$$\begin{aligned} 0 &= \int_{B_{t^n}^s} \left( \frac{D^s \rho^w}{Dt} + \rho^w \nabla \cdot \mathbf{v}^s \right) \sum_j u_j N_j - \sum_j u_j \nabla N_j \cdot \mathbf{q}^w dx \\ &\quad + \int_{\partial B_{t^n}^s} \sum_j u_j N_j \beta^w \theta s ds(\mathbf{x}) \\ &\approx \int_{B_{t^n}^s} \left( \frac{\tilde{\rho}^w - \rho^w}{\Delta t} + \rho^w \nabla \cdot \mathbf{v}^s \right) \sum_j u_j N_j - \sum_j u_j \nabla N_j \cdot \mathbf{q}^w dx \\ &\quad + \int_{\partial B_{t^n}^s} \sum_j u_j N_j \beta^w \theta s ds(\mathbf{x}). \end{aligned}$$

If we use the positions  $\mathbf{x}_p^n$  as quadrature points with weights  $V_p^{s,n}$  for volume integrals and  $S_p^{s,n}$  for surface integrals, then we can further approximate as

$$\begin{aligned} 0 &= \frac{1}{\Delta t} \sum_p (\tilde{\rho}_p^{w,n+1} - \rho_p^{w,n}) V_p^{s,n} \sum_j u_j w_{jp} + \\ &\quad \sum_p \rho_p^{w,n} V_p^{s,n} \nabla \cdot \mathbf{v}^s(\mathbf{x}_p^n, t^n) \sum_j u_j w_{jp} - \sum_p V_p^{s,n} \mathbf{q}_p^{w,n} \cdot \sum_j u_j \nabla w_{jp} \\ &\quad + \sum_{p \in \partial B_{t^n}^s} \beta^w \theta_p^n S_p^{s,n} \sum_j u_j w_{jp} \end{aligned}$$

where  $\tilde{\rho}_p^{w,n+1} = \tilde{\rho}(\mathbf{x}_p^n, t^{n+1})$  and  $\rho_p^{w,n} = \rho(\mathbf{x}_p^n, t^n)$ . Since this must hold for any  $u_j$ , we choose  $u_j = \delta_{ij}$  to conclude

$$\begin{aligned} 0 &= \sum_p \frac{(\tilde{\rho}_p^{w,n+1} - \rho_p^{w,n})}{\Delta t} V_p^{s,n} w_{ip} + \sum_p \rho_p^{w,n} V_p^{s,n} \nabla \cdot \mathbf{v}^s(\mathbf{x}_p^n, t^n) w_{ip} \\ &\quad - \sum_p V_p^{s,n} \mathbf{q}_p^{w,n} \cdot \nabla w_{ip} + \sum_{p \in \partial B_{t^n}^s} \beta^w \theta_p^n S_p^{s,n} w_{ip} \end{aligned}$$

for all  $i$ . Noting that  $\rho_p^{w,n} V_p^{s,n} = m_p^{w,n}$ , we can see that  $m_i^{w,n} = \sum_p \rho_p^{w,n} V_p^{s,n} w_{ip}$  from Equation (15). With this observation, we can see that the discrete equations give us the following update of the grid water mass

$$\tilde{m}_i^{w,n+1} = m_i^{w,n} + \Delta m_i^{w,n} \quad (24)$$

$$\begin{aligned} \Delta m_i^{w,n} &= -\Delta t \sum_p m_p^{w,n} \sum_k \mathbf{v}_k^{s,n} \cdot \nabla w_{kp} w_{ip} \\ &\quad + \Delta t \sum_p V_p^{s,n} \mathbf{q}_p^{w,n} \cdot \nabla w_{ip} - \Delta t \sum_{p \in \partial B_{t^n}^s} \beta^w \theta_p^n S_p^{s,n} w_{ip} \end{aligned} \quad (25)$$

where we use  $\nabla \cdot \mathbf{v}^s(\mathbf{x}_p^n, t^n) = \sum_k \mathbf{v}_k^{s,n} \cdot \nabla w_{kp}$  with  $\mathbf{v}_k^{s,n}$  from Equation (16). Here  $\tilde{m}_i^{w,n+1} = \sum_p \tilde{\rho}_p^{w,n+1} w_{ip} V_p^{s,n}$  is the updated grid mass computed from the time  $t^{n+1}$  density and time  $t^n$  grid.

We use  $\tilde{m}_i^{w,n+1}$  to compute time  $t^{n+1}$  grid node saturations from  $s_i^{n+1} = \frac{\tilde{m}_i^{w,n+1}}{\rho_0^w V_i^{s,n}}$  where  $V_i^{s,n}$  is the representative volume of the grid node from Equation (18). This follows since  $\tilde{m}_i^{w,n+1} \approx \rho_i^{w,n+1} V_i^{s,n}$  and  $\rho_i^{w,n+1} = s_i^{n+1} \rho_0^w$  where  $\rho_0^w$  is the initial spatially constant mass density of water.



Fig. 13. **Dehydrating Apple Slices.** Apple slices with various initial thickness are dried at constant temperature. The skin wrinkles and the slices buckle up and experience significant volume loss.

### 6.3 Heat transfer update

The discretization of Equation (9) is similar to that of the water mass update and [Stomakhin et al. 2014]. We summarize it as

$$\begin{aligned} \frac{1}{\Delta t} \left( \alpha m_i^{s,n} \theta_i^{n+1} - \alpha m_i^{s,n} \theta_i^n \right) &= \sum_{p \in \partial B_{i,n}^s} S_p \beta (\theta_p^{n+\gamma} - \theta_{out}) w_{ip} \\ &+ \sum_p V_p^{s,n} \left( \sum_j \theta_j^{n+\gamma} \nabla w_{jp} \cdot \nabla w_{ip} - \frac{c \lambda \theta_p}{J_p^{s,n}} \sum_k \mathbf{v}_k^{s,n} \cdot \nabla w_{kp} w_{ip} \right) \end{aligned}$$

where  $\gamma = 0$  for forward Euler and  $\gamma = 1$  for backward Euler. We note that the last term expresses the effects of thermal expansion, which were not considered in [Stomakhin et al. 2014].

### 6.4 Grid momentum update

We discretize Equation (7) in a variational MPM manner to derive the grid momentum and velocity update. Multiplying Equation (7) by an arbitrary function  $\mathbf{u}(\mathbf{x}) = \sum_i \mathbf{u}_i N_i(\mathbf{x})$ , taking an arbitrary  $\mathbf{u}_i$ , using  $\rho \mathbf{v} = \rho \mathbf{v}^s + \mathbf{q}^w$  and treating the  $\frac{D^s \rho \mathbf{v}^s}{Dt}$  term as in Equation (23), we can conclude

$$\begin{aligned} \int_{B_{i,n}^s} \left( \frac{\tilde{\rho} \tilde{\mathbf{v}}^s + \tilde{\mathbf{q}}^w - \rho \mathbf{v}^s - \mathbf{q}^w}{\Delta t} + (\rho \mathbf{v}^s + \mathbf{q}^w) \nabla \cdot \mathbf{v}^s \right) N_i dx &= \\ \int_{B_{i,n}^s} \mathbf{f} N_i - (\boldsymbol{\sigma} - \mathbf{v}^w \otimes \mathbf{q}^w) \nabla N_i dx + \int_{\partial B_{i,n}^s} (\mathbf{t} + \beta^w \theta_s \mathbf{v}^w) N_i ds(\mathbf{x}) \end{aligned}$$

with the convention that function  $\tilde{f}$  is inherited as a function over  $B_{i,n}^s$  as  $\tilde{f} = f(\boldsymbol{\phi}^s(\boldsymbol{\phi}^{s^{-1}}(\mathbf{x}, t^n), t^{n+1}), t^{n+1})$  for functions  $f$  defined over  $B_{i,t^{n+1}}^s$ .

Using the particle positions  $\mathbf{x}_p^n$  as the quadrature points with weights  $V_p^{s,n} = J_p^{s,n} V_p^{s,0}$  we conclude

$$\sum_p \tilde{\rho}_p^{n+1} \tilde{\mathbf{v}}_p^{s,n+1} w_{ip} V_p^{s,n} + \tilde{\mathbf{q}}_p^{w,n+1} w_{ip} V_p^{s,n} = \quad (26)$$

$$\sum_p \rho_p^n \mathbf{v}_p^{s,n} w_{ip} V_p^{s,n} + \mathbf{q}_p^{w,n} w_{ip} V_p^{s,n} + \Delta t \Delta(\rho \mathbf{v}^s)_i \quad (27)$$

$$\begin{aligned} \Delta(\rho \mathbf{v}^s)_i &= - \sum_p \left( \rho_p^n \mathbf{v}_p^{s,n} + \mathbf{q}_p^{w,n} \right) w_{ip} V_p^{s,n} \sum_k \mathbf{v}_k^{s,n} \cdot \nabla w_{kp} \\ &+ \sum_p \mathbf{f}_p^n w_{ip} - \left( \boldsymbol{\sigma}_p^n - \mathbf{v}_p^{w,n} \otimes \mathbf{v}_p^{s,n} \right) V_p^{s,n} \nabla w_{ip} \\ &+ \sum_{p \in \partial B_{i,n}^s} S_p w_{ip} \left( \mathbf{t}_p^n + \beta^w \theta_p^n s_p^n \mathbf{v}_p^{w,n} \right) \end{aligned}$$

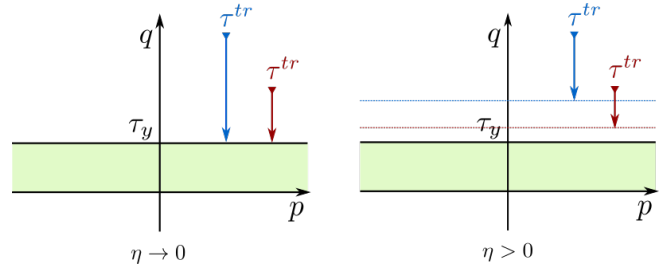


Fig. 14. **Return Mapping.** The projections of two trial state stresses are illustrated. With  $\eta \rightarrow 0$  (left) the trial stresses are projected onto the yield surface; with  $\eta > 0$  (right) the projections have the same directions but only a portion of the distance.

where  $\tilde{\rho}_p^{n+1} = \tilde{\rho}(\mathbf{x}_p^n, t^{n+1})$ ,  $\rho_p^n = \rho(\mathbf{x}_p^n, t^n) = s_p^n \rho_0^w + \frac{m_p^s}{V_p^{s,n}}$ , and  $\tilde{\mathbf{q}}_p^{w,n+1} = \tilde{\mathbf{q}}^w(\mathbf{x}_p^n, t^{n+1})$  is from Equation (22). We note that Equation (22) does not have a tilde superscript because it will be used with particle positions  $\mathbf{x}_p^{n+1}$  at the end of the time step, whereas  $\tilde{\mathbf{q}}^w(\mathbf{x}_p^n, t^{n+1})$  is the same value, but being used with time  $t^n$  positions  $\mathbf{x}_p^n$ . Noting that  $\sum_p \rho_p^n \mathbf{v}_p^{s,n} w_{ip} V_p^{s,n} \approx m_i^n \mathbf{v}_i^{s,n}$ , we use Equation (26) as an update for the solid grid velocity as

$$\tilde{\mathbf{v}}_i^{s,n+1} = \frac{1}{\tilde{m}_i^{n+1}} \left( m_i^n \mathbf{v}_i^{s,n} + \Delta t \Delta(\rho \mathbf{v}^s)_i + \sum_p \Delta \mathbf{q}_p w_{ip} V_p^{s,n} \right),$$

where  $\Delta \mathbf{q}_p = \mathbf{q}_p^{w,n} - \tilde{\mathbf{q}}_p^{w,n+1}$ . We note that  $\tilde{m}_i^{n+1} = \tilde{m}_i^{w,n+1} + m_i^{s,n}$  where  $\tilde{m}_i^{w,n+1}$  is from Equation (25) and  $m_i^{s,n}$  is from Equation (15). We note that  $m_i^{s,n}$  does not change with time in this update since we track the motion of the solid constituents in a Lagrangian manner. Lastly we note that  $\boldsymbol{\sigma}_p^n$  is a sum of the stresses in Equations (10), (13) and (14).

### 6.5 Return mapping

Plasticity is applied by first assuming all deformation is elastic and getting a trial stress  $\boldsymbol{\tau}^{s,Etr} = \frac{\partial \Psi}{\partial \mathbf{F}^{s,E}}(\mathbf{F}_p^{s,Etr}) \mathbf{F}_p^{s,Etr-T}$  from Equation (10). Following [Simo and Meschke 1993] we can write the return mapping discretization of Equations (11) and (12) in the form of a constrained minimization problem:

$$\boldsymbol{\tau}^{s,En+1} = \operatorname{argmin}_{\boldsymbol{\tau} \in \mathbb{E}} \frac{1}{2} (\boldsymbol{\tau}^{s,Etr} - \boldsymbol{\tau})^T \mathbf{C}^{-1} (\boldsymbol{\tau}^{s,Etr} - \boldsymbol{\tau})$$

where  $\mathbb{E}$  stands for the elastic region bounded by the yield surface with  $f(\boldsymbol{\tau}) \leq 0$ , and  $\mathbf{C}$  is the elastic tensor from  $\boldsymbol{\tau}^{s,E} = \mathbf{C} \boldsymbol{\epsilon}$ . For viscoplasticity the hard constraint that the minimization be over  $\mathbb{E}$  is replaced by a penalty term from viscosity and we have the unconstrained minimization

$$\boldsymbol{\tau}^{s,En+1} = \operatorname{argmin}_{\boldsymbol{\tau}} \frac{1}{2} (\boldsymbol{\tau}^{s,Etr} - \boldsymbol{\tau})^T \mathbf{C}^{-1} (\boldsymbol{\tau}^{s,Etr} - \boldsymbol{\tau}) + \frac{\Delta t}{\eta} g(f(\boldsymbol{\tau}))$$

where  $\Delta t$  is the time step. To solve the minimization, we take the derivative with respect to  $\boldsymbol{\tau}$  and set it equal to zero. The elastic Hencky strain can then be obtained trivially as  $\boldsymbol{\epsilon}_p^{s,En+1} = \mathbf{C}^{-1} \boldsymbol{\tau}^{s,En+1}$ . Then the new deformation gradient can be obtained using  $\mathbf{F}_p^{s,En+1} = \mathbf{U}_p^{s,En+1} \exp \left( \boldsymbol{\Lambda}_p^{s,En+1} \right) \left( \mathbf{V}_p^{s,En+1} \right)^T$ , where  $\boldsymbol{\Lambda}_p^{s,En+1}$  is derived from



Fig. 15. **Bread.** The **top left** shows raw dough, one is left intact and the other two have different slits on top. When baked(**right**) the breads expand in size and the slits open up. The bread without an initial slit also cracked on the top surface.

the eigen decomposition  $\epsilon_p^{s,En+1} = \mathbf{U}_p^{s,En+1} \Lambda_p^{s,En+1} (\mathbf{U}_p^{s,En+1})^T$ , and  $\mathbf{U}_p^{s,En+1}$  and  $\mathbf{V}_p^{s,En+1}$  are derived from the singular value decomposition  $\mathbf{F}_p^{s,Etr} = \mathbf{U}_p^{s,En+1} \Sigma_p^{s,Etr} (\mathbf{V}_p^{s,En+1})^T$  respectively. This can be solved efficiently in terms of the eigenvalues of  $\tau^{s,En+1}$  and is similar to the methods in [Gaume et al. 2018; Klár et al. 2016]. We refer the reader to [Ding et al. 2019] for specific details and illustrate the return mapping in the zero porosity case in Figure 14.

## 6.6 Boundary conditions

We need to treat discrete boundary integrals in various terms in Sections 6.2, 6.3 and 6.4. We define the set of boundary particles  $p \in \partial B_{t^n}^s$  as follows. First, we define a grid node  $\mathbf{x}_i$  to be on the boundary if it has a neighboring grid node with zero mass  $m_i^n$ . Each boundary grid node then appends the particle closest it to the set of boundary particles. We assume that each boundary particle owns a local spherical region, the volume of which is its representative volume, i.e. we have

$$V_p^{s,n} = \frac{4\pi}{3} (r_p^{s,n})^3$$

where  $r_p^{s,n}$  is the approximated radius of the sphere. The surface area of the particle is then computed as  $\pi (r_p^{s,n})^2$  which can be written in the form

$$S_p^{s,n} \approx \left(\frac{9\pi}{16} (V_p^{s,n})^2\right)^{1/3}$$

This serves as the quadrature weight at position  $\mathbf{x}_p^n$  when needed for the surface integral approximation in Sections 6.2, 6.3 and 6.4.

## 7 RESULTS

We demonstrate the efficacy of our model with several examples that illustrate the baking and cooking process of mixtures with various textures, and show our method is able to capture some of the key visual effects. Runtime performance is listed in Table 1 and the parameters used are listed in Table 2. Simulations are run on an Intel Xeon E5-2687W v4 system with 48 threads and an Intel Xeon X5690 with 12 threads. We report the computation runtime in terms of average seconds per frame. Particle counts are given for each example. In general, we chose  $\Delta x$  so that there are approximately six particles per grid cell in the initial state.  $\Delta t$  is chosen in an adaptive

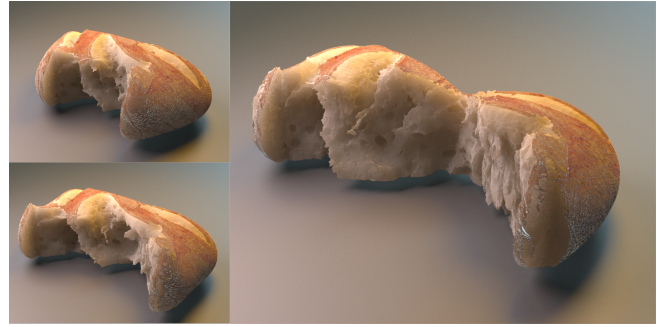


Fig. 16. **Tearing Bread.** We demonstrate tearing of the bread after baking. Surface meshes are generated through the mesh post-process of [Wang et al. 2019] and are used for rendering.

manner restricted by a CFL condition that no particles are allowed to travel more than a portion of  $\Delta x$  in each time step.

### 7.1 Effect of Water Loss

We demonstrate the effect of the water content evolution with the simple scenario of fruit dehydration. In Figure 13 we simulate the process of drying apple slices on a rack. The flesh part is visco-elastoplastic with zero porosity and high initial saturation. Fruit skin contains much less moisture than the flesh and thus is modelled as a two-dimensional elastic sheet with zero water content. We use the Lagrangian energy approach of Jiang et al. [Jiang et al. 2015] to couple the elastic surface. Ambient temperature is set to be constant. The apple skin wrinkles and the slices twist and buckle out of the plane, matching real life observations. Figure 9 shows drying grapes with a similar setting. Water evaporates through the skin of the grapes and results in shrinkage. The skin again has a much lower water content and thus experiences less shrinkage, causing the many wrinkles and folds as it tries to retain its surface area while losing volume. Figure 10 visualizes the saturation change of the grapes, coloring from blue to green and finally red for decreasing saturation.

### 7.2 Temperature Dependent Plasticity

Figure 5 depicts a marshmallow roasting on an electric stove. The marshmallow is modeled as a homogeneous mixture in its initial state. Our model achieves the visual effect of an initial volume gain from the thermal expansion followed by a slightly burnt crust with a gooey center by using temperature dependent plasticity parameters. The coloration of the marshmallow is also rendered according to temperature. Our model can successfully track the significant texture changes of the mixture during the baking process. Figure 4 demonstrates baking of a lava cake in a ramekin. The initial batter is again a homogeneous mixture, and our model captures its transition to a cake with fully baked exterior and the characteristic molten center. We also show pouring liquid pancake batter into a frying pan followed by cooking to get a soft elastic pancake that can be flipped, see Figure 11. In Figure 3, we test baking the same cookie dough with varying oven settings. The three cookies on the bottom row from left to right are baked under decreasing temperatures. With the temperature being too high, the dough gets heated up very



Fig. 17. **Cookie comparison.** Cookies are baked with decreasing temperature from left to right. **Top row:** Stomakhin et al. [2014] and **bottom row:** our model.

quickly, so the cookie does not get enough time to spread, and the dough baked under low temperature has the opposite issue. This comparison matches closely to real life observations. The temperature is also visualized in the right column. The color varies from blue to green then to red with increasing temperature.

### 7.3 Leavening

The influence of the chemical reaction aspect of our model is best illustrated by baking cookies with varying amount of leavening agents, as is shown in Figure 3. The top row of the cookies from left to right contains decreasing amount of leaveners and are baked under the same temperature. Too much leavening agent would produce a very tall cookie, while no leavener at all yields a flat one. The  $CO_2$  creation and the corresponding pressure is the main contribution to the rising and expansion in our muffin (Figure 2), lava cake (Figure 4) and bread (Figure 15) examples.

### 7.4 Fracture

Our simulations are particle-based from their MPM conception, but for simulations with fracture, we construct a reference tetrahedron mesh in the initial state for rendering purposes and adopt the post-processing techniques from [Wang et al. 2019] to obtain clean and consistent surfacing of the fractured material. The reference meshes are generated with TetWild [Hu et al. 2018]. We demonstrate these effects with tearing examples in Figure 2 and Figure 16. By modeling the combined effect of water diffusion, temperature change and chemical leavening, our method is able to achieve visually realistic baking and tearing of a muffin, see Figure 2. Drawing slits on the bread dough helps with the rising during baking as well as the formation of a nice crust. In Figure 15 we compare the baking process of bread with and without scoring the surface beforehand. Notice how the bread cracks in a more controlled and appealing manner when there are slits on the surface.

### 7.5 Comparisons

In this section we demonstrate the strength of our model by comparing with some existing techniques and/or simpler approaches. We first compare our model against [Stomakhin et al. 2014] with baking

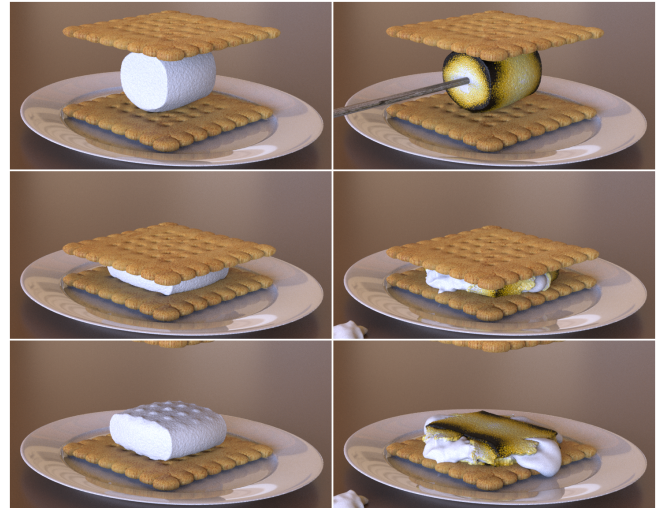


Fig. 18. **Marshmallow comparison.** The marshmallow (**left column**) has a uniform texture, it gets pressed down by the cracker then springs back a little after the pressure is released; after roasting (**right column**) a crust forms on the exterior and the interior is melted. When pressed the crust breaks and the inside oozes out.

cookies and heating a pancake. Figure 17 shows baking cookies under different temperatures. Our model is able to produce more realistic domed cookies with the addition of chemical reactions, while the model from Stomakhin et al. [2014] produces cookies with flat or even dented tops. Figure 19 depicts pouring pancake batter into a pan, due to lack of viscoplasticity the bottom pancake batter experiences undesirable and hard-to-control spreading, while our model successfully captures a more realistic thick batter texture. We further demonstrate the thermal effect of our model with Figure 18, where we press the marshmallow between two crackers then release, the different textures of the marshmallow before and after roasting is clearly illustrated. The marshmallow without thermal effects is similar to the model in [Yue et al. 2015] since they do not address the thermal change of the material during roasting. The higher complexity of our model allow us to capture more details in the baking and cooking processes.

## 8 DISCUSSION AND LIMITATIONS

We demonstrate that our method is capable of recreating a number of representative baking and cooking examples. However, our approach has a number of limitations. First, we simplify the mixture of non-water or gas constituents to be represented by a single phase. A more principled approach could be taken by including each of the flour, fat, egg, etc species in a mixture model. This would undoubtedly allow for a wider range of dough rheologies. Also, our kinematic assumption that the gas does not move relative to the solid mixture precludes diffusion of the gas through the dough. Lastly, due to the high-complexity of our coupled porous thermomechanical model, we did not investigate fully implicit treatment of water diffusion and material stiffness. While we did not experience a need for excessively small time steps given the low stiffness and

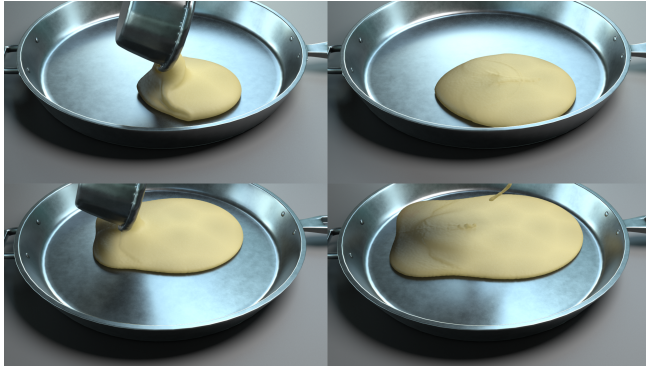


Fig. 19. **Pancake comparison.** Pancake batter is poured from a pitcher into a pan. **Top row:** our model and **bottom row:** Stomakhin et al. [2014].

Table 1. All simulations were run on either Intel Xeon E5-2690 v4 system with 48 threads (Machine 1) or Intel Xeon X5690 with 12 threads (Machine 2). Simulation time is measured in average seconds per frame. Element denotes number of tetrahedra in the volumetric mesh or number of triangles in the surface mesh when applicable. Particle denotes the total number of MPM particles in the simulation.

Simulation	Time	Machine	Element	Particle
Bread bake (Fig 15)	20	1	3.2M	620K
Bread tear (Fig 16)	12	1	3.2M	620K
Cookies (per cookie) (Fig 3)	48	1	6.3M	1.2M
Pancake (Fig 11)	25	1	N/A	1M
Dough (per dough) (Fig 6)	11	1	2.7M	500K
Grape dehydration (Fig 9)	100	1	96K (surface)	346K
Muffin (Fig 2)	150	2	5.4M	1.2M
Apple dehydration (Fig 13)	25	2	76K (surface)	541K
Lava cake (Fig 4)	145	2	10.8M	2M
S'more (Fig 5)	130	2	N/A	1.1M

Table 2. We provide a list of the main parameters used in the simulations. The changes in yield stress and viscosity follow the curves in Figure 8.

Parameters	Bread	Cookies	Pancake	Muffin	Lava cake	S'more
Young's	2e3	4e4	4e4	4e4	2e3	4e4
Density	500	500	500	500	500	500
Specific heat	20	20	20	20	20	200
$\kappa$	100	30	200	100	250	150
Yield surface	20	20	2	2	1	200
$\eta$	500	500	50	10	10	60
$\theta_0$	1	0	0	0	0	0
Ambient $\theta$	100	150	N/A	105	100	400
Ambient $\beta$	100	25	N/A	75	110	120
Conductive $\theta$	100	150	200	105	100	400
Conductive $\beta$	150	35	70	150	110	360
$\alpha^g$	0.03	0.015	0.02	0.03	0.025	N/A
$\theta_r$	40	40	30	40	20	N/A
$\Delta\theta$	20	20	20	20	10	N/A
$\Delta x$	0.01	0.0051	0.016	0.008	0.0075	0.003
$\Delta t$	4e-3	3e-4	5e-4	2e-4	4e-4	2e-4

diffusion time scales in the materials considered, baked goods with faster water diffusion rates and material wave speeds could benefit from a fully implicit discretization.

ACKNOWLEDGMENTS

The work is supported by NSF CCF-1422795, Intel STC-Visual Computing Grant (20112360) as well as a gift from Adobe Inc.

REFERENCES

K. Abe, K. Soga, and S. Bandara. 2014. Material Point Method for Coupled Hydromechanical Problems. *J Geotech Geoenv Eng* 140, 3 (2014), 04013033.

R. Atkin and R. Craine. 1976. Continuum theories of mixtures: basic theory and historical development. *Quart J Mech App Math* 29, 2 (1976), 209–244.

S. Bandara, A. Ferrari, and L. Laloui. 2016. Modelling landslides in unsaturated slopes subjected to rainfall infiltration using material point method. *Int J Num Anal Meth Geomech* 40, 9 (2016), 1358–1380.

S. Bandara and K. Soga. 2015. Coupling of soil deformation and pore fluid flow using material point method. *Comp Geotech* 63 (2015), 199–214.

K. Bao, X. Wu, H. Zhang, and E. Wu. 2010. Volume fraction based miscible and immiscible fluid animation. *Comp Anim Virtual Worlds* 21, 3-4 (2010), 401–410.

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

H. Barreiro, I. García-Fernández, I. Alduán, and M.-A. Otaduy. 2017. Conformation constraints for efficient viscoelastic fluid simulation. *ACM Transactions on Graphics (TOG)* 36, 6 (2017), 221.

M. Biot. 1941. General theory of three-dimensional consolidation. *J App Phys* 12, 2 (1941), 155–164.

J. Blutinger, Y. Meijers, P. Chen, C. Zheng, E. Grinspun, and H. Lipson. 2019. Characterization of CO2 laser browning of dough. *Innovative Food Science & Emerging Technologies* 52 (2019), 145–157.

J. Bonet and R. Wood. 2008. *Nonlinear continuum mechanics for finite element analysis*. Cambridge University Press.

J. Brackbill and H. Ruppel. 1986. FLIP: A method for adaptively zoned, Particle-In-Cell calculations of fluid flows in two dimensions. *J Comp Phys* 65 (1986), 314–343.

B. Broyart and G. Trystram. 2002. Modelling heat and mass transfer during the continuous baking of biscuits. *Journal of Food Engineering* 51, 1 (2002), 47–57.

M. Carlson, P. J. Mucha, R. B. Van Horn III, and G. Turk. 2002. Melting and flowing. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation*. ACM, 167–174.

P. Chen, J. Blutinger, Y. Meijers and C. Zheng, E. Grinspun, and H. Lipson. 2019. Visual modeling of laser-induced dough browning. *Journal of Food Engineering* 243 (2019), 9–21.

B. De Cindio and S. Corraera. 1995. Mathematical modelling of leavened cereal goods. *Journal of Food Engineering* 24, 3 (1995), 379–403.

G. Daviet and F. Bertails-Descoubes. 2016. A Semi-implicit Material Point Method for the Continuum Simulation of Granular Materials. *ACM Trans Graph* 35, 4 (2016), 102:1–102:13.

F. Debaste, A. Léonard, V. Halloin, and B. Haut. 2010. Microtomographic investigation of a yeast grain porous structure. *Journal of Food Engineering* 97, 4 (April 2010), 526–532. <https://doi.org/10.1016/j.jfoodeng.2009.11.012>

M. Ding, X. Han, S. Wang, T. Gast, and J. Teran. 2019. *Supplementary Technical Document*. Technical Report.

H. Faridi and J. Faubion. 2012. *Dough rheology and baked product texture*. Springer Science & Business Media.

Y. Fei, C. Batty, E. Grinspun, and C. Zheng. 2018. A multi-scale model for simulating liquid-fabric interactions. *ACM Trans Graph* 37, 4 (2018), 51:1–51:16. <https://doi.org/10.1145/3197517.3201392>

A. Fick. 1855. On liquid diffusion. *London, Edinburgh, and Dublin Philos Mag J Sci* 10 (1855), 30–39.

M. Gao, A. Pradhana, X. Han, Q. Guo, G. Kot, E. Sifakis, and C. Jiang. 2018a. Animating fluid sediment mixture in particle-laden flows. *ACM Trans Graph* 37, 4 (2018), 149:1–149:11. <https://doi.org/10.1145/3197517.3201309>

M. Gao, X. Wang, Kui K.Wu, A. Pradhana, E. Sifakis, C. Yuksel, and C. Jiang. 2018b. GPU optimization of material point methods. In *SIGGRAPH Asia 2018 Technical Papers (SIGGRAPH Asia '18)*. ACM, New York, NY, USA, Article 254, 12 pages. <https://doi.org/10.1145/3272127.3275044>

J. Gaume, T. Gast, J. Teran, A. van Herwijnen, and C. Jiang. 2018. Dynamic anticrack propagation in snow. *Nature Com* 9, 1 (2018), 3047.

D. Gerszewski, H. Bhattacharya, and A. Bargteil. 2009. A point-based method for animating elastoplastic solids. In *Proc 2009 ACM SIGGRAPH/Eurograph Symp Comp Anim*. ACM, 133–138. <https://doi.org/10.1145/1599470.1599488>

T. Goktekin, A. Bargteil, and J. O'Brien. 2004. A Method for Animating Viscoelastic Fluids. *ACM Trans Graph* 23, 3 (2004), 463–468.

O. Gonzalez and A. Stuart. 2008. *A first course in continuum mechanics*. Cambridge University Press.

V. Guillard, B. Broyart, S. Guilbert, C. Bonazzi, and N. Gontard. 2004. Moisture diffusivity and transfer modelling in dry biscuit. *Journal of Food Engineering* 64, 1 (2004), 81–87.

- Q. Guo, X. Han, C. Fu, T. Gast, R. Tamstorf, and J. Teran. 2018. A material point method for thin shells with frictional contact. *ACM Trans Graph* 37, 4 (2018), 147. <https://doi.org/10.1145/3197517.3201346>
- M. Haghi and L. Anand. 1992. A constitutive model for isotropic, porous, elastic-viscoplastic metals. *Mechanics of Materials* 13, 1 (March 1992), 37–53. [https://doi.org/10.1016/0167-6636\(92\)90034-B](https://doi.org/10.1016/0167-6636(92)90034-B)
- F. Harlow. 1964. The particle-in-cell method for numerical solution of problems in fluid dynamics. *Meth Comp Phys* 3 (1964), 319–343.
- X. He, H. Wang, F. Zhang, H. Wang, G. Wang, K. Zhou, and E. Wu. 2015. Simulation of Fluid Mixing with Interface Control. In *Proc ACM SIGGRAPH / Eurograph Symp Comp Anim*. ACM, 129–135.
- Y. Hu, Q. Zhou, X. Gao, A. Jacobson, D. Zorin, and D. Panozzo. 2018. Tetrahedral Meshing in the Wild. *ACM Trans. Graph.* 37, 4, Article 60 (July 2018), 14 pages. <https://doi.org/10.1145/3197517.3201353>
- I. Jassim, D. Stolle, and P. Vermeer. 2013. Two-phase dynamic analysis by material point method. *Int J Num Anal Meth Geomech* 37, 15 (2013), 2502–2522. <https://doi.org/10.1002/nag.2146>
- C. Jiang, T. Gast, and J. Teran. 2017a. Anisotropic elastoplasticity for cloth, knit and hair frictional contact. *ACM Trans Graph* 36, 4 (2017), 152.
- C. Jiang, C. Schroeder, A. Selle, J. Teran, and A. Stomakhin. 2015. The Affine Particle-In-Cell Method. *ACM Trans Graph* 34, 4 (2015), 51:1–51:10.
- C. Jiang, C. Schroeder, and J. Teran. 2017b. An angular momentum conserving affine-particle-in-cell method. *J Comp Phys* 338 (2017), 137 – 164.
- N. Kang, J. Park, J. Noh, and S. Shin. 2010. A Hybrid Approach to Multiple Fluid Simulation using Volume Fractions. *Comp Graph Forum* 29, 2 (2010), 685–694.
- R. Keiser, B. Adams, D. Gasser, P. Bazzi, P. Dutré, and M. Gross. 2005. A unified lagrangian approach to solid-fluid animation. In *Proceedings Eurographics/IEEE VGTC Symposium Point-Based Graphics, 2005*. IEEE, 125–148.
- G. Klár, T. Gast, A. Pradhana, C. Fu, C. Schroeder, C. Jiang, and J. Teran. 2016. Drucker-prager Elastoplasticity for Sand Animation. *ACM Trans Graph* 35, 4 (2016), 103:1–103:12.
- S. Liu, Z. Wang, Z. Gong, and Q. Peng. 2008. Simulation of Atmospheric Binary Mixtures Based on Two-fluid Model. *Graph Mod* 70, 6 (2008), 117–124.
- Y. Liu, X. Yang, Y. Cao, Z. Wang, B. Chen, J. Zhang, and H. Zhang. 2015. Dehydration of core/shell fruits. *Computers & Graphics* 47 (April 2015), 68–77. <https://doi.org/10.1016/j.cag.2014.11.003>
- F. Losasso, G. Irving, E. Guendelman, and R. Fedkiw. 2006. Melting and burning solids into liquids and gases. *IEEE Transactions on Visualization and Computer Graphics* 12, 3 (2006), 343–352.
- F. Losasso, J. Talton, N. Kwatra, and R. Fedkiw. 2008. Two-Way Coupled SPH and Particle Level Set Fluid Simulation. *IEEE Trans Visu Comp Graph* 14, 4 (2008), 797–804.
- N. Maréchal, E. Guérin, E. Galin, S. Mérillou, and N. Mérillou. 2010. Heat transfer simulation for modeling realistic winter sceneries. In *Computer Graphics Forum*, Vol. 29. Wiley Online Library, 449–458.
- V. Mihalef, D. Metaxas, and M. Sussman. 2009. Simulation of two-phase flow with sub-scale droplet and bubble effects. *Comp GraphForum* 28, 2 (2009), 229–238.
- M. Müller, B. Heidelberger, M. Teschner, and M. Gross. 2005. Meshless deformations based on shape matching. In *ACM transactions on graphics (TOG)*, Vol. 24. ACM, 471–478.
- M. Müller, R. Keiser, A. Nealen, M. Pauly, M. Gross, and M. Alexa. 2004. Point based animation of elastic, plastic and melting objects. In *Proc ACM SIGGRAPH/Eurograph Symp Comp Anim*. 141–151. <https://doi.org/10.1145/1028523.1028542>
- V. Nicolas, P. Salagnac, P. Glouannec, J.-P. Ploteau, V. Jury, and L. Boillereaux. 2014. Modelling heat and mass transfer in deformable porous media: Application to bread baking. *Journal of Food Engineering* 130 (June 2014), 23–35. <https://doi.org/10.1016/j.foodeng.2014.01.014>
- M. Nielsen and O. Osterby. 2013. A Two-continua Approach to Eulerian Simulation of Water Spray. *ACM Trans Graph* 32, 4 (2013), 67:1–67:10.
- D. Ram, T. Gast, C. Jiang, C. Schroeder, A. Stomakhin, J. Teran, and P. Kavehpour. 2015. A material point method for viscoelastic fluids, foams and sponges. In *Proc ACM SIGGRAPH/Eurograph Symp Comp Anim*. 157–163.
- B. Ren, Y. Jiang, C. Li, and M. Lin. 2015. A simple approach for bubble modelling from multiphase fluid simulation. *Comp Vis Media* 1, 2 (2015), 171–181.
- B. Ren, C. Li, X. Yan, M. Lin, J. Bonet, and S. Hu. 2014. Multiple-Fluid SPH Simulation Using a Mixture Model. *ACM Trans Graph* 33, 5 (2014), 171:1–171:11.
- K. Roscoe and J. Burland. 1968. On the generalized stress-strain behaviour of wet clay. *Eng Plast* (1968), 535–609.
- M. Sakin-Yilmazer, T. Kemerli, H. Isleroglu, O. Ozdestan, G. Guven, A. Uren, and F. Kaymak-Ertekin. 2013. Porous media based model for deep-fat vacuum frying potato chips. *Journal of Food Engineering* 110, 3 (Dec. 2013), 483–489. <https://doi.org/10.1016/j.foodeng.2011.12.024>
- A. Shah. 2007. Anyone Can Cook—Inside Ratatouille’s Kitchen. *SIGGRAPH 2007 Course Notes* 30 (2007), 58.
- J. Simo and G. Meschke. 1993. A new class of algorithms for classical plasticity extended to finite strains. Application to geomaterials. *Comput Mech* 11, 4 (1993), 253–278.
- A. Stomakhin, C. Schroeder, L. Chai, J. Teran, and A. Selle. 2013. A Material Point Method for snow simulation. *ACM Trans Graph* 32, 4 (2013), 102:1–102:10.
- A. Stomakhin, C. Schroeder, C. Jiang, L. Chai, J. Teran, and A. Selle. 2014. Augmented MPM for phase-change and varied materials. *ACM Trans Graph* 33, 4 (2014), 138:1–138:11.
- D. Sulsky, Z. Chen, and H. Schreyer. 1994. A particle method for history-dependent materials. *Comp Meth App Mech Eng* 118, 1 (1994), 179–196.
- T. Takahashi, H. Fujii, A. Kunimatsu, K. Hiwada, T. Saito, K. Tanaka, and H. Ueki. 2003. Realistic Animation of Fluid with Splash and Foam. *Comp Graph Forum* 22, 3 (2003), 391–400.
- A. P. Tampubolon, T. Gast, G. Klár, C. Fu, J. Teran, C. Jiang, and K. Museth. 2017. Multi-species simulation of porous sand and water mixtures. *ACM Trans Graph* 36, 4 (2017).
- D. Terzopoulos, J. Platt, and K. Fleischer. 1991. Heating and melting deformable models. *The Journal of Visualization and Computer Animation* 2, 2 (1991), 68–73.
- M. Teschner, B. Heidelberger, M. Muller, and M. Gross. 2004. A versatile and robust method for geometrically complex deformable solids. In *Proceedings Computer Graphics International, 2004*. IEEE, 312–319.
- K. Thorvaldsson and C. Skjöldebrand. 1998. Water diffusion in bread during baking. *LWT-Food Science and Technology* 31, 7-8 (1998), 658–663.
- N. Thürey, F. Sadlo, S. Schirm, M. Müller-Fischer, and M. Gross. 2007. Real-time Simulations of Bubbles and Foam Within a Shallow Water Framework. In *Proc ACM SIGGRAPH/Eurograph Symp Comp Anim*. Eurographics Association, 191–198.
- F. Vanin, T. Lucas, and G. Trystram. 2009. Crust formation and its role during bread baking. *Trends in Food Science & Technology* 20, 8 (2009), 333–343.
- S. Wang, M. Ding, T. Gast, L. Zhu, S. Gagniere, C. Jiang, and J. Teran. 2019. Simulation and Visualization of Ductile Fracture with the Material Point Method. *Proceedings of the ACM on Computer Graphics and Interactive Techniques* 2, 2, 18.
- X. Wei, W. Li, and A. Kaufman. 2003. Melting and Flowing of Viscous Volumes. In *Proceedings of the 16th International Conference on Computer Animation and Social Agents (CASA 2003)*. IEEE Computer Society, 54.
- M. Wicke, D. Ritchie, B. Klingner, S. Burke, J. Shewchuk, and J. O’Brien. 2010. Dynamic local remeshing for elastoplastic simulation. *ACM Trans Graph* 29, 4 (2010), 49:1–11.
- C. Wojtan and G. Turk. 2008. Fast viscoelastic behavior with thin features. *ACM Trans Graph* 27, 3 (2008), 1–8. <https://doi.org/10.1145/1360612.1360646>
- L. Yang, S. Li, A. Hao, and H. Qin. 2014. Hybrid Particle-grid Modeling for Multi-scale Droplet/Spray Simulation. *Comp Graph Forum* 33, 7 (2014), 199–208.
- T. Yang, J. Chang, M. C. Lin, R. R. Martin, J. J. Zhang, and S.-M. Hu. 2017. A unified particle system framework for multi-phase, multi-material visual simulations. *ACM Transactions on Graphics (TOG)* 36, 6 (2017), 224.
- T. Yang, J. Chang, B. Ren, M. Lin, J. Zhang, and S. Hu. 2015. Fast Multiple-fluid Simulation Using Helmholtz Free Energy. *ACM Trans Graph* 34, 6 (2015), 201:1–201:11.
- A. Yerro, E. Alonso, and N. Pinyol. 2015. The material point method for unsaturated soils. *Geotechnique* 65, 3 (2015), 201–217.
- Y. Yue, B. Smith, C. Batty, C. Zheng, and E. Grinspun. 2015. Continuum foam: a material point method for shear-dependent flows. *ACM Trans Graph* 34, 5 (2015), 160:1–160:20.
- Y. Yue, B. Smith, P. Chen, M. Chantharayukhonthorn, K. Kamrin, and E. Grinspun. 2018. Hybrid grains: adaptive coupling of discrete and continuum simulations of granular media. *ACM Trans Graph* 37, 6 (2018), 283:1–283:19.
- H. Zhang, K. Wang, and Z. Chen. 2009. Material point method for dynamic analysis of saturated porous media under external contact/impact of solid bodies. *Comp Meth App Mech Eng* 198, 17 (2009), 1456–1472. <https://doi.org/10.1016/j.cma.2008.12.006>
- J. Zhang and A. Datta. 2006. Mathematical modeling of bread baking process. *Journal of Food Engineering* 75, 1 (July 2006), 78–89. <https://doi.org/10.1016/j.foodeng.2005.03.058>
- Q. Zhao and P. Papadopoulos. 2013. Modeling and simulation of liquid diffusion through a porous finitely elastic solid. *Computational Mechanics* 52, 3 (Sept. 2013), 553–562. <https://doi.org/10.1007/s00466-012-0831-6>
- Y. Zhao, L. Wang, F. Qiu, A. Kaufman, and K. Mueller. 2006. Melting and flowing in multiphase environment. *Computers & Graphics* 30, 4 (2006), 519–528.