IQ-MPM: An Interface Quadrature Material Point Method for Non-sticky Strongly Two-Way Coupled Nonlinear Solids and Fluids

YU FANG*, University of Pennsylvania
ZIYIN QU*, University of Pennsylvania
MINCHEN LI, University of Pennsylvania
XINXIN ZHANG, Tencent
YIXIN ZHU, University of California, Los Angeles
MRIDUL AANJANEYA, Rutgers University
CHENFANFU JIANG, University of Pennsylvania

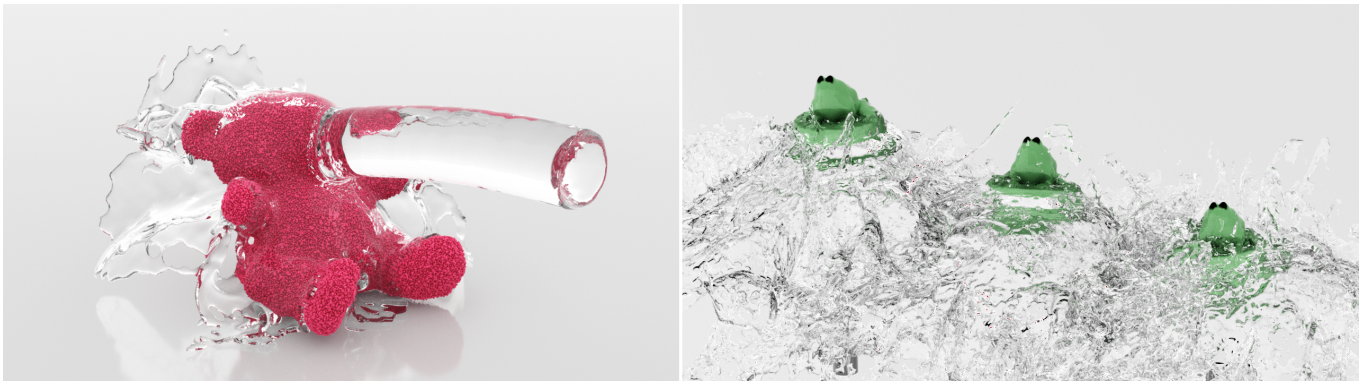


Fig. 1. (Left) A liquid jet hits a hyperelastic bear, pushing it down. (Right) Three hyperelastic frogs with different densities are lifted by liquid fountains, creating vibrant splashes. Our novel formulation properly enforces the free-slip boundary condition, while allowing for strong two-way coupled dynamics.

We propose a novel scheme for simulating two-way coupled interactions between nonlinear elastic solids and incompressible fluids. The key ingredient of this approach is a *ghost matrix* operator-splitting scheme for strongly coupled nonlinear elastica and incompressible fluids through the weak form of their governing equations. This leads to a stable and efficient method handling *large time steps* under the CFL limit while using *a single monolithic solve* for the coupled pressure fields, even in the case with highly nonlinear elastic solids. The use of the Material Point Method (MPM) is essential in the designing of the scheme, it not only preserves discretization consistency with the hybrid Lagrangian-Eulerian fluid solver, but also works naturally with our novel *interface quadrature* (IQ) discretization for free-slip boundary conditions. While traditional MPM suffers from sticky numerical artifacts, our framework naturally supports discontinuous tangential velocities at the solid-fluid interface. Our IQ discretization results in an easy-to-implement, fully particle-based treatment of the interfacial boundary, avoiding the additional complexities associated with intermediate level set or explicit mesh representations. The efficacy of the proposed scheme is verified by various challenging simulations with fluid-elastica interactions.

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

Additional Key Words and Phrases: Fluids, numerical methods, MPM, Fluid-structure interaction

ACM Reference Format:
Yu Fang, Ziyin Qu, Minchen Li, Xinxin Zhang, Yixin Zhu, Mridul Aanjaneya, and Chenfanfu Jiang. 2020. IQ-MPM: An Interface Quadrature Material Point Method for Non-sticky Strongly Two-Way Coupled Nonlinear Solids and Fluids. *ACM Trans. Graph.* 39, 4, Article 51 (July 2020), 16 pages. https://doi.org/10.1145/3386569.3392438

1 INTRODUCTION

Modern day applications thrive on the dynamic interactions between solids and fluids, such as air-filled rubber tires, hydraulics, a flying airplane, a floating ship, windmills, etc. Apart from the natural appeal of modeling rich physics-based interactions in virtual environments for more realistic visual effects, there is a growing need for fast methods for solid-fluid coupling due to emerging applications in virtual surgery [Lee et al. 2018; Mitchell et al. 2015], digital fabrication [Ma et al. 2017], and soft robotics [Hu et al. 2019].

The simplest approach to solid-fluid coupling is via the use of *partitioned* schemes, that iterate separately between the solid and

*equal contributions

Authors' addresses: Yu Fang, University of Pennsylvania; Ziyin Qu, University of Pennsylvania; Minchen Li, University of Pennsylvania; Xinxin Zhang, Tencent; Yixin Zhu, University of California, Los Angeles; Mridul Aanjaneya, Rutgers University; Chenfanfu Jiang, University of Pennsylvania.

0730-0301/2020/7-ART51 $15.00
https://doi.org/10.1145/3386569.3392438

Fig. 2. **Balls.** Our framework supports two-way coupling of incompressible fluids with linear and nonlinear elastic solids. Water-filled perforated spheres with different hyperelastic constitutive models are dropped on the ground; water gushes out of the spheres as they undergo large deformation and contact.

the fluid, and use one to define appropriate boundary conditions on the other [Arash et al. 2003; Guendelman et al. 2005]. While straightforward to implement, such an approach is known to suffer from stability issues, mandating the need for small time steps. Thus, researchers have explored the design of strongly-coupled or *monolithic* schemes, that exhibit better stability properties and allow for large time steps [Akbay et al. 2018; Batty et al. 2007; Chentanez et al. 2006; Robinson-Mosher et al. 2011, 2008; Teng et al. 2016; Zarifi and Batty 2017]. The design of monolithic systems is strongly influenced by the representations employed for the solid and the fluid. Solids, by definition, have a well-defined "rest" shape, and a Lagrangian mesh is commonly used for their evolution [Sifakis and Barbic 2012]. In contrast, fluids do not have a rest shape, and standard practice is to use an Eulerian grid [Bridson 2015]. This makes it challenging to enforce the constraint that the solid-fluid interface should move with the *same* normal velocity, which is either enforced only discretely at grid faces [Batty et al. 2007; Chentanez et al. 2006; Robinson-Mosher et al. 2011, 2008], or with expensive cut-cell formulations [Zarifi and Batty 2017]. To circumvent these issues, researchers have investigated the use of the same representation for both the solid and the fluid. Although impressive results were achieved, these methods may require expensive remeshing [Clausen et al. 2013] or particle resampling operations [Akinci et al. 2012], or suffer from increased numerical dissipation [Levin et al. 2011; Teng et al. 2016].

Another challenge arises due to nonlinear constitutive models for solids, which are conventionally solved using linear approximations (e.g., Newton's method), requiring outer iterations for converging to the exact solution for the nonlinear problem [Narain et al. 2016; Sifakis and Barbic 2012]. Prior work has exclusively focused on formulating monolithic systems for two-way coupling using instantaneous linear approximations, which necessitates outer iterations for solving the nonlinear system [Chentanez et al. 2006; Robinson-Mosher et al. 2011, 2008; Teng et al. 2016; Zarifi and Batty 2017]. However, this can be computationally very expensive.

In light of the above, the versatile Material Point Method (MPM) [Gao et al. 2018a,b; Hu et al. 2018; Jiang et al. 2015, 2016] presents an attractive alternative, that treats both solids and fluids in a unified fashion. Lagrangian particles are used to carry material information, and a background Eulerian grid is used for force computations. The use of particles avoids the numerical dissipation characteristic of grid-based schemes, while the use of a grid allows for regular finite-element stencils. MPM has been successfully used for simulating a wide variety of material behaviors [Daviet and Bertails-Descoubes 2016; Fang et al. 2019; Gao et al. 2018a; Guo et al. 2018; Nagasawa et al. 2019; Ram et al. 2015; Stomakhin et al. 2014; Wolper et al. 2019; Yue et al. 2015, 2018]. While MPM naturally supports automatic grid-based collision-handling and two-way coupling, the dynamic multi-material interactions are inherently *sticky*, as pointed out in [Stomakhin et al. 2014]. Traditional MPM [Jiang et al. 2016] does not support discontinuous tangential velocities at the multi-material interface, which leads to numerical stickiness. Visually, this can be quite disturbing, as shown by a water jet colliding with a deformable dragon [Fang et al. 2018]. To allow tangentially discontinuous velocities, researchers have considered using different grids per material, and enforce continuity of normal velocities as a post-process [Yan et al. 2018], but this is limited to explicit time integration and cannot capture incompressible fluids. Implicit integration with tangentially discontinuous velocities has been addressed in prior work using *cut-cell* methods [Boyd and Bridson 2012; Zarifi and Batty 2017] that carefully alter the topology of the simulation mesh and compute physical quantities accordingly to ensure that velocity and force profiles are continuous in the normal direction, but discontinuous tangentially. Clearly, such an approach requires delicate floating-point operations, is computationally expensive, and assumes that directions normal to the solid-fluid interface can be easily computed. Standard practice is to construct an intermediate level set representation for computing normal vectors, and manually move particles around if necessary to ensure consistency [Boyd and Bridson 2012]. Such an approach may be feasible for FLIP/PIC schemes, where particles are merely passive markers indicating the presence (or absence) of material, but can be disastrous for MPM (e.g., violate momentum conservation), where particles carry material information (such as mass, momentum, strain, etc.) and the particle distribution strongly influences the resulting forces.

1.1 Contributions

Motivated by these issues, our work makes several new contributions that significantly advance the state of the art for MPM simulations. We present a new *ghost matrix* operator-splitting scheme for monolithic coupling of nonlinear elastic solids with incompressible fluids. Specifically, we assume that an air-like, massless matrix skeleton material is applied to the solid MPM particles to enforce strong two-way coupling with the fluid through a mixed-FEM pressure field that is continuous at the solid-fluid interface by construction. Our monolithic system allows for stable solutions and large time steps under the CFL limit, while the operator-splitting approach

requires a *single* monolithic solve, even for highly nonlinear elastic solids. To the best of our knowledge, ours is the *first* method that satisfies all these properties. Unlike traditional MPM [Jiang et al. 2016], which suffers from sticky numerical artifacts, our framework naturally supports discontinuous tangential velocities at the solid-fluid interface.

For properly enforcing the free-slip boundary condition at the moving interface, we design a novel *interface quadrature* (IQ) cut-cell MPM formulation that is derived using the variational weak form principle. Our IQ discretization is easy to implement and leads to a fully particle-based treatment of the interfacial boundary conditions, without requiring additional level set or explicit mesh representations (as employed in FLIP schemes [Boyd and Bridson 2012]). We demonstrate the efficacy of our method on challenging examples with complex solid-fluid interactions.

2 RELATED WORK

While our focus is on MPM simulations [Jiang et al. 2016], we review all the different approaches proposed in prior work for two-way coupling rigid and deformable bodies with incompressible fluids.

Coupling Lagrangian Solids to Eulerian Fluids: Eulerian fluid simulation typically uses the marker-and-cell (MAC) grid discretization [Harlow and Welch 1965], first introduced to computer graphics by [Foster and Metaxas 1996]. Grid-aligned boundaries were readily handled in this work, but curved boundaries resulted in significant voxelization artifacts. Proper treatment of solid boundary conditions, particularly for advection, was subsequently addressed by [Foster and Fedkiw 2001; Rasmussen et al. 2004], although issues with the voxelized pressure solve remained unaddressed. Conforming volumetric meshes were used in [Feldman et al. 2005] for accurately treating irregular boundaries, and this work was also extended to dynamic coupling with rigid bodies [Klingner et al. 2006]. However, frequent remeshing operations were a major bottleneck.

An alternative "rigid-fluid" approach for two-way coupling was proposed in [Carlson et al. 2004], which momentarily treats the rigid body as a fluid, but this can cause the fluid to leak through the solid. A more accurate "leakproof" treatment was later proposed in [Guendelman et al. 2005] using one-sided interpolation during advection. The popular class of "cut-cell" methods, which formulate a modified Poisson system by clipping Eulerian grid voxels to conform to irregular object boundaries, was introduced in [Roble et al. 2005]. Batty et. al [2007] built upon this idea for two-way coupling fluids with rigid bodies, by casting the pressure solve as an energy minimization that accounts for the partial cell volumes. This approach was extended to two-way coupling with volumetric and thin shell deformable bodies in [Robinson-Mosher et al. 2011, 2008], where an explicit approach was used for the elastic forces, and the damping forces were solved together with the incompressible pressure in a monolithic, symmetric system. A Multigrid-based solver for two-way coupling rigid bodies with incompressible fluids was recently proposed by Aanjaneya [2018]. These techniques were preceded by the work of [Chentanez et al. 2006], who also formulated a monolithic system, albeit asymmetric. A symmetric positive definite (SPD) system that strongly couples both elastic and damping forces with the incompressible pressure was proposed in [Zarifi and Batty 2017]. In contrast to the above works, a stream function approach for two-way coupling voxelized rigid bodies with incompressible fluids was proposed in [Ando et al. 2015], which requires the solution of a vector Poisson system, but yields perfectly divergence-free velocities, even in the unsimulated air phase.

Coupling Lagrangian Solids to Lagrangian Fluids: Many of the challenges encountered by Eulerian grid-based fluids, such as volume preservation, support for implicit surface tension, or stable solid-fluid coupling, can be overcome by using a conforming Lagrangian mesh representation [Clausen et al. 2013; Misztal et al. 2012]. However, this entails the computational expense of frequent remeshing operations, and gases cannot be readily supported by such an approach. In contrast, particle-based methods [Akinci et al. 2012; Gissler et al. 2019; Macklin et al. 2014; Mazhar et al. 2015; Peer et al. 2018; Solenthaler et al. 2007] allow for a mesh-free implementation and avoid this overhead; however, their unstructured nature leads to expensive neighbor-lookups and poorly conditioned implicit systems, and foregoes the benefits of cache-locality and parallelism, as offered by grid-based methods [McAdams et al. 2010]. Inspired by solid-fluid coupling, there has also been some work on collision-handling between rigid and deformable bodies. Sifakis et al [2008] carefully formulated a Poisson-like system by meshing the air region between two colliding surfaces to apply globally-coupled impulses. A similar idea was later proposed in [Müller et al. 2015], although they used mesh optimization instead to enforce non-inversion of tetrahedra.

Coupling Eulerian Solids to Eulerian Fluids: An Eulerian discretization for elastic solids was first proposed in [Levin et al. 2011], which has the benefit of directly simulating volumetric data from CT scans, with full support for contact and collisions. This idea was later extended to thin strands [Sachdeva et al. 2015; Sueda et al. 2011], cloth and skin simulation [Li et al. 2013; Weidner et al. 2018], moving grids [Fan et al. 2013, 2014], as well as two-way coupling with incompressible fluids [Teng et al. 2016]. Eulerian discretizations have the benefit of employing the computational machinery of immersed boundary methods [Peskin 2002] for straightforward two-way coupling of different materials, as is evident from the impressive results achieved by the above works. However, they suffer from increased numerical dissipation, which can result in noticeable visual artifacts, such as volume loss. These issues can be mitigated (although not completely resolved) through the use of adaptivity [Aanjaneya et al. 2017; Ando et al. 2013; Chentanez and Muller 2011; Ferstl et al. 2014; Gao et al. 2017; Losasso et al. 2004; Setaluri et al. 2014], which places more resolution in regions of interest, and less resolution elsewhere.

Hybrid Particle-Grid Methods: Since the seminal work of [Zhu and Bridson 2005], extensive research has been done on hybrid particle-grid methods [Gao et al. 2009; Hong et al. 2008a; Lee et al. 2009; Losasso et al. 2008; Raveendran et al. 2011; Zhu et al. 2010]. Patkar et al. [2013] proposed a monolithic system for sub-grid scale bubbles, inspired by the earlier work of [Hong et al. 2008b]. More tight Voronoi diagram-based coupling approaches have also been proposed [Brochu et al. 2010; d. Goes et al. 2015; Sin et al. 2009]. Our work builds upon the Material Point Method (MPM) [Sulsky et al. 1995], first introduced to computer graphics by [Stomakhin et al.

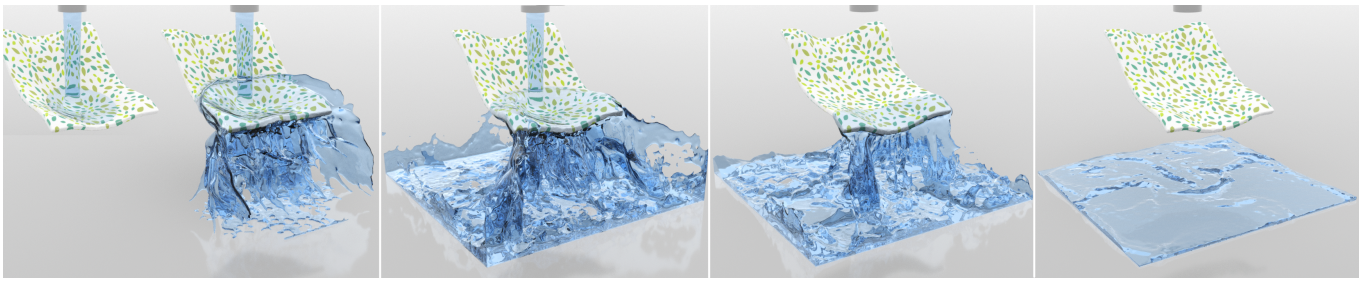


Fig. 3. **Flush rubber.** Pouring water on a rubber mat with fixed corners reveals intricate coupling behavior, without suffering from any stickiness issues.

2013]. MPM naturally supports automatic grid-based collision handling and two-way coupled interactions, albeit with a *no-slip* boundary condition at the colliding interface. Different grids have been used per material to enforce a free-slip boundary condition [Han et al. 2019; Yan et al. 2018], where the normal velocity is made continuous as a post-process, but this is limited to explicit time integration and requires small time steps for stability. Hu et al. [2018] proposed CPIC which modifies MPM transfers to enable velocity discontinuity using a single grid. Daviet et al. [2016] coupled MPM and rigid bodies with accurate frictional contact. MPM has been extended to model wet sand [Pradhana et al. 2017] and sediment mixtures [Gao et al. 2018a]. These latter works adopt a mixture theory approach, by applying a drag force to the discrete particles that is computed using the continuum phase. However, these formulations also correspond to a no-slip boundary condition, preventing the fluid from freely sliding against the solid. Fei et al. [2018] use an anisotropic drag force with separate velocity grids, and achieve more flexible cloth fluid interaction. Hybrid Lagrangian-Eulerian methods have also been used for collision handling in hair simulation [McAdams et al. 2009].

In particular, none of these prior works address strong two-way coupling of incompressible fluids with nonlinear elastic solids *without* requiring multiple monolithic solves. Our approach is inspired by the flux-splitting approach in [Stomakhin et al. 2014] for separately treating the deviatoric and dilational components of the stress tensor. However, we derive our method from first principles to support arbitrary constitutive models for nonlinear elastic solids, properly enforce the free-slip boundary condition, and also avoid the use of a MAC grid, which would require wider interpolation kernels and lead to increased numerical dissipation.

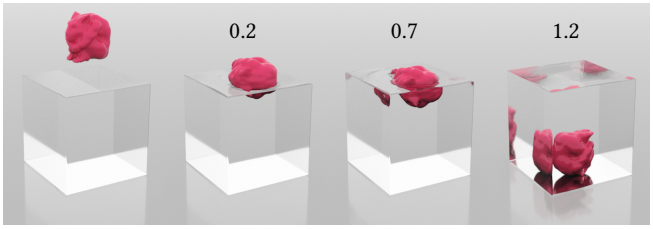


Fig. 4. **Buoyancy.** (From left to right) Soft creatures with increasing density ratios compared to water are dropped and achieve stable floating behavior.

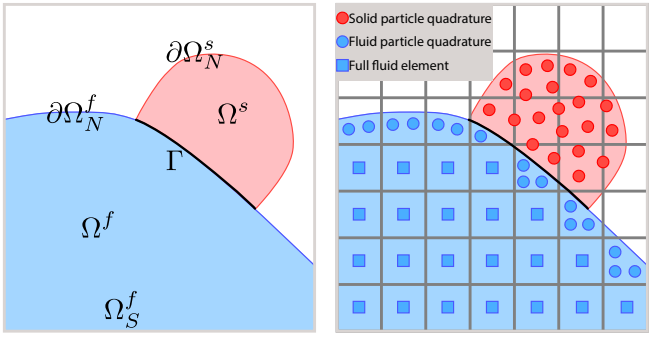


Fig. 5. (Left) A coupled solid-fluid system with a free-slip interface. (Right) MPM allows us to discretize the system using a combination of particle quadratures and analytic finite elements on a Cartesian background grid.

While coupling can be done with different approaches, we choose MPM for its ease of handling self-collision, topology change, plasticity, and multi-material interaction. Our examples are designed to demonstrate these benefits. By solving the stickiness issue in MPM's implicit solid-fluid coupling, we increase the types of phenomena that can be simulated with MPM.

3 GOVERNING EQUATIONS

Consider a coupled solid-fluid system (Fig. 5) where each domain is labeled with $k \in \{s, f\}$. For domain Ω^k, the single material governing equations [Bonet and Wood 2008] are

$$\frac{D\rho^k}{Dt} + \rho^k \nabla \cdot \mathbf{v}^k = 0, \qquad \mathbf{x} \in \Omega^k, \tag{1}$$

$$\rho^k \frac{D\mathbf{v}^k}{Dt} - \nabla \cdot \sigma^k - \rho^k \mathbf{g} = 0, \qquad \mathbf{x} \in \Omega^k \tag{2}$$

$$\sigma^k \cdot \mathbf{n}^k = \mathbf{b}, \qquad \mathbf{x} \in \partial\Omega^k_N \tag{3}$$

$$\mathbf{v}^k \cdot \mathbf{n}^k = v^k_S, \qquad \mathbf{x} \in \partial\Omega^k_S \tag{4}$$

$$\mathbf{v}^k = \mathbf{v}^k_{\mathrm{NS}}, \qquad \mathbf{x} \in \partial\Omega^k_{\mathrm{NS}} \tag{5}$$

where $\mathbf{g}$ is gravity, $\mathbf{n}^k$ is the outward pointing normal of Ω^k, ρ^k is the density, $\mathbf{v}^k$ is the velocity, σ^k is the Cauchy stress, $\mathbf{b}$, v^k_S and $\mathbf{v}^k_{NS}$ are prescribed free surface, slip, and no-slip boundary conditions. Note that Eq. (1) reduces to $\nabla \cdot \mathbf{v}^k = 0$ for incompressible materials. For inviscid fluids, we have $\sigma^k = p\mathbf{I}$, where p denotes the pressure.

For simulations with different materials, MPM [Stomakhin et al. 2013] and augmented MPM [Stomakhin et al. 2014] discretize these

equations on an entire continuum domain and, as such, momentum exchanges between different materials can be achieved automatically since particles exchange information through the same grid. Such convenience comes at a great cost: **the automatic MPM coupling is inherently restricted to sticky and no-slip interactions.** As a result, solid-solid interfaces experience infinite friction, while solid-fluid interfaces experience numerical stickiness. Without the ability to support free-slip boundary conditions, the practical versatility of MPM is greatly limited, producing animations with visual artifacts under certain setups.

To remedy this issue, we propose a method to allow for the free-slip boundary condition at the interface Γ between Ω^s and Ω^f by only enforcing the normal velocity continuity

$$(\mathbf{v}^s - \mathbf{v}^f) \cdot \mathbf{n}^s = 0, \qquad \mathbf{x} \in \Gamma \tag{6}$$

and pressure continuity

$$p^s - p^f = 0, \qquad \mathbf{x} \in \Gamma. \tag{7}$$

Since the normal vector can usually be estimated with greater accuracy in the solid domain, we choose the interface normal as $\mathbf{n}^s$. To switch back to a no-slip boundary condition, we can either require $\mathbf{v}^s - \mathbf{v}^f = \mathbf{0}$, or revert to single domain MPM.

4 METHOD

In this section we propose a ghost matrix strategy (§4.1) for continuum solid materials. By assuming the existence of an air-like massless matrix material, nonlinear elastic solids can be strongly coupled with fluids with a slip interface through a pressure-only linear system. Combined with a unified weak form MPM discretization (§4.3), the resulting monolithic system resembles an elliptic PDE discretization, thus benefiting from efficient linear solvers.

4.1 Ghost matrix for nonlinear solids

Let's first look at one single material and omit the superscript k in previous equations. For materials near the incompressible limit, directly solving for velocities in Eq. (2) will result in kinematic locking [Mast et al. 2012]. In finite elements, a velocity-pressure ($\mathbf{v} - p$) formulation is usually adopted [Bonet and Wood 2008], where p serves as the Lagrangian multiplier for the divergence-free velocity constraint. As demonstrated by Stomakhin *et al.* [2014], it is possible to reformulate Eqs. (1) and (2) into a pressure projection problem if there is a linear relationship between pressure and density $p = -\lambda(J-1)$ where $J = \det(\mathbf{F}) = \frac{\rho_0}{\rho}$ is the determinant of the deformation gradient $\mathbf{F}$ and λ is Lamé's first parameter. Correspondingly, one can show that the evolution equation of pressure is [Gonzalez and Stuart 2008]

$$\frac{Dp}{Dt} = -\lambda J \nabla \cdot \mathbf{v}. \tag{8}$$

During time integration from t^n to t^{n+1} and assuming an intermediate velocity field $\mathbf{v}^*$, Stomakhin *et al.* [2014] first integrate $\mathbf{v}^n$ to an intermediate velocity field by only applying the deviatoric stress, and then solve for the dilational stress to find the pressure, through which the velocity is advanced to $\mathbf{v}^{n+1}$ by applying the pressure gradient. Unfortunately, doing so strongly limits the choices of nonlinear solid constitutive models and imposes a high computational cost in the Jacobian and Hessian computations of the deviatoric elastic potential energy.

To support arbitrary nonlinear hyperelastic constitutive models, we adopt a different splitting strategy by viewing each infinitesimal material domain as a combination of a hyperelastic component and a mass-less ghost matrix material that acts similarly to compressible air (Fig. 7). In particular, we express the total energy density as

$$\Psi(\mathbf{F}^s, J^g) = \Psi^s(\mathbf{F}^s) + \Psi^g(J^g), \tag{9}$$

where $\Psi^s(\mathbf{F}^s)$ is a standard nonlinear solid constitutive model and

$$\Psi^g(J^g) = \frac{1}{2}\lambda^g (J^g - 1)^2 \tag{10}$$

describes the air-like response of the ghost matrix. To compensate for the extra volumetric response introduced by Ψ^g, we set Lamé's first parameter λ^s (in Ψ^s) and λ^g to be half of the traditional solid's Lamé's first parameter computed from the Young's modulus and the Poisson's ratio (Fig. 6). Here J^g and $\mathbf{F}^s$ are completely independent of each other and evolve using their own discretized evolution rules. Since the massless ghost matrix material is colocated with the solid, it shares the same velocity field with the solid domain. Note that we use the matrix as a purely conceptual mathematical tool for deriving our solid-fluid coupling scheme. The massless and colocation nature of the matrix enforces the equivalence of our solid-matrix mixture to a traditional solid in terms of mass conservation, momentum conservation, and strain-stress relationship.

4.2 Time-stepping the coupled equations

Taking into consideration the hybrid solid-matrix constitutive model $\Psi(\mathbf{F}^s, J^g)$ and the ghost matrix pressure evolution from Eq. (8), we can discretize the solid governing equations from time t^n to t^{n+1} as

$$\frac{\rho^{s,n}\mathbf{v}^{s,n+1}}{\Delta t} - \nabla \cdot \sigma^{s,n+1} + \nabla p^{g,n+1} = \frac{\rho^{s,n}\mathbf{v}^{s,n}}{\Delta t} + \mathbf{f}^{fs,n+1}, \tag{11}$$

$$\nabla \cdot \mathbf{v}^{s,n+1} + \frac{p^{g,n+1}}{\lambda^g J^{g,n} \Delta t} = \frac{p^{g,n}}{\lambda^g J^{g,n} \Delta t}, \tag{12}$$

where we have discretized the non-convective part of Dp^g/Dt as $(p^{n+1} - p^n)/\Delta t$. For notational simplicity, we assume an energy based solid-wall collision treatment (§6.2.4), and absorb the corresponding stress into $\sigma^{s,n+1}$. Eq. (12) adopts $\mathbf{v}^{s,n+1}$ rather than $\mathbf{v}^{g,n+1}$ because during discretization we enforce the ghost matrix continuum to be colocated with the solid continuum, thus $\mathbf{v}^s = \mathbf{v}^g$ throughout the simulation domain. We use $\mathbf{f}^{fs,n+1}$ to denote the unknown strong coupling force exerted from the fluid to the solid, and enforce it to be a normal pressure force through $\mathbf{f}^{fs,n+1} = h^{n+1}\mathbf{n}^{s,n}$, where h^{n+1}

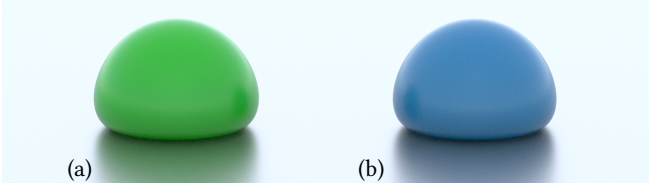


Fig. 6. **Volumetric compensation.** We demonstrate that compensating the volumetric response by modifying λ does not introduce any instability. (a) modified λ, (b) original λ.

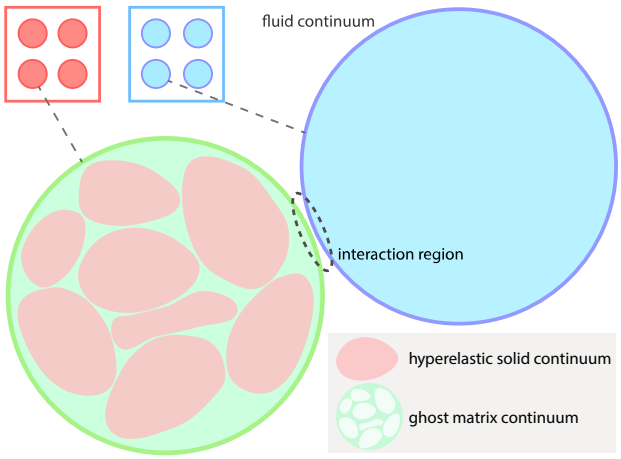


Fig. 7. **Ghost matrix.** We view the solid continuum as a combination of a hyperelastic solid component and an air-like massless ghost matrix continuum. Solid-fluid interaction is novelly reformulated as the pressure only interaction between the matrix and the fluid.

is the unknown pressure at the solid-fluid interface ($p^s = p^f = h$ at Γ is thus implicitly enforced), and $\mathbf{n}^{s,n}$ is the solid interface normal at time t^n. According to Newton's third law, we have $\mathbf{f}^{sf,n+1} = -h^{n+1}\mathbf{n}^{s,n}$ as the force exerted from the solid to the fluid.

Assuming no viscosity, the incompressible fluid domain can be described similarly, while taking the incompressibility assumption that $\lambda = \infty$:

$$\frac{\rho^f \mathbf{v}^{f,n+1}}{\Delta t} + \nabla p^{f,n+1} = \frac{\rho^f \mathbf{v}^{f,n}}{\Delta t} + \mathbf{f}^{sf,n+1} + \mathbf{f}^{w,n+1}, \tag{13}$$

$$\nabla \cdot \mathbf{v}^{f,n+1} = 0, \tag{14}$$

where $\mathbf{f}^{w,n+1} = -y^{n+1}\mathbf{n}^{f,n}$ captures the pressure from the slip boundary through y^{n+1}. Note that ρ^f does not change over time. We allow the fluid to touch slip boundary walls by requiring

$$\mathbf{v}^{f,n+1}\,|_{\partial\Omega_S^f} \cdot \mathbf{n}^f = v_S^f, \tag{15}$$

while $p\,|_{\partial\Omega_S^f} = y^{n+1}$ acts as the Lagrangian multiplier. The solid and the fluid are implicitly coupled through the interface pressure h^{n+1}, which, together with $\mathbf{v}^{s,n+1}$, $\sigma^{s,n+1}$, $p^{g,n+1}$, $\mathbf{v}^{f,n+1}$, $p^{f,n+1}$, and y^{n+1}, needs to be solved for while satisfying impenetrability

$$(\mathbf{v}^{s,n+1}\,|_\Gamma - \mathbf{v}^{f,n+1}\,|_\Gamma) \cdot \mathbf{n}^{s,n} = 0. \tag{16}$$

4.2.1 Splitting. The fully coupled constrained nonlinear system is extremely impractical to solve. We draw inspirations from Teng *et al.* [2016]'s treatment of linear systems to cast the fully coupled system into easier substeps with operator splitting. We first advance the solid velocity with a fully nonlinear Newton solve ($\mathbf{v}^{s,n} \to \mathbf{v}^{s,*}$) ignoring the matrix and the fluid:

$$\frac{\rho^{s,n}\mathbf{v}^{s,*}}{\Delta t} - \nabla \cdot \sigma^{s,*} = \frac{\rho^{s,n}\mathbf{v}^{s,n}}{\Delta t}, \tag{17}$$

then $\mathbf{v}^{s,*}$ replaces $\mathbf{v}^{s,n}$ in the fully linear coupled equations

$$\frac{\rho^{s,n}\mathbf{v}^{s,n+1}}{\Delta t} + \nabla p^{g,n+1} - h^{n+1}\mathbf{n}^{s,n} = \frac{\rho^{s,n}\mathbf{v}^{s,*}}{\Delta t}. \tag{18}$$

As shown in §5 a Galerkin weak form discretization of this $\mathbf{v} - p - y - h$ system leads to a reduced pressure-only discrete linear system on $p^{g,n+1}$, $p^{f,n+1}$, y^{n+1} and h^{n+1}. We solve for the pressures and substitute them back into Eq. (18) (by approximately delaying their non-orthogonal influence on σ^s) and Eq. (13) to compute the new velocities $\mathbf{v}^{s,n+1}$ and $\mathbf{v}^{f,n+1}$.

4.3 Weak form

4.3.1 Nonlinear elasticity step. We advance solid velocity from $\mathbf{v}^{s,n}$ to $\mathbf{v}^{s,*}$ through the nonlinear solid constitutive model $\Psi^s(\mathbf{F}^s)$ where $\sigma^{s,*} = \frac{1}{\det(\mathbf{F}^{s,*})}\frac{\partial \Psi^s}{\partial \mathbf{F}}\mathbf{F}^{s,*T}$ in the momentum equation (Eq. (17)). We follow the MLS-MPM weak form discretization by Hu *et al.* [2018] and solve the spatial-temporal discretization using the projected Newton-Ralphson solver [Gast et al. 2015; Wang et al. 2020], where the nonlinear hyperelastic stress derivative tensors are projected to be SPD so that preconditioned Conjugate Gradient can be used as the inner linear solver together with the backtracking line search [Nocedal and Wright 2006]. Since this step is standard for implicit MPM, we skip the derivation and refer to Jiang *et al.* [2016] for more details on the weak form discretization and Teran *et al.* [2005] for more details on the positive definiteness projection.

4.3.2 Weak form of the strong coupling system. We solve Eqs. (12) to (16) and (18)) using a Galerkin weak form approach. In turn, we multiply test functions $\mathbf{q}^s(\mathbf{x})$, $r^s(\mathbf{x})$, $\mathbf{q}^f(\mathbf{x})$, $r^f(\mathbf{x})$, $u^f(\mathbf{x})$, and $w(\mathbf{x})$ onto the equations, and integrate over their corresponding domains to get

$$\int_{\Omega^s} \frac{\rho^{s,n}\mathbf{q}^s \cdot \mathbf{v}^{s,n+1}}{\Delta t} d\mathbf{x} + \int_{\Omega^s} \mathbf{q}^s \cdot \nabla p^{g,n+1} d\mathbf{x} - \int_\Gamma \mathbf{q}^s \cdot \mathbf{n}^{s,n} h^{n+1} ds = \int_{\Omega^s} \frac{\rho^{s,n}\mathbf{q}^s \cdot \mathbf{v}^{s,*}}{\Delta t} d\mathbf{x}, \tag{19}$$

$$\int_{\Omega^s} r^s \nabla \cdot \mathbf{v}^{s,n+1} d\mathbf{x} + \int_{\Omega^s} \frac{p^{g,n+1}}{\lambda^g J^{g,n} \Delta t} r^s d\mathbf{x} = \int_{\Omega^s} \frac{p^{g,n}}{\lambda^g J^{g,n} \Delta t} r^s d\mathbf{x}, \tag{20}$$

$$\int_{\Omega^f} \frac{\rho^f \mathbf{q}^f \cdot \mathbf{v}^{f,n+1}}{\Delta t} d\mathbf{x} + \int_{\Omega^f} \mathbf{q}^f \cdot \nabla p^{f,n+1} d\mathbf{x} + \int_{\partial\Omega_s^f} \mathbf{q}^f \cdot \mathbf{n}^f y^{n+1} ds + \int_\Gamma \mathbf{q}^f \cdot \mathbf{n}^{s,n} h^{n+1} ds = \int_{\Omega^f} \frac{\rho^f \mathbf{q}^f \cdot \mathbf{v}^{f,n}}{\Delta t} d\mathbf{x}, \tag{21}$$

$$\int_{\Omega^f} r^f \nabla \cdot \mathbf{v}^{f,n+1} d\mathbf{x} = 0, \tag{22}$$

$$\int_{\partial\Omega_s^f} \mathbf{v}^{f,n+1} \cdot \mathbf{n}^f u ds = \int_{\partial\Omega_s^f} v_s^f u ds, \tag{23}$$

$$\int_\Gamma (\mathbf{v}^{s,n+1} - \mathbf{v}^{f,n+1}) \cdot \mathbf{n}^{s,n} w ds = 0, \tag{24}$$

which describes a continuous and symmetric system for unknown fields $\mathbf{v}^{s,n+1}, p^{g,n+1}, \mathbf{v}^{f,n+1}, p^{f,n+1}, y^{n+1}, h^{n+1}$.

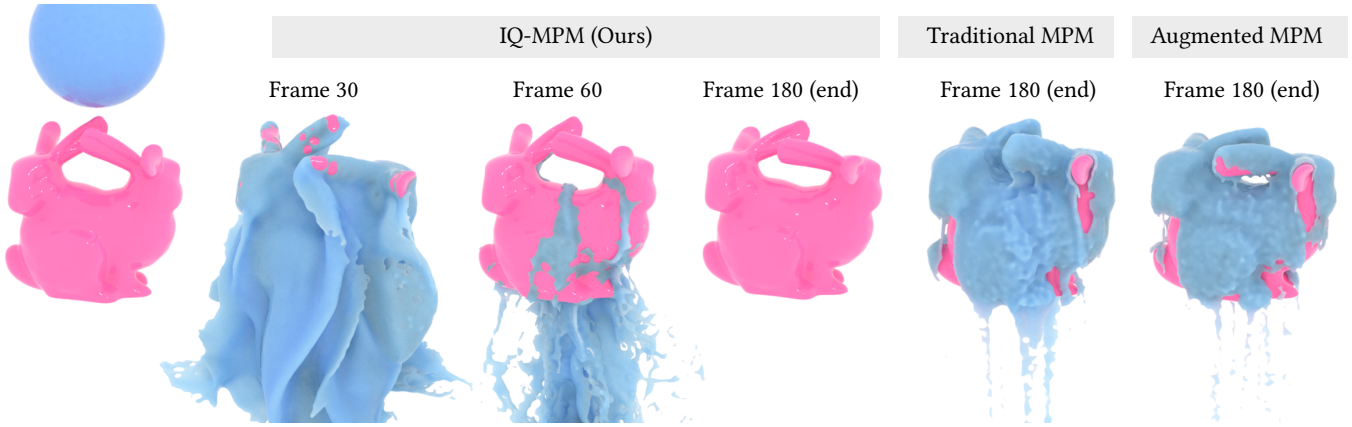


Fig. 8. **Bunny.** Our method achieves stable, monolithic two-way coupling between MPM-based nonlinear compressible elasticity and incompressible free surface fluids. Guaranteeing a free-slip interface with discontinuous tangential velocities, our method does not suffer from the notorious numerical stickiness artifacts in multi-material interactions, unlike traditional MPM [Jiang et al. 2016; Stomakhin et al. 2013] and augmented MPM [Stomakhin et al. 2014].

5 THE INTERFACE QUADRATURE METHOD

There are many possible interpolation kernels for the continuous quantities and test functions in the weak form. In traditional $\mathbf{v} - p$ mixed Finite Elements [Hughes 2012], it is well known that for incompressible materials, velocities usually should be discretized with higher-order polynomials than the pressure to satisfy inf-sup stability and to prevent kinematic locking. Traditional MAC-grid based fluids, which are popular in computer graphics [Bridson 2015], correspond to a piecewise constant interpolation of pressure at cell centers and piecewise linear interpolation at cell faces – leading to gradient and divergence operators that are equivalent to central finite-difference. Inappropriate choices of the velocity-pressure pair, such as adopting fully colocated linear interpolations for both, suffer from spurious modes like odd-even decoupling.

In the context of MPM, we have more limited choices. For MPM solids, the velocities require at least C^1 continuous functions to prevent the cell-crossing instability caused by discontinuous forces [Steffen et al. 2008]. Since mass lumping is typically needed (to prevent a singular mass matrix [Love and Sulsky 2006]), the particle-grid interpolation function in MPM also needs to always be nonnegative. Resultingly, MPM typically uses quadratic or cubic B-splines for discretizing the velocity field on the grid, and many advanced particle-grid information transfer schemes also assume such choices [Fu et al. 2017; Hu et al. 2018; Jiang et al. 2015].

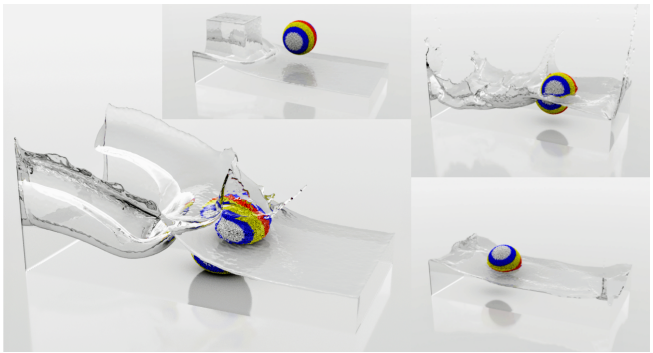


Fig. 9. **Dam jello ball.** A hyperelastic ball with density $500kg/m^3$ experiences dam break and eventually stably stays calm on the water surface with half of its volume immersed.

Without loss of generality, we put solid velocity degrees of freedom on the nodes of the solid grid. In the meantime, we assume the fluid grid has the same origin and resolution as the solid grid. We use $N_\mu^{s/f,\gamma}(\mathbf{x})$ to denote B-spline interpolation kernels of degree γ on the solid (s) or the fluid (f) grid, where $\mu \in i, j$ corresponds to a nodal kernel, and $\mu \in b, c$ corresponds to a cell-centered kernel.

We can then spatially discretize all continuous functions $\mathbf{v}(\mathbf{x})$, $\mathbf{q}(\mathbf{x})$, $r(\mathbf{x})$, $p(\mathbf{x})$, $y(\mathbf{x})$, $h(\mathbf{x})$, $u(\mathbf{x})$, and $w(\mathbf{x})$ through particular combinations of B-spline kernels. We limit the B-spline degree to be less than or equal to 2, and explore the space of stable choices. To prevent cell-crossing noise, we always expand solid velocities using quadratic B-splines $\mathbf{v}^s(\mathbf{x}) = \sum_i \mathbf{v}_i^s N_i^{s,2}(\mathbf{x})$. To satisfy the inf-sup condition in mixed FEM [Hughes 2012], we can discretize the pressure-like quantities using either linear or constant kernels.

The first choice is to pick linear kernels. A resulting scheme that unifies the discretization of solid and fluid is then to use N^2 for velocities and N^1 for matrix, fluid, boundary, and interface pressures. Following standard notations in mixed finite elements [Elman et al. 2014], we label this approach "B2B1-B1-B2B1" since both the solid-ghost system and the fluid system are discretized using quadratic B-spline velocities at nodes and linear pressures at cell centers, while the coupling pressure is also linear (thus the "B1" in between), see Fig. 12 (a). Such a "node-center" layout is usually referred to as semi-staggered and proven useful for fluid dynamics [Gagniere et al. 2020; Zhang et al. 2017]). Note that the closely related Q2Q1 element with Lagrangian polynomials (also known as the Taylor-Hood element) was first proposed by Taylor and Hood [1974] for solving the Navier-Stokes equations on quadrilateral elements. As pointed out by Rüberg and Cirak [2012], a *colocated* B2B1 B-spline

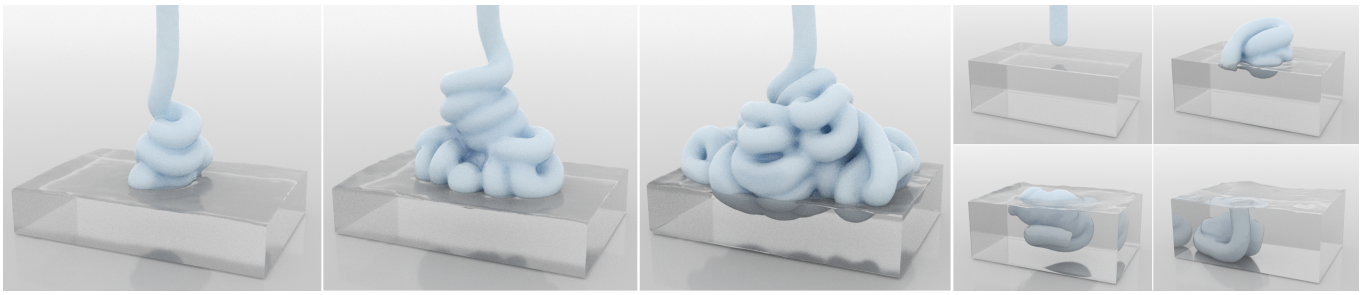


Fig. 10. **Shaving cream.** (left) Elastoplastic shaving cream (simulated using Saint Venant-Kirchhoff elasticity paired with von Mises plasticity following Gao *et al.* [2017]) is squeezed onto a tank of water, undergoing topology change and two-way coupling at the same time. (right) Comparing shaving creams with different densities, ranging from small to large in top right, bottom left, and bottom right.

kernel choice for Stokes leads to checker-board modes. The *semi-staggered* layout we use does not cause visually noticeable issues. A deeper exploration of its inf-sup conditions could be a promising future work.

Even though B2B1-B1-B2B1 provides a natural discretization, it also leads to a wide gradient operator and a dense pressure stencil for a corresponding discrete Laplacian operator. For efficiency and storage considerations, we prefer to discretize the fluid with a piecewise linear (B1) velocities paired with piecewise constant (B0) pressures, and couple with the solid through piecewise constant (B0) interface pressures. Accordingly, the ghost matrix pressure is also discretized with piecewise constant (B0) functions. We label the resulting new scheme as "B2B0-B0-B1B0" (Fig. 12 (b)).

5.1 B2B0-B0-B1B0 scheme

Adopting quadratic velocities for the solid, we expand $\mathbf{v}^s$ and $\mathbf{q}^s$ using $N_i^{s,2}(\mathbf{x})$:

$$\mathbf{v}^{s,\{n,n+1\}}(\mathbf{x}) = \sum_i \mathbf{v}_i^{s,\{n,n+1\}} N_i^{s,2}(\mathbf{x}), \tag{25}$$

$$\mathbf{q}^s(\mathbf{x}) = \sum_j \mathbf{q}_j^s N_j^{s,2}(\mathbf{x}). \tag{26}$$

The fluid velocity, $\mathbf{v}^f$, and the momentum test function, $\mathbf{q}^f$, instead get discretized using linear kernels defined at cell centers $N_b^{f,1}(\mathbf{x})$:

$$\mathbf{v}^{f,\{n,n+1\}}(\mathbf{x}) = \sum_b \mathbf{v}_b^{f,\{n,n+1\}} N_b^{f,1}(\mathbf{x}), \tag{27}$$

$$\mathbf{q}^f(\mathbf{x}) = \sum_c \mathbf{q}_c^f N_c^{f,1}(\mathbf{x}). \tag{28}$$

All other scalar pressure-like quantities (p, r, y, u, h, w) are discretized at nodes using $N_i^{s/f,0}(\mathbf{x})$ as piecewise constant fields. As extensively investigated in the CFD community, the B1B0 fluid element contains weakly singular modes that can be stabilized through regularization or damping [Christon 2002; Zhang et al. 2017]. In practice we do not visually observe spurious artificial velocities in any of our examples.

By letting test functions to in turn take nodal deltas, the weak form equations reduce to a symmetric linear system

$$\begin{pmatrix} \frac{1}{\Delta t}M^s & G^s & 0 & 0 & 0 & -H^{s\,T} \\ G^{s\,T} & -\frac{1}{\Delta t}S^s & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{\Delta t}M^f & G^f & B^T & H^{f\,T} \\ 0 & 0 & G^{f\,T} & 0 & 0 & 0 \\ 0 & 0 & B & 0 & 0 & 0 \\ -H^s & 0 & H^f & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} v^{s,n+1} \\ p^{g,n+1} \\ v^{f,n+1} \\ p^{f,n+1} \\ y^{n+1} \\ h^{n+1} \end{pmatrix} = r \tag{29}$$

with $r = \left(\frac{1}{\Delta t}M^s v^{s,n}, -\frac{1}{\Delta t}S^s p^{g,n}, \frac{1}{\Delta t}M^f v^{f,n}, 0, b, 0\right)^T$, where we have concatenated velocity and pressure unknowns into long column vectors. Here M^s is the lumped diagonal solid mass matrix with

$$M^s_{i_\alpha i_\alpha} = \int_{\Omega^s} \rho^{s,n} N_i^{s,2}(\mathbf{x}) d\mathbf{x}, \tag{30}$$

where $\alpha = 0, 1, 2$ denotes a coordinate component. G^s is the solid domain gradient operator (after applying divergence theorem on $\int_{\Omega^s}$) with

$$G^s_{i_\alpha j} = -\int_{\Omega^s} N_j^{s,0}(\mathbf{x})(\nabla N_i^{s,2}(\mathbf{x}))_\alpha d\mathbf{x}. \tag{31}$$

The scaling matrix S^s for ghost air pressure is given by

$$S^s_{ij} = \int_{\Omega^s} \frac{1}{\lambda^g J^{g,n}} N_i^{s,0}(\mathbf{x}) N_j^{s,0}(\mathbf{x}) d\mathbf{x}, \tag{32}$$

which we can lump into a diagonal matrix similarly to the mass:

$$S^s_{ii} = \int_{\Omega^s} \frac{1}{\lambda^g J^{g,n}} N_i^{s,0}(\mathbf{x}) d\mathbf{x}. \tag{33}$$

The fluid lumped mass, gradient, and boundary operators are given by

$$M^f_{c_\alpha c_\alpha} = \rho^f \int_{\Omega^f} N_c^{f,1}(\mathbf{x}) d\mathbf{x}, \tag{34}$$

$$G^f_{c_\alpha i} = -\int_{\Omega^f} N_i^{s,0}(\mathbf{x})(\nabla N_c^{f,1}(\mathbf{x}))_\alpha d\mathbf{x}, \tag{35}$$

$$B_{c_\alpha i} = \int_{\partial\Omega_s^f} N_c^{f,1}(\mathbf{x}) N_i^{f,0}(\mathbf{x}) n_\alpha^f ds \tag{36}$$

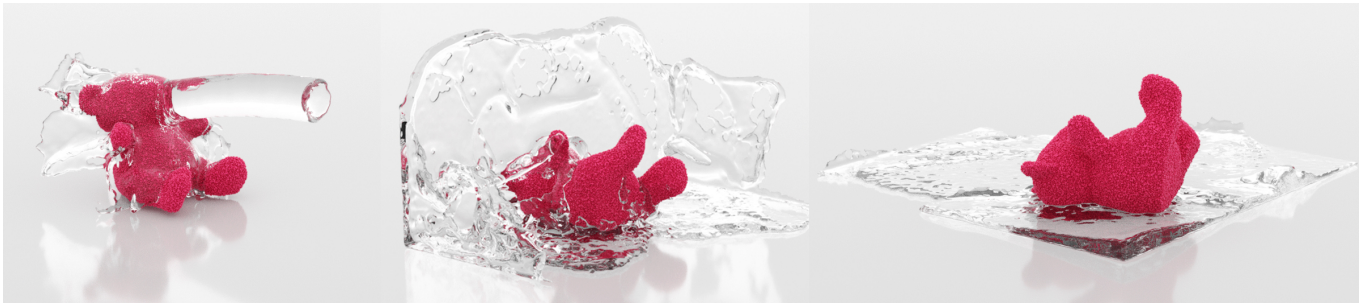
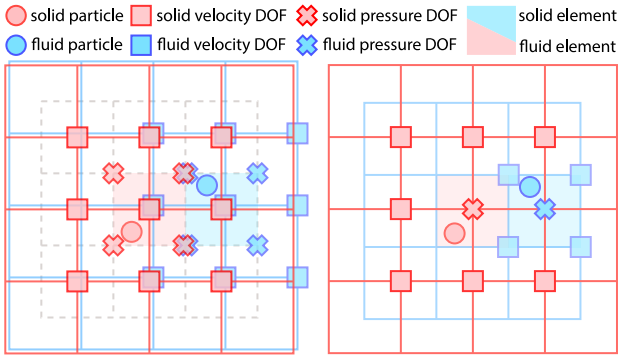


Fig. 11. **Bear bath.** Unlike the "dragon bath" example in Fang *et al.* [2018] where water numerically sticks to the dragon in a visually incorrect manner (an expected result from standard MPM), the bear couples with water in a fully waterproof way, allowing water to slide naturally along its body.



(a) B2B1-B1-B2B1 (b) B2B0-B0-B1B0

Fig. 12. MPM mandates a quadratic or higher-order velocity kernel for solids. Stable possibilities for discretizing the coupled system include (a) quadratic velocities with linear pressures for both domains and the interface, (b) (our choice) quadratic velocities for solids, linear velocities for fluids, and constant (discontinuous) pressures for both domains and the interface.

where ρ^f is a constant (unlike in the solid domain). Finally, the coupling terms are

$$H^s_{ji_\alpha} = \int_\Gamma N^{s,2}_i(\mathbf{x}) N^{s,0}_j(\mathbf{x}) n^{s,n}_\alpha ds, \tag{37}$$

$$H^f_{jc_\alpha} = \int_\Gamma N^{f,1}_c(\mathbf{x}) N^{f,0}_j(\mathbf{x}) n^{s,n}_\alpha ds, \tag{38}$$

and constructed using different velocity kernels, but both using a piecewise constant interface pressure.

5.2 Particle quadrature for the solid and matrix

As commonly done in MPM [Jiang et al. 2016], we discretize all integrals that are only associated with solid and matrix using particle quadratures, since particles naturally encode mass, volume, constitutive model parameters, and strain. Specifically, we have

$$M^s_{i_\alpha i_\alpha} = m^s_i = \sum_p m^s_p N^{s,2}_i(\mathbf{x}_p), \tag{39}$$

$$G^s_{i_\alpha j} = -\sum_p V^n_p N^{s,0}_j(\mathbf{x}_p)(\nabla N^{s,2}_i(\mathbf{x}_p))_\alpha, \tag{40}$$

$$S^s_{ii} = s^s_i = \sum_p \frac{1}{\lambda^g_p} V^0_p N^{s,0}_i(\mathbf{x}_p). \tag{41}$$

where m_p is the mass of solid particle p, V^n_p is its current volume (relating to its original volume V^0_p through $V^n_p = V^0_p \det(\mathbf{F}^n_p)$). We compute the previous time pressure $p^{g,n}$ in the right hand side taking volume weighted average over all ghost matrix particles

$$p^{g,n}_i = \left(\sum_p V^0_p J^{g,n}_p p^{g,n}_p N^{s,0}_i(\mathbf{x}_p)\right) / \left(\sum_p V^0_p J^{g,n}_p N^{s,0}_i(\mathbf{x}_p)\right), \tag{42}$$

where $p^{g,n}_p = -\lambda^g_p (J^{g,n}_p - 1)$.

Incompressible solid. An additional useful feature of our B2B0 solid formulation is that as we let $\lambda^g = \infty$, the only imposed change on the discretization is to let $S^s_{ii} = 0$. Continuously this can also be derived if we replace Eq. (12) with the divergence-free constraint $\nabla \cdot \mathbf{v}^{s,n+1} = 0$. In other words, our B2B0 solid formulation itself can also serve as a standalone mixed Finite Element scheme for incompressible solids.

5.3 Subgrid treatment for free surface and interface

For the fluid domain, one standard choice is to analytically integrate the interpolation functions to compute M^f, G^f and B. This results in grid aligned staircase artifacts near the boundaries, as extensively studied by Gibou *et al.* [2002] and Batty *et al.* [2007] on MAC-grid based fluids. Instead of following standard approaches by computing level set based cut-cells, we use particle quadratures for fluid elements near the free surface and the solid-fluid interface to capture subgrid information.

5.4 Interface quadrature

The major challenge lies in the discretization of H, which involves integration over the solid-fluid interface Γ. A superficially straightforward choice is to construct two level sets wrapping the two domains and reconstruct a sharp interface through averaging. Unfortunately, even though this works fine for two-phase fluids [Boyd and Bridson 2012], it is easy to see that this reconstructed level set may only partially overlap one of the two domains in the solid-fluid coupling case. For example, this interfacial codimensional surface could be influencing grid nodes on the solid grid that does not even have a solid degree of freedom (i.e., mass). Furthermore, due to the offset between solid and fluid particles, such an interface would not influence the same set of pressure degrees of freedom on the two domains, breaking pressure continuity at the interface.

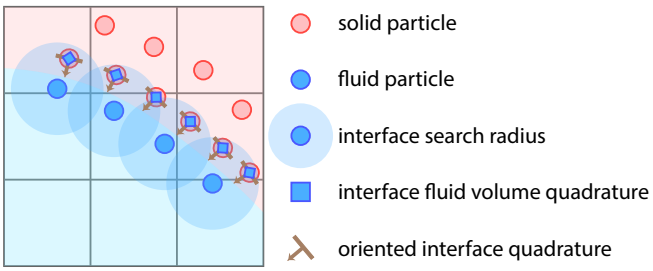


Fig. 13. **IQ allocation.** We sample interface quadratures with areas and normals at solid particles that are $0.5\Delta x$ close to fluid particles. At the same locations, we also sample additional fluid volume quadrature points so that the fluid domain is closely touching the solid domain at the interface with no missing degrees of freedom on the grid.

Correspondingly, we emphasize that it is important that the two domains see not only exactly the same interface but also that each has a complete set of mass-full degrees of freedom on the grid so that the interface is fully covered by the union of the interpolation kernels of the degrees of freedom.

To achieve this goal, we discretize the boundary integral of Γ through *interface quadrature* (IQ) points. To ensure the coverage in the solid domain, we sample IQ points directly at solid particle locations. At the beginning of each time step, we locate solid particles that are within $0.5\Delta x$ to fluid particles. For each such solid particle, we insert an IQ point q with area A_q and normal $\mathbf{n}_q$ (Fig. 13).

To ensure coverage in the fluid domain, we extend the fluid quadrature points towards the IQ locations. Continuously, this is equivalent to treating the fluid continuum as closely touching the solid domain when their material particles are sufficiently close. For each identified IQ point, we sample an additional fluid volume quadrature at this location, inheriting the average volume of the fluid particles that are $0.5\Delta x$ close-by. Since we already adopt particle quadratures for the fluid volume integrals near the interface, these additional volume quadratures can be easily accounted for. By sampling them at IQ locations, we guarantee that the fluid degrees of freedom on the grid will have kernel support at IQ points.

With IQ, we discretize the coupling term as

$$H^s_{ji_\alpha} = \sum_q A_q N_i^{s,2}(\mathbf{x}_q) N_j^{s,0}(\mathbf{x}_q) n^n_{q\alpha} ds, \tag{43}$$

$$H^f_{jc_\alpha} = \sum_q A_q N_c^{f,1}(\mathbf{x}_q) N_j^{f,0}(\mathbf{x}) n^n_{q\alpha} ds, \tag{44}$$

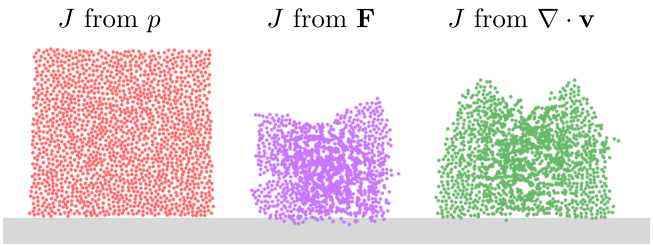


Fig. 14. **ghost matrix strain update.** Updating the ghost matrix strain J from the piecewise constant pressure interpolation results in a stable B2B0 solid simulation. Other approaches are not consistent with the weak form discretization and results in instabilities.

where we approximate IQ area, A_q, as the cross section area of the colocating solid particle assuming it is originally shaped as a sphere and deformed with $\mathbf{F}$. IQ normal, $\mathbf{n}_q$, is evaluated using the mass gradient of the solid domain (see §6.2.3).

5.5 Pressure-only system

The coupled system Eq. (29) corresponds to a saddle point KKT system and has extremely bad conditioning. Fortunately, since we use lumped mass matrices, we can eliminate the velocity degrees of freedom and solve for the pressure variables first:

$$\begin{pmatrix} A_{11} & 0 & 0 & A_{14} \\ 0 & A_{22} & A_{23} & A_{24} \\ 0 & A_{23}^T & A_{33} & A_{34} \\ A_{14}^T & A_{24}^T & A_{34}^T & A_{44} \end{pmatrix} \begin{pmatrix} p^{g,n+1} \\ p^{f,n+1} \\ y^{n+1} \\ h^{n+1} \end{pmatrix} = \begin{pmatrix} \frac{S^s p^{g,n}}{\Delta t} - G^{s\,T} v^{s,n} \\ G^{f\,T} v^{f,n} \\ B v^{f,n} - b \\ H^s v^{s,n} - H^f v^{f,n} \end{pmatrix} \tag{45}$$

where $A_{11} = \frac{S^s}{\Delta t} + \Delta t G^{s\,T} M^{s\,-1} G^s$, $A_{14} = -\Delta t G^{s\,T} M^{s\,-1} H^{s\,T}$, $A_{22} = \Delta t G^{f\,T} M^{f\,-1} G^f$, $A_{23} = \Delta t G^{f\,T} M^{f\,-1} B^T$, $A_{24} = \Delta t G^{f\,T} M^{f\,-1} H^{f\,T}$, $A_{33} = \Delta t B M^{f\,-1} B^T$, $A_{34} = \Delta t B M^{f\,-1} H^{f\,T}$, $A_{44} = \Delta t (H^s M^{s\,-1} H^{s\,T} + H^f M^{f\,-1} H^{f\,T})$. The system is symmetric positive semi-definite up to constant pressure-mode null space when the fluid has no free surface (which can be handled by projecting it out in the CG solver, see §6.2.1). After we solve for these pressure variables, we can substitute them back into the momentum equation to get the new velocities.

5.6 B2B0 strain update for the ghost matrix

After we get the new velocities, the particles can directly interpolate the velocity field and advect using APIC [Jiang et al. 2015, 2017]. The solid (together with ghost matrix) particles need to additionally update their strain. Since we use a quadratic B-spline kernel for the solid velocity $\mathbf{v}^{s,n+1}$, we can update the solid deformation gradient using $\mathbf{F}_p^{n+1} = (\mathbf{I} + \Delta t(\nabla \mathbf{v}^{n+1}(\mathbf{x}_p))\mathbf{F}_p^n$ as in traditional MPM.

The ghost matrix particles additionally carry strain J^g. Natural choices for updating this include (1) setting it to the determinant of the updated solid deformation gradient ($J^g = \det(\mathbf{F}^s)$), and (2) evolving it using the velocity divergence ($J_p^{g,n+1} = (1 + \Delta t(\nabla \cdot \mathbf{v}^{n+1})(\mathbf{x}_p))J_p^{g,n}$) following the MPM treatment of compressible fluids [Pradhana et al. 2017]. However, a very simple experiment with dropping an elastic cube easily reveals the instability of these two options, see Fig. 14. Through experiments, we also found that we could not observe these instabilities if we adopt a *B2B1* solid discretization. The problem is that with *B2B0*, the ghost matrix pressure is required to be discontinuous, and updating it with the high order velocity field causes inconsistency with the discretization choice. Therefore, we instead interpolate the updated pressure field $p^{g,n+1}$ using $N^{s,0}(\mathbf{x})$ to transfer back to the ghost matrix particles, and recompute their strain through $p = -\lambda(J-1)$. Being consistent with the weak form discretization successfully removes the instability and results in stable and natural behaviors.

6 ALGORITHM

6.1 Overview

IQ-MPM reuses a considerable amount of standard MPM procedures. We summarize the pipeline below and in Fig. 15.

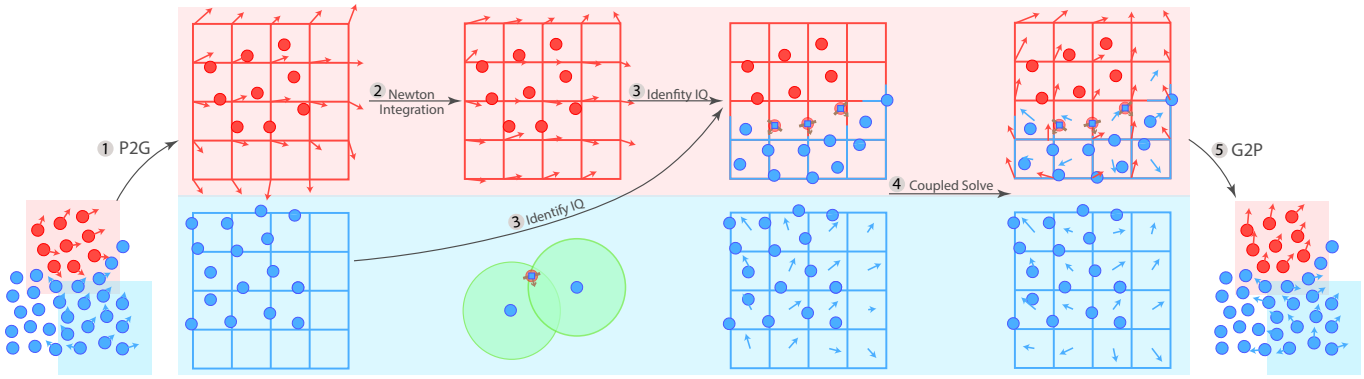


Fig. 15. **Algorithm overview.** Our method can be performed efficiently by launching standard MPM procedures, see §6.1 for more detailed explanation of every algorithmic step.

(1) **Particle-to-grid transfer.** Solid and fluid particles transfer their time, n, mass, and velocity, $\mathbf{v}^n$, onto the grids using APIC [Jiang et al. 2015]; ghost matrix particles transfer their pressures, $p^{g,n}$, onto the grid.
(2) **Projected Newton solver.** Solid velocities are integrated from $\mathbf{v}^{s,n}$ to $\mathbf{v}^{*,n}$ (see §4.3.1) considering nonlinear hyperelasticity and collision objects.
(3) **Identify IQ.** Following §5.4, interface quadrature points with areas and normals are identified, as well as additional fluid volume quadrature points. The fluid velocity field, $\mathbf{v}^n$, is constructed with contributions from the additional quadrature points.
(4) **Coupled solve.** The coupled pressure-only system Eq. (45) is constructed and solved. New velocities are reconstructed using the pressures.
(5) **Grid-to-particle transfer.** Solid and fluid particles update their velocities and advect using APIC; ghost matrix particles update their strain J^g using the interpolated $p^{g,n+1}$.

Overall, all steps other than step (3) can be performed by launching standard MPM particle-grid transfer kernels. Due to the usage of particle quadratures, the construction of the system matrix can also be finished with one particle-to-grid transfer operation through all particles and quadrature points. We use SPGrid [Setaluri et al. 2014] as our underlying sparse grid data structure, which enables efficient cache-friendly neighborhood search during IQ identification.

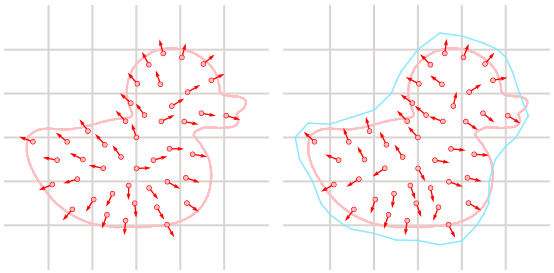


Fig. 16. **Normal estimation.** (left) We estimate the normal using B-spline interpolated mass gradient §6.2.3. (right) A more expensive option is to estimate the normal using the gradient of a level set constructed with particles as in Boyd and Bridson [2012].

6.2 More algorithmic details

6.2.1 Multigrid PCG. We solve Eq. (45) with an efficient algebraic multigrid-preconditioned Conjugate Gradient solver AMGCL [Demidov 2019]. We adopt Chebyshev relaxation and smoothed aggregation coarsening paired with a w-cycle iteration.

6.2.2 Improving the condition. In level set based cut-cell methods [Batty et al. 2007], to prevent extremely small mass and bad conditioning in the Poisson system, the nodal distance value at the cut cell is usually clamped to be larger than some threshold (such as $0.1\Delta x$). Such clamping is also adopted in the Ghost Fluid Method [Gibou et al. 2002] for robustly estimating extrapolated ghost pressure near the free surface. Since we use particle quadratures near the free surface and the solid-fluid interface, we achieve a similar conditioning improvement by shifting the quadrature point locations to make sure they are at least component-wise $0.1\Delta x$ away from element boundaries. Note that we only imaginarily shift them while evaluating the integrals in the weak form. During the normal particle-grid transfers, the material particle positions maintain their true values. We observe in our examples that doing so eliminates occasional instability from PCG failures. We collect statistics from the dam jello example (Fig. 9). With our shifting strategy, the average per-step PCG iteration reduces from 3.3 to 2.9. However, the maximum per-step PCG iteration reduces from 81 to 7 with a more than 6× timing difference.

6.2.3 Normal estimation. Our method requires high quality normal estimation at interface quadrature locations. For mesh-based solids, normals can usually be computed from the boundary mesh tessellation. For particles, a common practice is to construct a level set through a union of spheres [Boyd and Bridson 2012], reinitialize the signed distance function, and evaluate the level set gradient function on query points. In our method, we only need accurate normals near a narrow band of the solid boundary. Therefore we adopt a much more simple and efficient approach by picking normals to be the negative mass gradient field evaluated at the particles. After solid particles transfer their masses onto the grid ($m_i^s \leftarrow m_p^s$), we compute the mass gradient on each particle

$$(\nabla m^s)(\mathbf{x}_p^s) = \sum_i m_i^s \nabla N_i^s(\mathbf{x}_p^s) \tag{46}$$

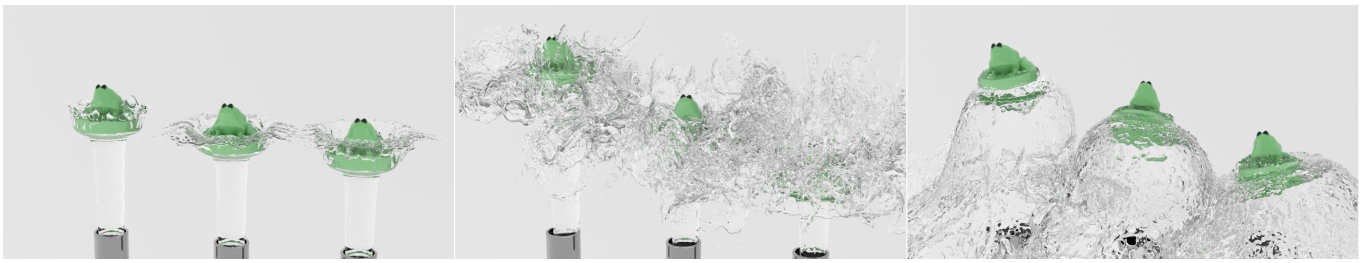


Fig. 17. **Frogs.** Three frogs, ranging from light to heavy, are lifted by fountains while creating beautiful spray patterns during this vibrant two-way coupling.

and let $\mathbf{n}_p^s = -(\nabla m^s)(\mathbf{x}_p^s)/|(\nabla m^s)(\mathbf{x}_p^s)|$. In Fig. 16 we compare the normal estimation results on a 2D geometry using a quadratic B-spline kernel versus using a level set. Note that for highly sparse and isolated particles such as Fig. 19, our normal estimation is more inaccurate than a high-resolution level-set result. However, such inaccuracy does not cause stability issues or artifacts in our experiments.

6.2.4 Collision object. The fluid domain collides with kinematic objects through the free-slip boundary condition defined through $\partial\Omega_S^f$. To support free wall separation ($\mathbf{v}^{f,n+1}|_{\partial\Omega_s^f} \cdot \mathbf{n}^f \geq v_s^f$), an LCP formulation is needed as pointed out by Batty *et al.* [2007] and we leave this investigation to future work. There are multiple choices for colliding the solid domain with walls. A standard way in MPM is to prescribe Dirichlet nodal (normal) velocities on the grid nodes inside the wall. Another choice is to enforce the boundary condition through the weak form as we do for the fluids. Unfortunately, neither of the two methods supports free separation, which is a much more important concern for solids. We instead take an energy-based approach following McAdams *et al.* [2011] by assigning per solid particle a unilateral anisotropic potential energy $\Psi(\mathbf{x}_p, \mathbf{x}_c) = k(\mathbf{x}_p - \mathbf{x}_c)^T \mathbf{n}\mathbf{n}^T(\mathbf{x}_p - \mathbf{x}_c)/2$ where $\mathbf{x}_c$ is the closest surface point to $\mathbf{x}_p^n$ and $\mathbf{n}$ is the normalized $\mathbf{x}_c - \mathbf{x}_p^n$. Here "unilateral" means we change the energy, force, and force derivatives to zeroes if $(\mathbf{x}_p - \mathbf{x}_c) \cdot \mathbf{n} < 0$. We add this energy to all penetrated particles and integrate the force implicitly using the MPM Lagrangian energy treatment in Jiang *et al.* [2015].

6.2.5 Position correction. In traditional hybrid Lagrangian-Eulerian fluid simulations including FLIP [Zhu and Bridson 2005] and APIC [Jiang et al. 2015], due to truncation errors in the advection, particles may get clumpy, sparse, lose/gain volume, or penetrate into collision objects. Various approaches in computer graphics have targeted this issue by correcting particle positions after advection [Ando et al. 2012; Ando and Tsuruno 2011; Boyd and Bridson 2012; Kugelstadt et al. 2019]. Our method is not immune to this problem, and we adopt Kugelstadt *et al.* [2019]'s method in our framework. The adaptation is straightforward, and we refer to Kugelstadt *et al.* [2019] for the algorithmic details, a comparison is shown in Fig. 18. One noteworthy feature is that our MPM particle-based representation of the solid object directly allows fluid densities near the solid-fluid interface to be correctly evaluated without suffering from underestimation.

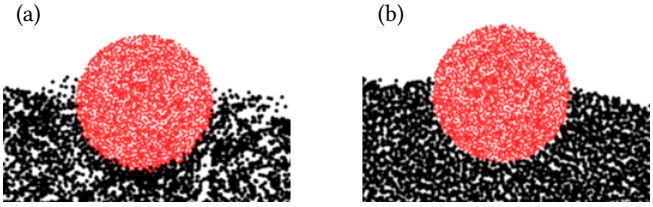


Fig. 18. **2D dambreak jello.** This example shows a 2D dam jello ball (zoomed in near the ball) (a) without and (b) with position correction. Position correction improves particle distibution, reduces volume loss, and prevents penetration from drifting.

7 RESULTS

The implementation overhead of our method is minimal since almost all operations specific to IQ-MPM can be performed and parallelized by launching standard MPM particle-grid transfer kernels. Our C++ code is released as open soruce. All simulations are performed on Intel Core i7-9700K with 32GB memory and Nvidia GTX 1050 Ti GPU for AMGCL. Timing and parameters are reported in Table 1. We reconstruct fluid surfaces with OpenVDB [Museth et al. 2019, 2013] and track sharp meshes for the solids following Wang *et al.* [2019].

Our method enables strong two-way coupling between MPM nonlinear hyperelastic solids and incompressible fluids, as demonstrated by shooting a bear with water jet (Fig. 11) and dropping a ball (Fig. 9) in a dam break. In Fig. 4, we demonstrate achieving a stable buoyancy effect for elastic solids with different densities relative to the liquid density. Our method supports arbitrary nonlinear hyperelasticity models such as the Fixed Corotated model [Stomakhin et al. 2012], the Neo-Hookean model, and the Saint Vernant-Kirchhoff model. We couple these three models with water in Fig. 2.

A significant advantage of MPM solids is its natural support for large deformation, plasticity, and topologically changing events such as fracturing and splitting-and-merge. We demonstrate these visual effects by showing a numerically fracturing water-filled armadillo (Fig. 23), squeezing plastic shaving cream onto water (Fig. 10), and injecting fluid onto a soil man (Fig. 19).

Traditional MPM [Jiang et al. 2016; Stomakhin et al. 2013] and augmented MPM [Stomakhin et al. 2014] suffer from numerical stickiness between different materials due to their interpolation of a single velocity field. IQ-MPM, on the other hand, allows solids and fluids to have discontinuous tangential velocities while enforcing a continuous interfacial pressure, as clearly demonstrated in Fig. 8 where a water ball is dropped onto an elastic bunny whose center is pinned in place.

Fig. 19. **Soil flush**. IQ-MPM can also animate the detailed interaction between fluid and granular media (modeled as an elastoplastic solid as in [Klár et al. 2016; Pradhana et al. 2017]).

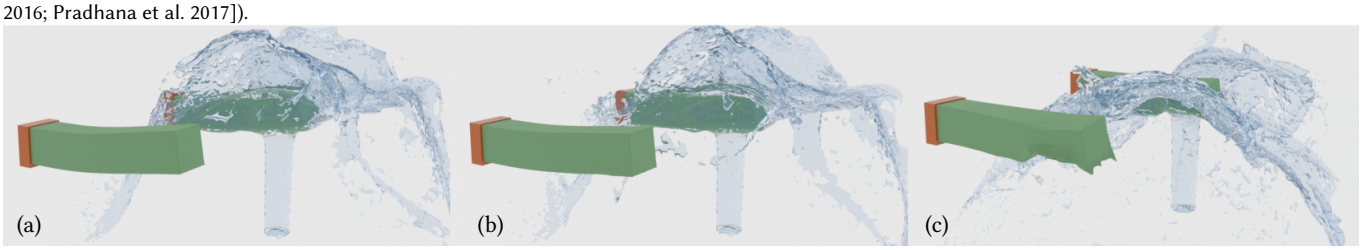


Fig. 20. **Time step size.** Breaking CFL may result in instability for stiff materials. (a) Δt is adaptively chosen by CFL so that no particles move more than $0.4\Delta x$ (the resulting Δt is around 0.001 in this example); (b) $\Delta t = 0.002$; (c) $\Delta t = 0.01$ (highly unstable).

Table 1. **Parameters and Timings.** Time per frame is provided as an average on an Intel Core i7-9700K and Nvidia GTX 1050 Ti GPU. Δt_{step} refers to the maximumly allowed time step for a time step. The actual time step is adaptively chosen by CFL so that no particles move more than $0.4\Delta x$ within a step. This CFL is necessary for MPM solids to prevent interpenetration [Gast et al. 2015]. We experiment with the Corotated model [McAdams et al. 2011], the fixed Corotated model* [Stomakhin et al. 2012], and also the †mixture of fixed Corotated, Neo-Hookean, and Saint Vernant-Kirchhoff models. We use the von Mises elastoplasticity model‡ [Gao et al. 2017] for the shaving cream example with a plastic yield stress of 100. The soil flush example adopts the granular elastoplasticity model $^{\diamond}$ [Stomakhin et al. 2013] with principal stretches restricted in $[1 - 2.5 \times 10^{-2}, 1 + 7.5 \times 10^{-3}]$.

Example	min/frame	Δt_{frame}	Δx	Δt_{step}	N_{solid}	N_{fluid}	Young's Modulus	Poisson's Ratio	Solid Density	Fluid Density
(Fig. 2) Balls†	12.8	1/48	0.008	0.004	0.78M	1.1M	5×10^5	0.3	0.8×10^3	1×10^3
(Fig. 3) Flush rubber	0.8	1/48	0.008	0.004	0.33M	0.5M	1×10^5	0.3	1×10^3	1×10^3
(Fig. 4) Buoyancy	2.5	1/24	0.01	0.004	0.3M	6.1M	1×10^5	0.4	$0.2, 0.7, 1.2 \times 10^3$	1×10^3
(Fig. 9) Dam jello ball*	5.3	1/24	0.0156	0.003	0.14M	2.35M	5×10^4	0.4	0.5×10^3	1×10^3
(Fig. 8) Bunny	3.4	1/48	0.01	0.005	0.5M	0.45M	1×10^5	0.4	1×10^3	$0.2, 1, 2 \times 10^3$
(Fig. 10) Shaving cream‡	4.5	1/24	0.01	0.004	0.5M	0.7M	1×10^5	0.4	0.2×10^3	1×10^3
(Fig. 17) Frogs	3.1	1/96	0.01	0.004	0.28M	0.9M	5×5^5	0.3	$0.3, 0.6, 0.9 \times 10^3$	1×10^3
(Fig. 11) Bear bath	2.4	1/48	0.01	0.004	0.32M	0.34M	1×10^5	0.4	1×10^3	1×10^3
(Fig. 19) Soil flush$^{\diamond}$	10.9	1/48	0.01	0.001	1.26M	1.15M	1.4×10^5	0.2	1×10^3	0.4×10^3
(Fig. 23) Armadillo toy	3.6	1/24	0.005	0.004	0.2M	0.15M	2×10^4	0.4	1×10^3	1×10^3

In Fig. 21 we show the runtime breakdown for a representative time step of the "dam jello" example (Fig. 9). AMGCL typically requires around 25 CG iterations, while the construction of the AMG preconditioner takes a considerable amount of time. Another major cost comes from building the system matrix, which we believe can be largely improved with more optimized sparse matrix operations in future work. This part also includes a neighborhood search to detect fluid-elastica interface. After contact, the DOF count increases and slows down the solve by 30% due to the occurrence of interfacial pressure unknowns.

Time step size. The coupling based on our splitting scheme is not unconditionally stable since it is semi-implicit (see also the discussion by Teng et al. [2016]) In Fig. 20, we simulate the interaction between a stiff elastic bar (with density 900 and Young's modulus 10^6) and water following an example in [Chentanez et al. 2006]. We use $\Delta x = 0.2$. When we set Δt accoring to CFL (such that particles do not move more than $0.4\Delta x$ per time step), the resulting Δt is around 0.001 and the simulation remains stable. A large step size such as 0.01, however, results in instability during coupling.

Choices of interpolation kernels. To illustrate the benefit of "B2B0-B0-B1B0", we compare it with "B2B1-B1-B2B1" in a 2D fluid-elastica interaction simulation (Fig. 22). The elastic object's density, Young's modulus, and Poisson's ratio are 2000, 10^5, and 0.4 respectively. The fluid's density is 1000. The two simulations give qualitatively similar visual qualities, but very different computational performance. While the solid projected Newton steps costs the same amount of time (0.012s per time step), "B2B0-B0-B1B0" requires around 3× less time in building (0.016s vs 0.049s) and solving (0.019s vs 0.052s) the

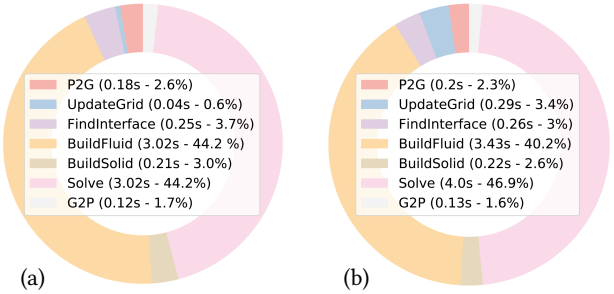


Fig. 21. **Timing breakdown.** In the "dam jello" example (Fig. 9), the average per-step timing costs before (a) and after (b) contact are 6.84s and 8.53s respectively. The coupling results in some overhead in building and solving the system.

coupling system with 4.5× fewer non-zero entries (49051 vs 223024) in the matrix.

8 DISCUSSIONS AND LIMITATIONS

We presented a novel algorithm for strong two-way coupling of incompressible fluids with volumetric elastic solids based on MPM. With CFL-rate time step sizes, our approach yields stable solutions in various challenging examples, and only requires a *single* monolithic solve, even for highly non-linear elastic solids. To allow the fluid to freely slide against the solid, we designed a novel interface quadrature scheme that is simple to implement, and does not require additional level set or explicit mesh representations for treating the interfacial boundary conditions, as employed in FLIP schemes [Boyd and Bridson 2012]. We demonstrated the benefit of our framework on several end-to-end 3D examples with complex solid-fluid interactions.

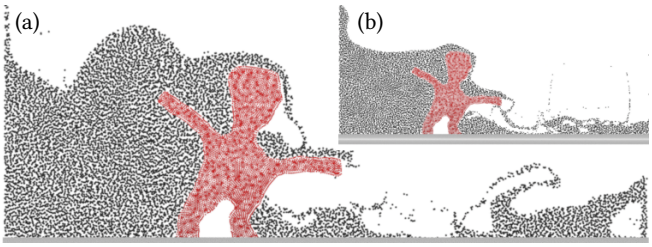


Fig. 22. **Choices of kernels.** Comparing (a)"B2B0-B0-B1B0" and (b)"B2B1-B1-B2B1" result in similar visual appearances. But "B2B0-B0-B1B0" is 3× faster in building and solving the 4.5× sparser coupling system.

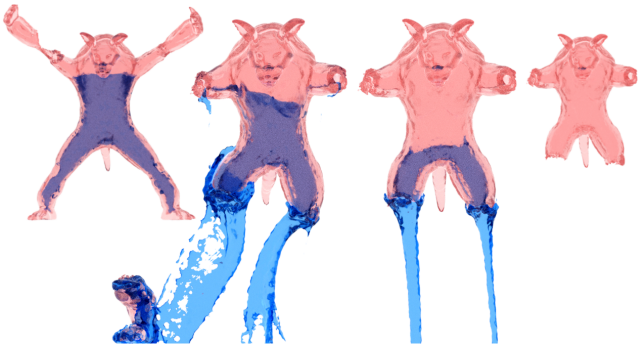


Fig. 23. **Armadillo toy.** A hyperelastic hollow toy armadillo is filled with water. When getting violently stretched, the elastic structure feels the water pressure; as it gets torn apart, water flows out naturally and completely.

There are several avenues for future research, as we list below. It would be interesting to extend our work to support fluid interactions with co-dimensional solids, such as hair [Fei et al. 2017] and cloth [Fei et al. 2018]. Our current framework uses the same grid resolution for both solids and fluids, and allowing for different grid resolutions [Aanjaneya et al. 2017; Akinci et al. 2013; Ando et al. 2013; Chentanez and Muller 2011; English et al. 2013; Ferstl et al. 2014; Gao et al. 2017] could provide richer solid-fluid interactions. The particle distribution inside MPM solids can develop big gaps in the presence of large deformations (if the solid does not fracture). In this case, the fluid MPM particles will creep into these gaps, rather than freely sliding past. Deriving an adaptive version of our interface quadrature scheme could alleviate such artifacts. It would also be interesting to develop a narrow band version of our solver following [Ferstl et al. 2016; Sato et al. 2018]. Incorporating viscosity similarly to the work of Batty and Bridson [2008] through a weak form discretization is also an exciting future work.

Our ghost matrix formulation is built on top of the approximation that we delay the non-orthogonal influence of the matrix pressure to the solid stress. A more accurate treatment of the velocities after solving for the pressures would be substituting the pressures back into Eq. (11) (rather than Eq. (18)) and iterating until convergence. However, for effects focused by this paper, our approximation achieves visually plausible behaviors with a much smaller computational cost.

MPM solids suffer from numerical plasticity with large time steps, while this is not the case for incompressible fluids [Chentanez and Müller 2012; Lentine et al. 2012]. This can also lead to stability issues with two-way coupling. At present, we use time steps that respect the CFL condition for the solids to avoid potential issues, but it would be beneficial to design methods that do not have *any* time step restrictions, for scaling to larger problem sizes. While our weak form discretization properly enforces the free-slip boundary condition for solids and fluids, more research is required to allow two solids to freely slide against one another [Levin et al. 2011]. This is because we extend the fluid domain to closely touch the solid, to achieve a consistent interfacial pressure. Such a scheme would have stability issues for pure solids. Furthermore, to correctly resolve solid-solid slip interaction in MPM, one must correctly handle self-collision and new interfaces generated from fracture, which often belong to the same solid. Finally, while we considered two-way coupling of MPM solids with MPM liquids, for wide-scale adoption, it would be interesting to design a consistent discretization for two-way coupling MPM solids with FLIP-based liquids.

ACKNOWLEDGMENTS

We would like to thank Joshuah Wolper for proofreading and narrating the video, and the anonymous reviewers for their valuable

comments. Penn authors were supported in part by NSF CAREER (IIS-1943199) and CCF-1813624, DOE ORNL contract 4000171342, a gift from Adobe Inc., NVIDIA GPU grants, and Houdini licenses from SideFX. M. A. was supported in part by the Rutgers University start-up grant, and the Ralph E. Powe Junior Faculty Enhancement Award. Y. Z. was supported in part by ONR MURI N00014-16-1-2007, DARPA XAI N66001-17-2-4029, and ONR N00014-19-1-2153.

REFERENCES

M. Aanjaneya. 2018. An Efficient Solver for Two-way Coupling Rigid Bodies with Incompressible Flow. *Comp Graph Forum* 37, 8 (2018), 59–68.

M. Aanjaneya, M. Gao, H. Liu, C. Batty, and E. Sifakis. 2017. Power Diagrams and Sparse Paged Grids for High Resolution Adaptive Liquids. *ACM Trans Graph* 36, 4, Article 140 (2017), 12 pages.

M. Akbay, N. Nobles, V. Zordan, and T. Shinar. 2018. An Extended Partitioned Method for Conservative Solid-Fluid Coupling. *ACM Trans Graph* 37, 4, Article 86 (2018).

N. Akinci, G. Akinci, and M. Teschner. 2013. Versatile surface tension and adhesion for SPH fluids. *ACM Trans on Graph (TOG)* 32, 6 (2013), 182.

N. Akinci, M. Ihmsen, G. Akinci, B. Solenthaler, and M. Teschner. 2012. Versatile Rigid-Fluid Coupling for Incompressible SPH. *ACM Trans Graph* 31, 4, Article 62 (2012).

R. Ando, N. Thuerey, and C. Wojtan. 2015. A Stream Function Solver for Liquid Simulations. *ACM Trans Graph* 34, 4, Article Article 53 (2015).

R. Ando, N. Thurey, and R. Tsuruno. 2012. Preserving fluid sheets with adaptively sampled anisotropic particles. *IEEE Trans Vis Comp Graph* 18, 8 (2012), 1202–1214.

R. Ando, N. Thurey, and C. Wojtan. 2013. Highly adaptive liquid simulations on tetrahedral meshes. *ACM Trans Graph* 32, 4 (2013), 103:1–103:10.

R. Ando and R. Tsuruno. 2011. A particle-based method for preserving fluid sheets. In *Symp. Comp. Anim.* 7–16.

O. E. Arash, O. Génevaux, A. Habibi, and J. m. Dischler. 2003. Simulating Fluid-Solid Interaction. In *in Graph Interface*. 31–38.

C. Batty, F. Bertails, and R. Bridson. 2007. A fast variational framework for accurate solid-fluid coupling. *ACM Trans Graph* 26, 3 (2007).

C. Batty and R. Bridson. 2008. Accurate viscous free surfaces for buckling, coiling, and rotating liquids. *Symp. Comp. Anim.* (2008), 219–228.

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

L. Boyd and R. Bridson. 2012. MultiFLIP for energetic two-phase fluid simulation. *ACM Trans Graph* 31, 2 (2012), 16:1–16:12.

R. Bridson. 2015. *Fluid simulation for computer graphics.*

T. Brochu, C. Batty, and R. Bridson. 2010. Matching fluid simulation elements to surface geometry and topology. *ACM Trans Graph* 29, 4 (2010), 47:1–47:9.

M. Carlson, P. J. Mucha, and G. Turk. 2004. Rigid Fluid: Animating the Interplay Between Rigid Bodies and Fluid. *ACM Trans Graph* 23, 3 (2004), 377–384.

N. Chentanez, T. G. Goktekin, B. E. Feldman, and J. F. O'Brien. 2006. Simultaneous Coupling of Fluids and Deformable Bodies. In *Proc of the 2006 ACM SIGGRAPH/EuroGraph Symp on Comp Anim (SCA '06)*. 83–89.

N. Chentanez and M. Muller. 2011. Real-time Eulerian water simulation using a restricted tall cell grid. *ACM Trans Graph* 30, 4 (2011), 82:1–82:10.

N. Chentanez and M. Müller. 2012. Mass-Conserving Eulerian Liquid Simulation. In *Proc of the ACM SIGGRAPH/EuroGraph Symp on Comp Anim.* 245âĂŞ254.

M. Christon. 2002. Dealing with pressure: FEM solution strategies for the pressure in the time-dependent Navier–Stokes equations. *International journal for numerical methods in fluids* 38, 12 (2002), 1177–1198.

P. Clausen, M. Wicke, J. R. Shewchuk, and J. F. O'Brien. 2013. Simulating Liquids and Solid-Liquid Interactions with Lagrangian Meshes. *ACM Trans Graph* 32, 2, Article 17 (2013).

F. d. Goes, C. Wallez, J. Huang, D. Pavlov, and M. Desbrun. 2015. Power Particles: An Incompressible Fluid Solver Based on Power Diagrams. *ACM Trans Graph* 34, 4, Article Article 50 (2015).

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).

D. Demidov. 2019. AMGCL: An Efficient, Flexible, and Extensible Algebraic Multigrid Implementation. *Lobachevskii Journal of Mathematics* 40, 5 (01 May 2019), 535–546. https://doi.org/10.1134/S1995080219050056

H. Elman, D. Silvester, and A. Wathen. 2014. *Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics.* Oxford University Press, USA.

R. E. English, L. Qiu, Y. Yu, and R. Fedkiw. 2013. Chimera Grids for Water Simulation. In *Proc of the 12th ACM SIGGRAPH/EuroGraph Symp on Comp Anim.* 85âĂŞ94.

Y. Fan, J. Litven, D. Levin, and D. Pai. 2013. Eulerian-on-lagrangian Simulation. *ACM Trans Graph* 32, 3 (2013), 22:1–22:9.

Y. Fan, J. Litven, and D. Pai. 2014. Active Volumetric Musculoskeletal Systems. *ACM Trans Graph* 33, 4 (2014), 152:1–152:9.

Y. Fang, Y. Hu, S. M. Hu, and C. Jiang. 2018. A temporally adaptive material point method with regional time stepping. *Comp Graph Forum* 37, 8 (2018), 195–204.

Y. Fang, M. Li, M. Gao, and C. Jiang. 2019. Silly rubber: an implicit material point method for simulating non-equilibrated viscoelastic and elastoplastic solids. *ACM Trans on Graph (TOG)* 38, 4 (2019), 1–13.

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.

Y. Fei, H. T. Maia, C. Batty, C. Zheng, and E. Grinspun. 2017. A Multi-Scale Model for Simulating Liquid-Hair Interactions. *ACM Trans Graph* 36, 4, Article Article 56 (2017).

B. Feldman, J. O'Brien, B. Klingner, and T. Goktekin. 2005. Fluids in Deforming Meshes. In *Symp on Comp Anim (SCA '05)*. 255–259.

F. Ferstl, R. Ando, C. Wojtan, R. Westermann, and N. Thuerey. 2016. Narrow band FLIP for liquid simulations. In *Computer Graphics Forum*, Vol. 35. 225–232.

F. Ferstl, R. u. d. Westermann, and C. Dick. 2014. Large-scale liquid simulation on adaptive hexahedral grids. *IEEE Trans on visualization and Comp Graph* 20, 10 (2014), 1405–1417.

N. Foster and R. Fedkiw. 2001. Practical Animation of Liquids. In *Proc. of ACM SIGGRAPH 2001*. 23–30.

N. Foster and D. Metaxas. 1996. Realistic animation of liquids. *Graph Mod Imag Proc* 58 (1996), 471–483.

C. Fu, Q. Guo, T. Gast, C. Jiang, and J. Teran. 2017. A polynomial particle-in-cell method. *ACM Trans on Graph (TOG)* 36, 6 (2017), 222.

S. Gagniere, D. Hyde, A. Marquez-Razon, C. Jiang, Z. Ge, X. Han, Q. Guo, and J. Teran. 2020. A Hybrid Lagrangian/Eulerian Collocated Advection and Projection Method for Fluid Simulation. *ArXiv e-prints* (2020).

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.

M. Gao, A. Pradhana, C. Jiang, and E. Sifakis. 2017. An adaptive generalized interpolation material point method for simulating elastoplastic materials. *ACM Trans on Graph (TOG)* 36, 6 (2017), 223.

M. Gao, X. Wang, K. Wu, A. Pradhana, E. Sifakis, C. Yuksel, and C. Jiang. 2018b. GPU Optimization of Material Point Methods. *ACM Trans Graph* 37, 6, Article 254 (2018), 12 pages.

Y. Gao, C. Li, S. Hu, and B. Barsky. 2009. Simulating gaseous fluids with low and high speeds. *Comp Graph Forum* 28, 28 (2009), 1845–1852.

T. Gast, C. Schroeder, A. Stomakhin, C. Jiang, and J. Teran. 2015. Optimization Integrator for Large Time Steps. *IEEE Trans Vis Comp Graph* 21, 10 (2015), 1103–1115.

F. Gibou, R. P. Fedkiw, L. T. Cheng, and M. Kang. 2002. A second-order-accurate symmetric discretization of the Poisson equation on irregular domains. *J of Comp Phys* 176, 1 (2002), 205–227.

C. Gissler, A. Peer, S. Band, J. Bender, and M. Teschner. 2019. Interlinked SPH pressure solvers for strong fluid-rigid coupling. *ACM Trans Graph* 38, 1 (2019), 1–13.

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

E. Guendelman, A. Selle, F. Losasso, and R. Fedkiw. 2005. Coupling Water and Smoke to Thin Deformable and Rigid Shells. *ACM Trans Graph* 24, 3 (2005), 973–981.

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.

X. Han, T. F. Gast, Q. Guo, S. Wang, C. Jiang, and J. Teran. 2019. A Hybrid Material Point Method for Frictional Contact with Diverse Materials. *Proc. ACM Comput. Graph Interact. Tech.* 2, 2, Article Article 17 (2019).

F. Harlow and E. Welch. 1965. Numerical Calculation of Time Dependent Viscous Flow of Fluid with a Free Surface. *Phys Fluid* 8, 12 (1965), 2182–2189.

J. Hong, H. Lee, J. Yoon, and C. Kim. 2008b. Bubbles Alive. *ACM Trans Graph* 27, 3 (2008), 48:1–48:4.

W. Hong, D. House, and J. Keyser. 2008a. Adaptive particles for incompressible fluid simulation. *Vis Comp* 24, 7 (2008), 535–543.

P. Hood and C. Taylor. 1974. Navier-Stokes equations using mixed interpolation. *Finite element methods in flow problems* (1974), 121–132.

Y. Hu, Y. Fang, Z. Ge, Z. Qu, Y. Zhu, A. Pradhana, and C. Jiang. 2018. A moving least squares material point method with displacement discontinuity and two-way rigid body coupling. *ACM Trans Graph* 37, 4 (2018), 150.

Y. Hu, J. Liu, A. Spielberg, J. B. Tenenbaum, W. T. Freeman, J. Wu, D. Rus, and W. Matusik. 2019. ChainQueen: A Real-Time Differentiable Physical Simulator for Soft Robotics. *Proc of the Int Conf on Robotics and Automation (ICRA)* (2019), 6265–6271.

T. J. Hughes. 2012. *The finite element method: linear static and dynamic finite element analysis.*

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. 2017. An angular momentum conserving affine-particle-in-cell method. *J. Comput. Phys.* 338 (2017).

C. Jiang, C. Schroeder, J. Teran, A. Stomakhin, and A. Selle. 2016. The material point method for simulating continuum materials. In *SIGGRAPH Course*. 24:1–24:52.

G. Klár, T. Gast, A. Pradhana, C. Fu, C. Schroeder, C. Jiang, and J. Teran. 2016. Drucker-prager elastoplasticity for sand animation. *ACM Trans Graph* 35, 4 (2016), 103:1–103:12.

B. Klingner, B. Feldman, N. Chentanez, and J. O'Brien. 2006. Fluid Animation with Dynamic Meshes. In *Proc. of ACM SIGGRAPH (SIGGRAPH '06)*. 820–825.

T. Kugelstadt, A. Longva, N. Thurey, and J. Bender. 2019. Implicit Density Projection for Volume Conserving Liquids. *IEEE Trans on Visualization and Comp Graph* (2019).

H. Lee, J. Hong, and C. Kim. 2009. Interchangeable SPH and level set method in multiphase fluids. *Vis Comp* 25, 5 (2009), 713–718.

S. Lee, R. Yu, J. Park, M. Aanjaneya, E. Sifakis, and J. Lee. 2018. Dexterous Manipulation and Control with Volumetric Muscles. *ACM Trans Graph* 37, 4, Article 57 (2018), 13 pages.

M. Lentine, M. Cong, S. Patkar, and R. Fedkiw. 2012. Simulating Free Surface Flow with Very Large Time Steps. In *Proc of the ACM SIGGRAPH/EuroGraph Symp on Comp Anim.*

D. Levin, J. Litven, G. Jones, S. Sueda, and D. Pai. 2011. Eulerian Solid Simulation with Contact. *ACM Trans Graph* 30, 4 (2011), 36:1–36:10.

D. Li, S. Sueda, D. Neog, and D. Pai. 2013. Thin Skin Elastodynamics. *ACM Trans Graph* 32, 4 (2013), 49:1–49:10.

F. Losasso, F. e. d. e. r. Gibou, and R. Fedkiw. 2004. Simulating water and smoke with an octree data structure. In *ACM Trans on Graph*, Vol. 23. 457–462.

F. Losasso, J. Talton, N. Kwatra, and R. Fedkiw. 2008. Two-way coupled SPH and Particle Level Set fluid simulation. *IEEE Trans Vis Comp Graph* 14 (2008), 797–804.

E. Love and D. Sulsky. 2006. An unconditionally stable, energy-momentum consistent implementation of the the material point method. *Comp Meth App Mech Eng* 195 (2006), 3903–3925.

L. K. Ma, Y. Zhang, Y. Liu, K. Zhou, and X. Tong. 2017. Computational Design and Fabrication of Soft Pneumatic Objects with Desired Deformations. *ACM Trans Graph* 36, 6, Article 239 (2017), 12 pages.

M. Macklin, M. Muller, N. Chentanez, and T. Kim. 2014. Unified particle physics for real-time applications. *ACM Trans Graph* 33, 4 (2014), 153:1–153:12.

C. Mast, P. Mackenzie-Helnwein, P. Arduino, G. R. Miller, and W. Shin. 2012. Mitigating kinematic locking in the material point method. *J of Comp Phys* 231, 16 (2012), 5351–5373.

H. Mazhar, T. Heyn, D. Negrut, and A. Tasora. 2015. Using Nesterov's method to accelerate multibody dynamics with friction and contact. *ACM Trans Graph* 34, 3 (2015), 32:1–32:14.

A. McAdams, A. Selle, K. Ward, E. Sifakis, and J. Teran. 2009. Detail Preserving Continuum Simulation of Straight Hair. *ACM Trans Graph* 28, 3 (2009), 62:1–62:6.

A. McAdams, E. Sifakis, and J. Teran. 2010. A Parallel Multigrid Poisson Solver for Fluids Simulation on Large Grids. In *Proc of the 2010 ACM SIGGRAPH/EuroGraph Symp on Comp Anim (SCA '10)*. 65–74.

A. McAdams, Y. Zhu, A. Selle, M. Empey, R. Tamstorf, J. Teran, and E. Sifakis. 2011. Efficient elasticity for character skinning with contact and collisions. In *ACM Trans on Graph (TOG)*, Vol. 30. 37.

M. K. Misztal, K. Erleben, A. Bargteil, J. Fursund, B. B. Christensen, J. A. Bærentzen, and R. Bridson. 2012. Multiphase Flow of Immiscible Fluids on Unstructured Moving Meshes. In *Proc of the ACM SIGGRAPH/EuroGraph Symp on Comp Anim (SCA '12)*. 97–106.

N. Mitchell, C. Cutting, and E. Sifakis. 2015. GRIDiron: An Interactive Authoring and Cognitive Training Foundation for Reconstructive Plastic Surgery Procedures. *ACM Trans Graph* 34, 4, Article 43 (2015), 12 pages.

M. Müller, N. Chentanez, T. Kim, and M. Macklin. 2015. Air Meshes for Robust Collision Handling. *ACM Trans Graph* 34, 4 (2015), 133:1–133:9.

K. Museth, N. Avramoussis, and D. Bailey. 2019. OpenVDB. In *ACM SIGGRAPH 2019 Courses*. 1–56.

K. Museth, J. Lait, J. Johanson, J. Budsberg, R. Henderson, M. Alden, P. Cucka, D. Hill, and A. Pearce. 2013. OpenVDB: An Open-source Data Structure and Toolkit for High-resolution Volumes. In *ACM SIGGRAPH 2013 Courses*. Article 19, 1 pages.

K. Nagasawa, T. Suzuki, R. Seto, M. Okada, and Y. Yue. 2019. Mixing sauces: a viscosity blending model for shear thinning fluids. *ACM Transactions on Graphics (TOG)* 38, 4 (2019), 1–17.

R. Narain, M. Overby, and G. E. Brown. 2016. ADMM ⊇ Projective Dynamics: Fast Simulation of General Constitutive Models. In *Symp Comp Anim*. 21–28.

J. Nocedal and S. J. Wright. 2006. *Numerical Optimization*.

S. Patkar, M. Aanjaneya, D. Karpman, and R. Fedkiw. 2013. A hybrid Lagrangian-Eulerian formulation for bubble generation and dynamics. In *Proc ACM SIGGRAPH/Eurograp Symp Comp Anim*. 105–114.

A. Peer, C. Gissler, S. Band, and M. Teschner. 2018. An implicit SPH formulation for incompressible linearly elastic solids. In *Computer Graphics Forum*, Vol. 37. 135–148.

C. S. Peskin. 2002. The immersed boundary method. *Acta numerica* 11 (2002), 479–517.

A. Pradhana, 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. 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 *Symp. Comp. Anim.* 157–163.

N. Rasmussen, D. Enright, D. Nguyen, S. Marino, N. Sumner, W. Geiger, S. Hoon, and R. Fedkiw. 2004. Directable Photorealistic Liquids. In *Proc of the 2004 ACM SIGGRAPH/EuroGraph Symp on Comp Anim*.

K. Raveendran, C. Wojtan, and G. Turk. 2011. Hybrid SPH. In *Proc 2011 ACM SIGGRAPH/Eurograp Symp Comp Anim*. 33–42.

A. Robinson-Mosher, C. Schroeder, and R. Fedkiw. 2011. A Symmetric Positive Definite Formulation for Monolithic Fluid Structure Interaction. *J. Comput. Phys.* 230, 4 (2011), 1547–1566.

A. Robinson-Mosher, T. Shinar, J. Gretarsson, J. Su, and R. Fedkiw. 2008. Two-way Coupling of Fluids to Rigid and Deformable Solids and Shells. *ACM Trans Graph* 27, 3, Article 46 (2008), 9 pages.

D. Roble, N. b. Zafar, and H. Falt. 2005. Cartesian Grid Fluid Simulation with Irregular Boundary Voxels. In *ACM SIGGRAPH 2005 Sketches (SIGGRAPH '05)*. Article 138.

T. Rüberg and F. Cirak. 2012. Subdivision-stabilised immersed b-spline finite elements for moving boundary flows. *Comp Meth in App Mech and Eng* 209 (2012), 266–283.

E. S., S. Marino, and J. Teran. 2008. Globally Coupled Collision Handling Using Volume Preserving Impulses. In *Symp. Comp. Anim.* 147–153.

P. Sachdeva, S. Sueda, S. Bradley, M. Fain, and D. K. Pai. 2015. Biomechanical Simulation and Control of Hands and Tendinous Systems. *ACM Trans Graph* 34, 4, Article Article 42 (2015).

T. Sato, C. Wojtan, N. Thuerey, T. Igarashi, and R. Ando. 2018. Extended narrow band FLIP for liquid simulations. In *Computer Graphics Forum*, Vol. 37. 169–177.

R. Setaluri, M. Aanjaneya, S. Bauer, and E. Sifakis. 2014. SPGrid: A Sparse Paged Grid Structure Applied to Adaptive Smoke Simulation. *ACM Trans Graph* 33, 6, Article 205 (2014), 12 pages.

E. Sifakis and J. Barbic. 2012. FEM Simulation of 3D Deformable Solids: A Practitioner's Guide to Theory, Discretization and Model Reduction. In *ACM SIGGRAPH 2012 Courses*. Article 20.

F. Sin, A. Bargteil, and J. Hodgins. 2009. A point-based method for animating incompressible flow. In *Symp. Comp. Anim.* 247–255.

B. Solenthaler, J. u. r. Schläfli, and R. Pajarola. 2007. A Unified Particle Model for Fluid and Solid Interactions: Research Articles. *Comput. Animat. Virtual Worlds* 18, 1 (2007), 69–82.

M. Steffen, R. M. Kirby, and M. Berzins. 2008. Analysis and reduction of quadrature errors in the material point method (MPM). *Int J Numer Meth Eng* 76, 6 (2008), 922–948.

A. Stomakhin, R. Howes, C. Schroeder, and J. Teran. 2012. Energetically consistent invertible elasticity. In *Proc Symp Comp Anim*. 25–32.

A. Stomakhin, C. Schroeder, L. Chai, J. Teran, and A. Selle. 2013. A material point method for snow simulation. *ACM Trans Graph* 32, 4 (2013), 102:1–102:10.

A. Stomakhin, C. Schroeder, C. Jiang, L. Chai, J. Teran, and A. Selle. 2014. Augmented MPM for phase-change and varied materials. *ACM Trans Graph* 33, 4 (2014), 138:1–138:11.

S. Sueda, G. Jones, D. Levin, and D. Pai. 2011. Large-scale Dynamic Simulation of Highly Constrained Strands. *ACM Trans Graph* 30, 4 (2011), 39:1–39:10.

D. Sulsky, S. Zhou, and H. Schreyer. 1995. Application of a particle-in-cell method to solid mechanics. *Comp Phys Comm* 87, 1 (1995), 236–252.

Y. Teng, D. I. W. Levin, and T. Kim. 2016. Eulerian Solid-fluid Coupling. *ACM Trans Graph* 35, 6, Article 200 (2016), 8 pages.

J. Teran, E. Sifakis, G. Irving, and R. Fedkiw. 2005. Robust quasistatic finite elements and flesh simulation. In *Proc of the 2005 ACM SIGGRAPH/EuroGraph Symp on Comp Anim*. 181–190.

S. Wang, M. Ding, T. F. Gast, L. Zhu, S. Gagniere, C. Jiang, and J. M. Teran. 2019. Simulation and Visualization of Ductile Fracture with the Material Point Method. *Proc of the ACM on Comp Graph and Interactive Techniques* 2, 2 (2019), 1–20.

X. Wang, M. Li, Y. Fang, X. Zhang, M. Gao, M. Tang, D. Kaufman, and C. Jiang. 2020. Hierarchical Optimization Time Integration for CFL-rate MPM Stepping. *ACM Trans on Graph (TOG)* (2020).

N. J. Weidner, K. Piddington, D. I. W. Levin, and S. Sueda. 2018. Eulerian-on-Lagrangian Cloth Simulation. *ACM Trans Graph* 37, 4, Article Article 50 (2018).

J. Wolper, Y. Fang, M. Li, J. Lu, M. Gao, and C. Jiang. 2019. CD-MPM: Continuum Damage Material Point Methods for Dynamic Fracture Animation. *ACM Trans Graph* 38, 4, Article 119 (2019).

X. Yan, C. F. Li, X. S. Chen, and S. M. Hu. 2018. MPM simulation of interacting fluids and solids. *Comp Graph Forum* 37, 8 (2018), 183–193.

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. Y. 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, Article 283 (2018), 19 pages.

O. Zarifi and C. Batty. 2017. A positive-definite cut-cell method for strong two-way coupling between fluids and deformable bodies. In *Symp on Comp Anim*. 7:1–7:11.

F. Zhang, X. Zhang, K. Y. Sze, Y. Lian, and Y. Liu. 2017. Incompressible material point method for free surface flow. *J of Comp Phys* 330 (2017), 92–110.

B. Zhu, X. Yang, and Y. Fan. 2010. Creating and preserving vortical details in SPH fluid. *Comp Graph Forum* 29, 7 (2010), 2207–2214.

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