Hybrid Grains: Adaptive Coupling of Discrete and Continuum Simulations of Granular Media

YONGHAO YUE*, The University of Tokyo
BREANNAN SMITH*, Columbia University
PETER YICHEN CHEN*, Columbia University
MAYTEE CHANTHARAYUKHONTHORN*, Massachusetts Institute of Technology
KEN KAMRIN+, Massachusetts Institute of Technology
EITAN GRINSPUN+, Columbia University

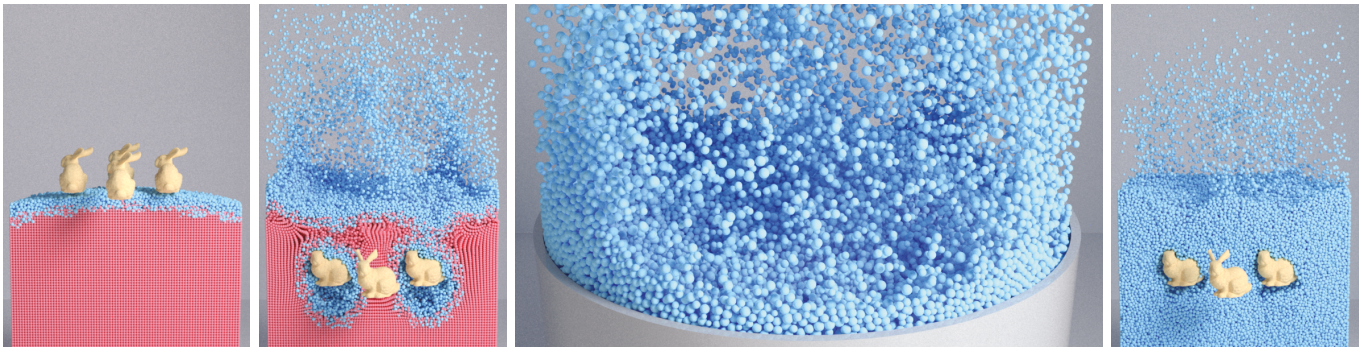


Fig. 1. We script four rigid bunnies to rapidly rotate about the vertical axis while simultaneously moving into and out of a packing of 360,000 grains. The bunnies inject a large amount of energy into the system, inducing significant displacements in the interior while producing an energetic splash near the free surface. Our hybrid algorithm for simulating granular materials couples a continuum treatment of the interior to a discrete element treatment of the free surface, inheriting the dual benefits of both approaches. The first two frames show a cutaway view of our hybrid algorithm, the third frame shows a full rendering of our hybrid algorithm, and the fourth frame shows a cutaway view of a fully discrete simulation. Our hybrid algorithm uses 88% fewer discrete grains and achieves a 6.84× speedup relative to a purely discrete simulation.

We propose a technique to simulate granular materials that exploits the dual strengths of discrete and continuum treatments. Discrete element simulations provide unmatched levels of detail and generality, but prove excessively costly when applied to large scale systems. Continuum approaches are computationally tractable, but limited in applicability due to built-in modeling assumptions; e.g., models suitable for granular flows typically fail to capture clogging, bouncing and ballistic motion. In our hybrid approach, an oracle dynamically partitions the domain into continuum regions where safe, and discrete regions where necessary. The domains overlap along transition zones, where a Lagrangian dynamics mass-splitting coupling principle enforces agreement between the two simulation states. Enrichment and homogenization operations allow the partitions to evolve over time. This approach accurately and efficiently simulates scenarios that previously required an entirely discrete treatment.

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

Additional Key Words and Phrases: Granular materials, Material point method, Contact dynamics, Constraints, Physical simulation

ACM Reference format:
Yonghao Yue*, Breannan Smith*, Peter Yichen Chen*, Maytee Chantharayukhonthorn*, Ken Kamrin+, and Eitan Grinspun+. 2018. Hybrid Grains: Adaptive Coupling of Discrete and Continuum Simulations of Granular Media. *ACM Trans. Graph.* 37, 6, Article 283 (November 2018), 19 pages.
https://doi.org/10.1145/3272127.3275095

*Co-first authors — authors contributed equally.
+Corresponding authors (e-mail: kkamrin@mit.edu, eitan@cs.columbia.edu).

0730-0301/2018/11-ART283 $15.00
https://doi.org/10.1145/3272127.3275095

1 INTRODUCTION

Granular materials, as a ubiquitous component of our natural world, are an important topic of study in fields ranging from the geosciences and film visual effects [Ammann et al. 2007] to robotics [Brown et al. 2010]. Granular materials trail only water as the most commonly handled industrial material [Richard et al. 2005], and a predictive understanding of granular materials is key to the industrial scale processing of construction materials, pharmaceuticals, food, and sand. Despite their importance, formulating predictive models of granular materials remains challenging, as they exhibit interesting behavior at multiple scales and phases.

When viewed from up close, the varied contact and tribological interactions between individual grains govern grain-scale behaviors such as packing, clogging, rolling and bouncing. These behaviors are accurately captured by discrete particle models; indeed,

such models are considered the "gold standard" and are often used as a ground-truth validation for continuum models [Rycroft et al. 2009]. Unfortunately, discrete particle models are computationally intractable when scaled to a large number of grains.

When viewed from afar, ensembles of grains can exhibit continuum behaviors of elastic solids (forming static piles), liquids (flowing plastically and sustaining permanent shape change), and gases (sparse grains in free flight ballistic motion) [Jaeger et al. 1996]. Continuum models of granular media do not explicitly represent individual grains, which makes these models amenable to the simulation of large numbers of grains. However, built in modeling assumptions specialize these models to certain phases and boundary conditions. For instance, a model that focuses on dense soils typically fails to capture other phenomena. Indeed, the state of the art in continuum modeling lacks a general, accurate, validated treatment of those grain-scale effects that can profoundly determine the evolution of the continuum-scale bulk: small scale rapid collisions near free surfaces allow individual grains to separate from the bulk material, the finite size nature of individual grains can lead to clogging in narrow chutes or during drainage scenarios, and grain-scale shear banding can mediate larger scale deformations.

We are seemingly faced with a harsh trade-off. We can choose a general, accurate, but slow discrete treatment; or a fast, scalable, but specialized continuum treatment. To provide a more nuanced trade-off, we propose to hybridize the discrete and continuum approaches.

We propose to apply a continuum model where it is safe to do so, while resorting to a discrete model where necessary (as in Fig. 1). We derive computational performance from the continuum model, and invest extra computation for grain-scale phenomena where needed.

In formulating a hybrid method to simulate granular materials, we introduce three core contributions: a method to efficiently couple a continuum material point simulation to a discrete element simulation, an adaptive granular oracle to identify regions that behave according to a continuum model (the regions, which for the purposes of this paper, are based off of distances from granular free surfaces), and a homogenization and coarsening technique to dynamically covert between continuum and discrete representations.

2 RELATED WORK

We draw on a wide body of literature on the modeling and simulation of granular materials with discrete and continuum treatments. We also draw inspiration from work on hybridizing other physical systems, particularly crystalline solids.

Discrete particle simulations. *Discrete particles*, originating in the molecular dynamics models of Alder and Wainwright [Alder and Wainwright 1957, 1959, 1960], have a long and successful history across scientific and engineering disciplines [Frenkel and Smit 2001; Hoover 1986; Rapaport 2004]. These methods explicitly model the behavior of individual particles and their subsequent interactions, and have been successfully extended as discrete element methods to simulate granular materials [Cundall and Strack 1979; Gallas et al. 1992; Haff and Werner 1986; Pöschel and Schwager 2005; Walton and Braun 1986]. These methods can be augmented to permit inelastic grain-level mechanics such as grain breakage/fragmentation, as arises in many complex real-world situations [Åström and Herrmann 1998; Ben-Nun et al. 2010; Nguyen et al. 2015; Tsoungui et al. 1999]. Discrete particle approaches employ either a "soft" or "hard" contact and friction formulation.

Penalty methods define a "soft" contact level constitutive relation and integrate the resulting forces. They are popular owing both to their simplicity and flexibility in defining interaction properties [Kruggel-Emden et al. 2007; Shäfer et al. 1996]. Penalty's disadvantages lie in setting parameters that often require multiple tuning and calibration passes, stiff, parasitic forces that are difficult to integrate, and challenges in modeling incompressibility. The graphics community has explored penalty-based simulations of granular materials, with works focused on the formulation of the constitutive model [Luciani et al. 1995; Miller and Pearce 1989], approximations of static friction through irregularly shaped grains [Bell et al. 2005], resolution up-sampling [Alduán et al. 2009], extensions to position-based simulations [Macklin et al. 2014], and interactions with fluids [Rungjiratananon et al. 2008].

Contact dynamics (CD) methods treat interactions with "hard" constraints [Acary and Brogliato 2008; Baraff 1989; Brogliato 2012; Jean 1999; Stewart 2011]. Originating in the work of Moreau [1983; 1988], CD treats contact and friction through constrained formulations, typically equivalent to *mixed linear complementarity formulations* (MLCP) [Stewart 2000; Stewart and Trinkle 1996]. While CD avoids many drawbacks of penalty methods, finding optimal solutions to their MLCP formulation is computationally difficult [Kaufman et al. 2008]. Significant attention has been directed to the accurate numerical solution of these models [Alart and Curnier 1991; Bonnefon and Daviet 2011; Daviet et al. 2011; Duriez et al. 2006; Jean and Moreau 1992; Jourdan et al. 1998; Stewart 2001], to convex relaxations of CD [Anitescu and Hart 2004b; Mazhar et al. 2015; Preclik 2014], to numerical methods suited for interactive simulations [Erleben 2007; Tonge et al. 2012], and to non-zero restitution [Smith et al. 2012; Uchida et al. 2015].

Continuum granular models. Our work builds on a rich history in the study of granular materials, the foundations of which were laid by Coulomb who first posited a relationship between the imposed pressure and the resistance to shear motion through a coefficient of friction. Modern variants include critical-state [Schofield and Wroth 1968] and anisotropic models [Dafalias et al. 2004; Rothenburg and Bathurst 1989]. The rate-sensitive inertial rheology [Groupement de Recherche Milieux Divisés (GDR MiDi) 2004; Jop et al. 2006] has proven effective at modeling fast flows, while non-local models [Aranson and Tsimring 2002; Kamrin and Koval 2012; Mohan et al. 2002] account for finite grain sizes by introducing non-local terms to the continuum description.

Continuum simulation methods. Within the scientific and engineering communities, a variety of simulation techniques for granular materials have been proposed, typically targeted towards specific flow regimes. For quasi-static and small-strain simulations, standard finite element analysis techniques are applicable [Zienkiewicz and Taylor 2000]. Finite volume discretizations have been successfully applied to faster flow regimes [Lagrée et al. 2011]. The material point method (MPM) [Sulsky et al. 1994] has been extended to the

simulation of granular materials [Więckowski 2004], with Dunatunga and Kamrin [Dunatunga and Kamrin 2015] proposing recent extensions to more accurately model the dilute phase.

Within graphics the simulation of granular materials with hybrid Eulerian-Lagrangian discretizations has received significant attention. Zhu et al. [2005] augmented an incompressible fluid-implicit particle (FLIP) solver with a Mohr-Coulomb yield criterion to identify rigid and flowing regions. Narain et al. [2010] draw an analogy to a discrete contact dynamics solver, and by mirroring the discrete solver's structure obtain unilaterally incompressible granular simulations with a faceted frictional yield criterion. Daviet et al. [2016] remedy artifacts in Narain's approach by employing a smooth yield criterion and support more general models like the $\mu(I)$ rheology, while Klár et al. [2016] augment an elastoplastic MPM simulation with a Drucker-Prager yield criterion featuring an angular momentum conserving APIC transfer [Jiang et al. 2015]. More recently, multi-physics techniques to simulate mixtures of sand and water [Gao et al. 2018; Tampubolon et al. 2017] and sand and rigid bodies [Hu et al. 2018] with MPM have been proposed.

In concert with the developments in hybrid Eulerian-Lagrangian discretizations, the graphics community has pushed work on fully Lagrangian discretizations forward, beginning with incompressible smooth-particle hydrodynamics (SPH) techniques [Lenaerts and Philip 2009]. Subsequent work has explored unilateral incompressibility in the context of SPH [Alduán and Otaduy 2011]. Recent works have continued to incorporate general advancements in SPH, with impressive visual results obtained through force-aware upsampling [Ihmsen et al. 2013].

The graphics community has also explored the specialized simulation of granular *surfaces* [Chanclou et al. 1996; Li and Moshell 1993; Onoue and Nishita 2003; Pla-Castells et al. 2008; Sumner et al. 1999], achieving fast simulations for restricted modes of deformation.

Coupled fluid simulations. Previous works couple complementary fluid simulation methods. Closest to our work, Golas et al. [2012] simulate interior regions with a Lagrangian vorticity method and treat boundary interactions with an Eulerian technique, and couple these domains using a Schwarz method [Toselli and Widlund 2006]. Other works simulate free surfaces using narrow bands of FLIP particles and treat the interior using an Eulerian simulation [Ferstl et al. 2016], couple SPH, shallow water, and Eulerian treatments within a single framework [Chentanez et al. 2015], and couple SPH simulations to a particle level set method [Wang et al. 2013].

Hybrid granular simulations. *Quasi-Continuum* and *Arlequin-type* methods have been explored primarily for crystalline solids, to expedite otherwise lengthy atomistic simulations by hybridizing with a crystal plasticity continuum model in zones where atomistic refinement is not needed [Dhia 1998; Shimokawa et al. 2007; Smith et al. 2001; Tadmor et al. 1996; Zhang and Ge 2005]. The idea of hybridizing discrete-particle and continuum approaches to simulate granular media is in its infancy, with only initial work done to show the validity of communicating mechanics between discrete grains and finite-element facets [Yan et al. 2010]. Recent work has explored when continuum and discrete treatments are simultaneously accurate [Kamrin 2010; Kamrin and Koval 2014; Rycroft et al. 2009], including an Arlequin-type method that couples statically-defined regions of a discrete element (DEM) simulation to the interior of a continuum FEM-based simulation to enrich stress fields around drill tips, for instance [Wellmann and Wriggers 2012]. We build on and extend these ideas to target regimes in which enriched degrees of freedom are required at surfaces, and where the boundary between continuum and discrete regimes evolves dynamically.

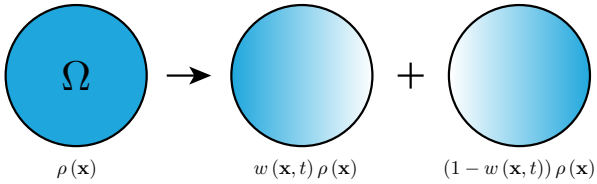


Fig. 2. **Blurred Density**: (Left) The reference domain Ω of an object with density $\rho(\mathbf{x})$. Mass density is colored in blue. (Right) A partition of unity of the density mediated by a weight function $w(\mathbf{x}, t)$.

In the granular physics and graphics literature, lower-level ideas have been tried where instead of implementing a general continuum model, the user imposes kinematic constraints to the particle motion in certain regions, often chosen based on experience with the problem at hand. The graphics literature has explored freezing rigid bodies that are sufficiently stationary [Smith et al. 2005]. Similar techniques have been proposed to accelerate the generation of granular packings for industrial applications [Mio et al. 2009]. In common granular setups such as rotating tumblers and growing sand piles, semi-empirical models can be used to guess zones of rigid material, and grains in these zones can be removed from the discrete update [Bouchaud et al. 1994; Hsu and Keyser 2010; McCarthy and Ottino 1998; Zhu and Yang 2010]. Holladay et al. [2012] carve out interior regions of granular materials moving at constant velocities and replace these groups of grains with meshes, but this method does not homogenize over rigidly rotating regions or over shear flows, and as the paper notes, can lead to volume loss. These ideas have been developed further in follow-up work [Holladay 2013; Munns 2015]. These methods make no claims as to the accuracy of the techniques for science and engineering applications, and have not yet demonstrated stable granular flows.

3 HYBRID KINEMATICS AND DYNAMICS

Our ultimate goal is to couple two consistent discretizations of a granular medium that are tailored to different scales. We begin by deriving a general method to spatially partition the equations of motion of a mechanical system into coupled, overlapping domains. With a simple constraint, the combined evolution of these systems exactly reproduces the behavior of the original, unpartitioned system. With this formalism in place, we will apply separate discretizations to the two partitioned domains, corresponding to a discrete and a continuum treatment. Finally, we propose a method to identify regions that require a discrete treatment, and a method to dynamically transition between discrete and continuum treatments as a simulation evolves.

3.1 Hybridization Through Constraints

Suppose that we view a single system as if it were two separate systems. Let the mass density $\rho(\mathbf{x})$ be defined over the reference coordinates $\mathbf{x} \in \Omega$ of a body (Fig. 2). We partition the density with a space-time dependent weight function $w(\mathbf{x}, t) \in [0, 1]$, ensuring that we recover the original density:

$$\rho(\mathbf{x}) = w(\mathbf{x}, t)\,\rho(\mathbf{x}) + (1 - w(\mathbf{x}, t))\,\rho(\mathbf{x}) \ . \tag{1}$$

If we treat the new partitions as separate systems with generalized coordinates $(\mathbf{q}_1, \mathbf{v}_1)$ and $(\mathbf{q}_2, \mathbf{v}_2)$, we can recover the kinematic description of the original system by requiring that $\mathbf{q}_1(\mathbf{x}, t) = \mathbf{q}_2(\mathbf{x}, t)$.

With identical initial configurations $\mathbf{q}_1(x, t_0) = \mathbf{q}_2(x, t_0)$, we can equivalently enforce equal velocities via the constraint $\mathbf{c}(\mathbf{x}, t) = \mathbf{v}_1(\mathbf{x}, t) - \mathbf{v}_2(\mathbf{x}, t) = \mathbf{0}$. Consistent with Fig. 3, our ultimate goal is to treat the discrete particles and the continuum as the two subsystems with a *reconciliation* zone wherever $0 < w < 1$.

We derive our governing equations for the hybrid coupling from Hamilton's Variational Principle [Lanczos 2012]; for dissipative systems, the analogous derivation follows from the Lagrange d'Alembert Principle, but the end result for our purposes is the same. This is a didactic derivation, and we will later apply these coupling forces to systems with friction, which still satisfies force/momentum balance.

Suppose kinetic T and potential U energies are given by

$$\begin{aligned} T &= \frac{1}{2}\int_\Omega \rho w \boldsymbol{v}_1^T \boldsymbol{v}_1 dV + \frac{1}{2}\int_\Omega \rho(1-w)\boldsymbol{v}_2^T \boldsymbol{v}_2 dV, \\ U &= \int_\Omega \rho w e[\boldsymbol{q}_1] dV + \int_\Omega \rho(1-w) e[\boldsymbol{q}_2] dV, \end{aligned} \tag{2}$$

for potential energy per mass e. Here and henceforth, we omit explicit parameters $(\mathbf{x}, t)$ when the dependence is clear. In addition, we couple the two systems via the augmented constraint $C = \int_\Omega \boldsymbol{\lambda}^T(\boldsymbol{v}_1 - \boldsymbol{v}_2)\,dV$, where $\lambda(\mathbf{x}, t)$ is a Lagrange multiplier field.

The Lagrangian $L = T - U + C$ is then incorporated into the action functional $\int_t L\,dt$, and the calculus of variations yields the Euler-Lagrange equations

$$\begin{aligned} \int_\Omega w\rho\boldsymbol{a}_1\,dV &= -\int_\Omega w\,\underbrace{\rho\frac{\delta e}{\delta \boldsymbol{q}_1}}_{\frac{\text{Force}}{\text{Volume 1}}}\,dV - \underbrace{\int_\Omega \boldsymbol{\lambda}\,dV}_{\text{coupling force}}, \\ \int_\Omega (1-w)\rho\boldsymbol{a}_2\,dV &= -\int_\Omega (1-w)\,\underbrace{\rho\frac{\delta e}{\delta \boldsymbol{q}_2}}_{\frac{\text{Force}}{\text{Volume 2}}}\,dV + \underbrace{\int_\Omega \boldsymbol{\lambda}\,dV}_{\text{coupling force}}, \end{aligned} \tag{3}$$

which are subject to the coupling constraint $\boldsymbol{v}_1 = \boldsymbol{v}_2$. The coupling force, which acts equally and oppositely on the two systems to enforce the constraint, arises naturally from the calculus of variations, averting the formulation of ad-hoc communication models between the two systems.

If we sum the two equations and substitute in the coupling constraint, we recover the original equations of motion for the entire simulation domain. The weight function naturally defines a partition of unity for the masses and the energies, with smaller weight values corresponding to a system having less influence in a given region. Outside the reconciliation zone, (3) is simply the (usual) equations of motion for two independent systems.

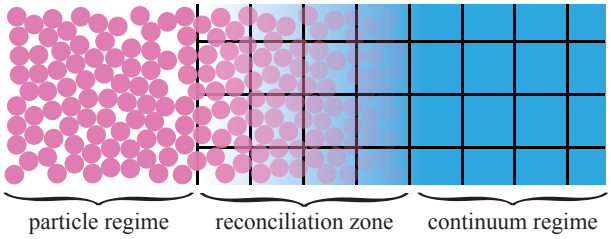


Fig. 3. **Hybrid Reconciliation Zone as a Blurred Transition**: A density partition applied to a partially discrete (left), partially continuum (right) system. Note that for simplicity, in the current study we set the densities to fixed values of 1/2 for each system throughout the reconciliation zone.

The Lagrange multiplier term operates as an external (constraint) force to ensure that the velocities of each system are equal. Under operator splitting, the above equations can be interpreted as first having two sets of (usual) decoupled equations of motion

$$\begin{aligned} \rho\boldsymbol{a}_1 &= -\rho\frac{\partial e}{\partial \boldsymbol{q}_1}, \\ \rho\boldsymbol{a}_2 &= -\rho\frac{\partial e}{\partial \boldsymbol{q}_2}, \end{aligned} \tag{4}$$

that are scaled by the weights, with a subsequent correction from the equal and opposite constraint force to guarantee equal velocities.

We now replace these two abstract systems with a discrete particle system and a continuum system (Fig. 3). By ansatz, we require the stress in the continuum domain to be compatible with the homogenized frictional forces in the discrete domain. This homogenization is realizable through the so-called Christoffersen formula [Christoffersen et al. 1981], which relates the continuum stress to the discrete frictional contact forces via:

$$\sigma_{ij} = \frac{1}{V}\sum_{\alpha \in contacts}^{N} \frac{1}{2}(f_i^\alpha d_j^\alpha + f_j^\alpha d_i^\alpha), \tag{5}$$

where σ_{ij} is the (i,j) component of the stress tensor, V is the volume about which one is homogenizing the stress, N is the number of contacts in that volume, f_i^α is the i^{th} component of the contact force vector at the α^{th} contact, and d_i^α is the i^{th} component of the vector connecting the centroids of the two grains in contact.

We employ a penalty formulation for the discrete frictional contact forces, while we model the continuum as an elastoplastic material with a Drucker-Prager yield criterion. We discretize and simulate the continuum with the material point method.

3.2 Discrete Particle Simulation

Consider a granular medium in which each grain is modeled as a rigid body. In 2D, each grain's configuration is parameterized by three degrees of freedom: two for the center of mass, and one for the orientation. All 2D grains are circular in geometry. In 3D, all grains are modeled as spheres and each grain's configuration is parameterized by six degrees of freedom, including three degrees of freedom for the body's center of mass $\boldsymbol{x}_k$ and three degrees of freedom for the body's orientation $\boldsymbol{R}_k$. Given K discrete grains, we

concatenate all degree of freedom into a single generalized configuration vector $\boldsymbol{q}_d \in \mathbb{R}^{3K}$ in 2D, and $\boldsymbol{q}_d \in \mathbb{R}^{6K}$ 3D. We can similarly build a generalized velocity vector $\boldsymbol{v}_d \in \mathbb{R}^{3K}$ and a generalized mass matrix $M_d \in \mathbb{R}^{3K\times 3K}$ in 2D, and $\boldsymbol{v}_d \in \mathbb{R}^{6K}$ and $M_d \in \mathbb{R}^{6K\times 6K}$ in 3D. The evolution of the system is now compactly described by

$$\begin{aligned} M_d \boldsymbol{a}_d &= \boldsymbol{f}_d(\boldsymbol{q}, \boldsymbol{v}_d, t), \\ \dot{\boldsymbol{q}}_d &= \boldsymbol{\omega}(\boldsymbol{q})\, \boldsymbol{v}_d, \end{aligned} \tag{6}$$

where $\boldsymbol{a}_d$ is the generalized acceleration, $\boldsymbol{f}_d$ contains all forces acting on the system. In 3D the force includes the so-called "quadratic velocity" term, and $\boldsymbol{\omega}(\boldsymbol{q})$ is the linear map from infinitesimal velocity changes to infinitesimal configuration changes. Note that the later two quantities depend on the choice of coordinates used to encode each body's orientation, with standard formulas available in the literature [Shabana 2013].

To model contacts between bodies, we employ a discrete element force-based penalty model [Cundall and Strack 1979]. This model is well studied, is able to model stable piles, frictional size-dependent jamming (clogging), and stick-slip behaviors, and has extensive validation against experimental results. Of importance to our needs, the Cundall and Strack penalty model has been shown to agree with continuum predictions in the regimes where we seek to homogenize the discrete dynamics [Rycroft et al. 2009]. The penalty force computation is divided into two components, including a normal force to resolve inter-grain penetration, and a tangential force to model friction. The normal and tangential forces are in turn related by a Coulomb friction constraint.

Concretely, the normal force is given by $\boldsymbol{f}_n = k_n d\boldsymbol{n} - \gamma_n \boldsymbol{v}_n$, where d is the penetration depth at the contact, $\boldsymbol{n}$ is the contact normal, k_n is the normal contact stiffness constant, $\boldsymbol{v}_n$ is the relative velocity projected into the normal direction, and γ_n is the normal damping coefficient. The tangential frictional force is given by $\boldsymbol{f}_t = k_t \Delta\boldsymbol{s} - \gamma_t \boldsymbol{v}_t$, where k_t is the tangential contact stiffness, $\Delta\boldsymbol{s}$ is the tangential anchor spring (described below), γ_t is the tangential damping coefficient, and $\boldsymbol{v}_t$ is the relative velocity projected into the tangent plane. At the end of each time step, we update $\Delta\boldsymbol{s}$ by integrating the tangential relative velocity at the contact. We then project any normal component out of $\Delta\boldsymbol{s}$, and rescale $\Delta\boldsymbol{s}$ so the Coulomb constraint $\boldsymbol{f}_t \le \mu \boldsymbol{f}_n$ is satisfied, where μ is the coefficient of Coulomb friction.

A Newtonian restitution model can be implemented in this framework by tuning the normal damping γ_n relative to the normal stiffness k_n. Given a desired coefficient of restitution (COR) e and a normal stiffness k_n, γ_n is determined by $\gamma_n = -2\ln e \sqrt{m k_n/(\pi^2 + \ln^2 e)}$, where m is the mean mass of a grain [Kamrin and Koval 2014]. Note that we run all simulations with a slight polydispersity in granular radii (and thus mass) to better match real shape distributions and to avoid crystallization. k_n itself is chosen to be as stiff as possible while still retaining a reasonable cost per time step, usually on the order of 10^{-6} seconds.

3.3 Continuum Modeling

We model continuum granular regions as an elastoplastic material with a Drucker-Prager plastic yield criterion. The evolution of the system is governed by the conservation of momentum

$$\rho \frac{D\boldsymbol{v}}{Dt} = \nabla \cdot \boldsymbol{\sigma} + \rho \boldsymbol{f}_{ext}, \tag{7}$$

and the conservation of mass

$$\frac{D\rho}{Dt} + \rho \nabla \cdot \boldsymbol{v} = 0, \tag{8}$$

where $\boldsymbol{\sigma}$ denotes the Cauchy stress tensor, $\frac{D_\cdot}{Dt}$ denotes the material derivative, and $\boldsymbol{f}_{ext}$ denotes any external body forces (e.g. gravity). We use a multiplicative decomposition of the deformation gradient $\boldsymbol{F} = \boldsymbol{F}^e \boldsymbol{F}^p$ where $\boldsymbol{F}^e$ and $\boldsymbol{F}^p$ denote the elastic and the plastic component of the deformation gradient, respectively. We denote the left Cauchy-Green strain by $\boldsymbol{b} = \boldsymbol{F}\boldsymbol{F}^T$.

Under small strains, the material behaves elastically. We employ a simple strain energy density to model the elastic behavior. In 2D the strain energy and Kirchhoff stress are given by

$$W = \frac{\kappa}{2}\left[\frac{1}{2}(J^2 - 1) - \ln J\right] + \frac{1}{2}\mu(Tr[\bar{\boldsymbol{b}}^e] - 2), \tag{9}$$

$$\tau = \frac{\kappa}{2}\left(J^2 - 1\right)\boldsymbol{I} + \mu\, \mathrm{dev}[\bar{\boldsymbol{b}}^e], \tag{10}$$

where $\bar{\boldsymbol{b}}^e = \det(\boldsymbol{b}^e)^{-1/2}\boldsymbol{b}^e$ is the volume preserving elastic left Cauchy-Green strain and $J = \det(\boldsymbol{F})$. In 3D, the strain energy density becomes $W = \frac{1}{2}\kappa\left[\frac{1}{2}(J^2 - 1) - \ln J\right] + \frac{1}{2}\mu(Tr[\bar{\boldsymbol{b}}^e] - 3)$ while $\bar{\boldsymbol{b}}^e = \det(\boldsymbol{b}^e)^{-1/3}\boldsymbol{b}^e$. κ and μ are the bulk and shear moduli respectively of the material. κ is chosen such that it is compatible with the DEM stiffness according to §3.3.2 of [Kamrin 2008], and μ is chosen to produce a Poisson's ratio of 0.3.

To allow the granular medium to separate, we consider the medium to be one-sided in the following sense: the granular medium can resist compression, but not extension. We model this effect with a free-flow mode, similar to Dunatunga et al. [2015]. When we detect that the material is in extension, indicated by $\det[\boldsymbol{b}^e] > 1$, we project the strain to $\det[\boldsymbol{b}^e] = 1$ by assigning $\boldsymbol{b}^e \leftarrow \det(\boldsymbol{b}^e)^{-1/2}\boldsymbol{b}^e$ ($\boldsymbol{b}^e \leftarrow \det(\boldsymbol{b}^e)^{-1/3}\boldsymbol{b}^e$ in 3D). The net result is zero pressure when the material is under extension.

We model plastic flow with the Drucker-Prager yield condition

$$\Phi = s - \alpha p \le 0, \tag{11}$$

where $s = \|\mathrm{dev}[\tau]\|_F$ is the magnitude of the shear stress (recall that $\mathrm{dev}[\boldsymbol{X}] = \boldsymbol{X} - \frac{1}{2}\mathrm{Tr}[\boldsymbol{X}]\boldsymbol{I}$), $p = -\frac{1}{2}\mathrm{Tr}[\tau]$ is the pressure ($\mathrm{dev}[\boldsymbol{X}] = \boldsymbol{X} - \frac{1}{3}\mathrm{Tr}[\boldsymbol{X}]\boldsymbol{I}$ and $p = -\frac{1}{3}\mathrm{Tr}[\tau]$ in 3D), and α controls the angle of repose. It often proves useful to express the yield in terms of strain, in which case we find that $\Phi = \mu\|\mathrm{dev}[\bar{\boldsymbol{b}}^e]\|_F + \alpha\frac{\kappa}{2}\left(J^2 - 1\right)$. The Drucker-Prager yield criterion corresponds intuitively with the notion of Coulomb friction in the discrete setting; the shear stress (continuum analog of force in the tangential plane) is capped at the magnitude of the pressure (continuum analog of the normal force) scaled by a constant α (continuum analog of μ). When $\Phi \le 0$, the material behaves elastically. When $\Phi > 0$, however, the material sustains plastic shape change. Crucially, a Drucker-Prager treatment of yield allows us to tune the angle of repose of a pile to match that of a discrete element simulation.

We model the granular medium as perfectly plastic: all excess yield is immediately converted into plastic deformation. We discretize the elastoplastic update rule for the strain $\dot{\boldsymbol{b}}^e = \nabla\boldsymbol{v}\boldsymbol{b}^e +$

$\boldsymbol{b}^e\nabla\boldsymbol{v}^T + \mathcal{L}_{\boldsymbol{v}}\boldsymbol{b}^e$ with the return mapping method [Simo and Hughes 1998]. The result is a predictor-corrector style update for the strain. We first update $\boldsymbol{b}^e$ through $\dot{\boldsymbol{b}}^e = \nabla\boldsymbol{v}\boldsymbol{b}^e + \boldsymbol{b}^e\nabla\boldsymbol{v}^T$, ignoring any plastic flow. This predicted strain can now violate the yield condition. To remove yield excess from $\boldsymbol{b}^{e,*}$, where $*$ denotes some violating state, we impose two constraints on the plastic flow to project the material to a violation-free state. First, the plastic granular flow should conserve volume, and the projected strain $\det[\boldsymbol{b}^{e,p}]$ should satisfy $\det[\boldsymbol{b}^{e,p}] = \det[\boldsymbol{b}^{e,*}]$. Second, to satisfy the yield condition, we seek $\Phi\left(\tau\left(\boldsymbol{b}^{e,p}\right)\right) = 0$. We consider the flow to be in the direction of the shear and we decompose the projected strain as $\boldsymbol{b}^{e,p} = \lambda_1\boldsymbol{I} + \lambda_2\,\mathrm{dev}[\det[\boldsymbol{b}^{e,*}]]$.

Expanding the constraint $\Phi\left(\tau\left(\boldsymbol{b}^{e,p}\right)\right) = 0$, we find:

$$\begin{aligned}\Phi\left(\tau\left(\boldsymbol{b}^{e,p}\right)\right) &= \mu\|\,\mathrm{dev}[\boldsymbol{b}^{e,p}]\|_F + \alpha\frac{\kappa}{2}(J^2-1)\\ &= \mu\lambda_2\|\,\mathrm{dev}[\boldsymbol{b}^{e,*}]\|_F + \alpha\frac{\kappa}{2}(J^2-1).\end{aligned} \tag{12}$$

Equating to 0 and solving for λ_2, we conclude that:

$$\lambda_2 = -\left(\alpha\frac{\kappa}{2}(J^2-1)\right)/\left(\mu\|\,\mathrm{dev}[\boldsymbol{b}^{e,*}]\|_F\right). \tag{13}$$

Similarly, if we expand $\det[\boldsymbol{b}^{e,p}]$ we find

$$\det[\boldsymbol{b}^{e,p}] = \det[\lambda_1\boldsymbol{I} + \lambda_2\,\mathrm{dev}[\boldsymbol{b}^{e,*}]]. \tag{14}$$

In 2D, recalling that $\det[\boldsymbol{I}+\boldsymbol{A}] = 1 + \det[\boldsymbol{A}] + \mathrm{Tr}[\boldsymbol{A}]$, we have $\det[\boldsymbol{I} + \mathrm{dev}[\boldsymbol{A}]] = 1 + \det[\mathrm{dev}[\boldsymbol{A}]]$. Then, we arrive at

$$\begin{aligned}\det[\boldsymbol{b}^{e,p}] &= \lambda_1^2\det[\boldsymbol{I} + \frac{\lambda_2}{\lambda_1}\mathrm{dev}[\boldsymbol{b}^{e,*}]]\\ &= \lambda_1^2\left(1 + \frac{\lambda_2^2}{\lambda_1^2}\det[\mathrm{dev}[\boldsymbol{b}^{e,*}]]\right)\\ &= \lambda_1^2 + \lambda_2^2\det[\mathrm{dev}[\boldsymbol{b}^{e,*}]].\end{aligned} \tag{15}$$

Equating to $\det[\boldsymbol{b}^{e,*}]$ and solving for λ_1 we find:

$$\lambda_1 = \sqrt{\det[\boldsymbol{b}^{e,*}] - \lambda_2^2\det[\mathrm{dev}[\boldsymbol{b}^{e,*}]]}. \tag{16}$$

In 3D, recalling that $\det[\boldsymbol{I}+\boldsymbol{A}] = 1 + \det[\boldsymbol{A}] + \mathrm{Tr}[\boldsymbol{A}] + \frac{1}{2}\mathrm{Tr}[\boldsymbol{A}]^2 - \frac{1}{2}\mathrm{Tr}[\boldsymbol{A}^2]$, we have $\det[\boldsymbol{I}+\mathrm{dev}[\boldsymbol{A}]] = 1 + \det[\mathrm{dev}[\boldsymbol{A}]] - \frac{1}{2}\|\,\mathrm{dev}[\boldsymbol{A}]\|_F^2$. Then, we arrive at

$$\lambda_1^3 + \lambda_2^3\det[\mathrm{dev}[\boldsymbol{b}^{e,*}]] - \frac{\lambda_1\lambda_2^2}{2}\|\,\mathrm{dev}[\boldsymbol{b}^{e,*}]\|_F^2 - \det[\boldsymbol{b}^{e,*}] = 0. \tag{17}$$

We employ Cardano's method to analytically solve (17) and obtain λ_1. With λ_1 and λ_2 in hand, we are able to easily project $\boldsymbol{b}^{e,*}$ and enable the plastic flow. See Algorithm 12 in the supplemental material for a compact description of the overall projection procedure.

3.4 MPM Discretization

We simulate the plastic continuum medium using the material point method [Sulsky et al. 1994]. Material properties are advected in a Lagrangian fashion with a spatial sampling of material points, while forces are computed on an Eulerian background grid.

We use a standard, explicit MPM integrator with a return-mapping style treatment of plasticity [Yue et al. 2015]. While similar works advocate for cubic-spline based nodal shape functions [Steffen et al. 2008; Stomakhin et al. 2013], we found these to be expensive to compute in practice. As linear hat functions are known to cause instabilities when material points cross grid boundaries [Jiang et al. 2016], we instead employ a version of the Generalized Interpolation Material Point Method (GIMP) [Bardenhagen and Kober 2004; Gao et al. 2017], which, conceptually, assigns a finite width to material points to smooth the transition of the shape functions across grid boundaries. In all of our tests, we set the GIMP width to the cell width of the MPM grid divided by the number of MPM points per cell dimension (i.e. a 3D simulation with a grid width of 1 with 2 points per cell dimension would have 8 total points per element and each point would have a width of 0.5). We found GIMP to be both stable and efficient. As we treat all interactions with boundaries through a discrete particle model, no special consideration is needed for boundary conditions within our material point formulation.

We summarize our MPM implementation in the supplemental material. Note that we divide the MPM integrator into two phases for reasons that we discuss in Section 3.6. Further note that in our MPM implementation, we use a parameter β to interpolate between PIC and FLIP velocity updates. To better conserve angular momentum we desire a high degree of FLIP, and thus set β to a value between 0.95 and 1.0 for all simulations.

3.5 Hybridization

Having accounted for the velocity (and possibly position) update from the equations of motion, we can interpret each system in terms of forces-per-volume and then, within a finite volume element, correct the velocities and positions to enforce coupling via (3) subject to the constraint C. This allows us to interpret the constraint $\boldsymbol{v}_1 = \boldsymbol{v}_2$ in an average or homogenized sense [Bergou et al. 2007]: in the reconciliation zone, the velocity of every discrete particle is forced to agree with the interpolated velocity of the continuum. Let λ_k represent the constraint force on the kth discrete particle. Given the reconciliation zone, Ω_R, the pth material point moves as

$$\begin{aligned}&\frac{d}{dt}\boldsymbol{q}_p = \boldsymbol{v}_p,\\ &\frac{d}{dt}(w_pM_p\boldsymbol{v}_p) = \underbrace{\frac{d}{dt}(w_pM_p\boldsymbol{v}_p^*)}_{\text{unconstrained step}} \underbrace{-\sum_{k\in\Omega_R}\Gamma_{pk}\lambda_k}_{\text{constrained step}},\end{aligned} \tag{18}$$

while the kth discrete particle moves as

$$\begin{aligned}&\frac{d}{dt}\boldsymbol{q}_k = \boldsymbol{v}_k,\\ &\frac{d}{dt}((1-w_k)M_k\boldsymbol{v}_k) = \underbrace{\frac{d}{dt}((1-w_k)M_k\boldsymbol{v}_k^*)}_{\text{unconstrained step}} \underbrace{+\boldsymbol{\lambda}_k}_{\text{constrained step}},\end{aligned} \tag{19}$$

where $\boldsymbol{v}_p^*$ (v_k^*) are the predictions from continuum (discrete) simulation before coupling forces are added, and Γ_{pk} are material-point to discrete-particle interpolation coefficients.

3.6 Hybrid Coupling Discretization

We now present the discretized, hybrid coupling algorithm that follows from the previous discussion. Note that while general, implicit equations of motion would require a non-linear Newton solve, with explicit equations of motion, the coupled solve can be reduced to a single linear solve with a predictor-corrector structure. See

Appendix A for a full derivation. We further observe that MPM forces are defined on a background Eulerian grid, which provides a natural scratch-pad on which to compute the constraint forces. Concretely, a full hybrid time step begins by stepping the continuum and discrete systems in isolation, omitting the advection steps, to obtain the predicted momenta. In our implementation, we compute a predictor discrete step by taking a full step of the discrete system and discarding the position update. For the continuum system, we split the full MPM step into two stages, where the predictor first stage steps the system to just the instant that material point properties are rasterized to the gird. In the DEM force computation, we weight each discrete spring force by $1 - w$, evaluated at the corresponding contact point, and each body mass by $1 - w$, evaluated at the center of mass. Likewise, in the MPM force computation, we weight the stress and mass of each material point by w, evaluated at the point location. We next assemble and solve a linear system to constrain the velocities of the discrete and continuum simulations in the reconciliation zone:

$$\begin{bmatrix} W_c M_c & 0 & \Gamma_c \\ 0 & W_d M_d & -\Gamma_d \\ \Gamma_c^T & -\Gamma_d^T & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_c^{n+1} \\ \boldsymbol{v}_d^{n+1} \\ \lambda \end{bmatrix} = \begin{bmatrix} W_c \left(M_c \boldsymbol{v}_c^n + h \boldsymbol{f}_c^n \right) \\ W_d \left(M_d \boldsymbol{v}_d^n + h \boldsymbol{f}_d^n \right) \\ 0 \end{bmatrix}, \quad (20)$$

where W_c and W_d are diagonal matrices that contain the mass weights for the continuum and discrete systems, M_c and M_d are the mass matrices, $\boldsymbol{v}_c$ and $\boldsymbol{v}_d$ are the velocities, and $\boldsymbol{f}_c$ and $\boldsymbol{f}_d$ are explicit forces from the continuum and discrete systems. $M_c \boldsymbol{v}_c^n + h \boldsymbol{f}_c^n$ and $M_d \boldsymbol{v}_d^n + h \boldsymbol{f}_d^n$ are the predicted momenta of the continuum and discrete systems, respectively, and solving the linear system gives the corrected continuum and discrete velocities $\boldsymbol{v}_c^{n+1}$ and $\boldsymbol{v}_d^{n+1}$. Γ_d and Γ_c are defined such that $\Gamma_d^T \boldsymbol{v}_d - \Gamma_c^T \boldsymbol{v}_c$ produces the residual relative velocity of discrete bodies within the background velocity field defined by the material point grid. Under this definition, Γ_d reduces to the identity matrix, while each column of Γ_c contains the weights that recover each discrete body's center of mass from the MPM grid's basis functions. We compute Γ_c using the positions of the discrete bodies and material points at the beginning of the step. Note that we can restrict the size of this system to only the degrees of freedom in the reconciliation zone. After solving the linear system, we update the discrete velocities and advect the discrete bodies along the corrected velocities, concluding the discrete step, and we compute the latter stage of the material point step using the new, constrained velocities, concluding the continuum step. This concludes a full hybrid time step, the details of which are summarized in Alg. 1.

3.7 Inexpensive Coupling via Grid vs. Grid Hybridization

While this method to compute coupling forces indeed works, a further speedup is possible by defining a second background grid that is co-located with the background MPM grid. The velocities of the discrete bodies can be rasterized to this second grid as if they were material points. The constraint matrices Γ_d and Γ_c now reduce to the identity, and the system in (20) can be solved in closed form (31) as shown in Appendix B. Crucially, the solution at each grid node is independent of the other grid nodes, and takes the simple form of an inelastic collision between two particles with masses and velocities equal to those of the grid node (Alg. 3). This method to compute hybridization forces is simple, robust, and trivially parallelized. After solving this system, the discrete grid-based velocities are mapped back to the discrete bodies in the same manner as MPM points. We always employ this grid vs. grid hybridization technique, and never directly solve the linear system (20).

3.8 Learning from Early Attempts

Before we settled on a penalty-based method [Cundall and Strack 1979] for our discrete simulation, we tested a hard-particle, impulse-based approach [Kaufman et al. 2008]. Like our current version, we made use of operator splitting and performed prediction-correction style integration; we performed the impulse based and continuum integration separately, and then corrected the velocities and positions with a coupling step. We found that this approach led to overlap artifacts, however, where the discrete bodies in the reconciliation zone slowly drifted into the corresponding continuum, as the impulse-based approach was unable to fix the overlap between grains caused by the correction step. This is due to the fact that the corrector step may apply forces to the grains that cause motions violating the hard-particle constraint. We tried to use spring-based penalties and post-stabilization [Anitescu and Hart 2004a; Cline and Pai 2003] to correct this overlap over time, but the addition of the springs led to smaller time steps, eliminating the primary advantage of the impulse-based model. The penalty-based discrete element method we use now, on the other hand, can naturally fix the overlaps caused by the coupling step and is a more physical model for grains of known elasticity. Indeed, we find that our choice of the penalty-based method together with the material point method and the coupling procedure produces simulations without any observable drift between domains in the hybrid zone.

3.9 Substepping the Discrete Simulation

The maximal stable time step of standalone DEM, which is decided by the grain sizes, is smaller than that of standalone MPM, which is decided by the background cell width. In the hybrid system, we therefore seek to exploit MPM's ability to take larger time steps than DEM by substepping the DEM simulation with respect to MPM, i.e. by carrying out a number of DEM substeps for each MPM step. In all examples we ran DEM at its largest stable time step, while we were able to stably step MPM at ten times DEM's time step when coupled, i.e. ten DEM substeps for each MPM step. We observe instabilities if we push the number of substeps beyond this threshold. We enforce the hybrid coupling constraint during the MPM step. In effect, MPM allows us to homogenize in both space and time. Details of hybrid system with substepping are listed in Alg. 2.

4 HYBRID ORACLE

With a coupling method in place, a full hybrid simulation requires three additional core features: an *oracle* to identify regions that can be safely approximated as a continuum, a homogenization operator that is able to convert a discrete region of material to a continuum region, and an enrichment operator that is able to synthesize discrete grains in a manner consistent with the continuum.

Algorithm 1 Hybrid_Step

```
Copy_Positions_Of_Discrete_Bodies
Discrete_Step
Restore_Discrete_Positions
MPM_Step_First_Phase                                  ▷ Alg. 20
Rasterize_Discrete_And_Continuum_Velocities           ▷ Alg. 4
Hybrid_Constraint_Solve                               ▷ Alg. 3
Transfer_Constrained_Velocities_To_Systems            ▷ Alg. 5
Update_Hybrid_Discrete_Positions                      ▷ Alg. 6
MPM_Step_Second_Phase                                 ▷ Alg. 21
Update_Hybrid_State                                   ▷ Alg. 9
```

Algorithm 2 Hybrid_Step_With_Substepping

```
Copy_Positions_Of_Discrete_Bodies
for i = 0 ... num_substeps do
    Discrete_Step
    Accumulate_Discrete_Momentum_Changes
end for
Restore_Discrete_Positions
MPM_Step_First_Phase                                  ▷ Alg. 20
Rasterize_Discrete_And_Continuum_Velocities           ▷ Alg. 4
Hybrid_Constraint_Solve                               ▷ Alg. 3
Transfer_Constrained_Velocities_To_Systems            ▷ Alg. 5
Update_Hybrid_Discrete_Positions                      ▷ Alg. 6
MPM_Step_Second_Phase                                 ▷ Alg. 21
Update_Hybrid_State                                   ▷ Alg. 9
```

Algorithm 3 Hybrid_Constraint_Solve

```
for i = 0 ... num_grid_nodes_in_reconciliation_zone do
    p_i ← w_{c,i} · m_{c,i} · v_{c,i} + w_{d,i} · m_{d,i} · v_{d,i}
    m_i ← w_{c,i} · m_{c,i} + w_{d,i} · m_{d,i}
    if m_i ≠ 0 then
        v_i ← p_i / m_i
    else
        v_i ← 0
    end if
end for
```

Algorithm 4 Rasterize_Discrete_And_Continuum_Velocities

```
for point ∈ Material_Points_in_Reconciliation_Zone do
    for node ∈ Stencil(point) do
        w ← Weight(point, node)
        node.m_c += w · point.m
        node.p_c += w · point.m · point.v
    end for
end for
for body ∈ Discrete_Grains_in_Reconciliation_Zone do
    for node ∈ Stencil(body) do
        w ← Weight(body, node)
        node.m_d += w · point.m
        node.p_d += w · body.m · body.v
    end for
end for
for node ∈ Grid_Nodes do
    if node.m_d > 0 then
        node.v_d ← node.p_d / node.m_d
    end if
    if node.m_c > 0 then
        node.v_c ← node.p_c / node.m_c
    end if
end for
```

Algorithm 5 Transfer_Constrained_Velocities_To_Systems

```
for point ∈ Material_Points_in_Reconciliation_Zone do
    point.v ← 0
    for node ∈ Stencil(point) do
        w ← Weight(point, node)
        point.v += w · node.v
    end for
end for
for body ∈ Discrete_Grains_in_Reconciliation_Zone do
    body.v ← 0
    for node ∈ Stencil(body) do
        w ← Weight(body, node)
        body.v += w · node.v
    end for
end for
```

Algorithm 6 Update_Hybrid_Discrete_Positions

```
for body ∈ Discrete_Grains_in_Reconciliation_Zone do
    body.p ← body.p_prev + dt · body.v
end for
```

4.1 A Continuum Oracle

Critical to our hybridization method is an oracle that is able to flag regions of the simulation domain as safe for a continuum treatment. Regions are unfit for a continuum treatment when one of any number of conditions are satisfied. First, in regions of low pressure, grains are more likely to separate from the material bulk and undergo ballistic motion. Second, high strain rate gradients suggest that the granular flow varies too rapidly to be safely represented as a homogenized continuum [Dijksman and van Hecke 2010; Kamrin 2010; Koval et al. 2009]. Finally, in thin flows, grain-level dynamics can dominate, leading to finite size effects (e.g., clogging) not captured by local continuum models [Beverloo et al. 1961; Groupement de Recherche Milieux Divisés (GDR MiDi) 2004; Pouliquen 1999; Sheldon and Durian 2010]. We have found that the packing fraction serves as an effective signal for these sources of fundamentally discrete behavior. The emphasis on packing fraction is further motivated by the fact that graphical fidelity should be retained on the exterior of granular systems of interest, as it is this low packing fraction exterior that is seen by an observer.

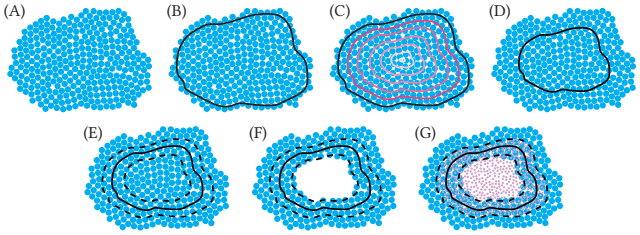


Fig. 4. Initialization of a hybrid simulation: (A) We begin with a collection of DEM grains. (B) We next locate a level set corresponding to a given low density, here denoted as a black line. (C) Across the domain, we compute the distance to the density threshold, indicated by lines in lighter shades of red as the distance increases. (D) We select a user-tunable distance to the density level-set that serves as the center of the hybrid "reconciliation" zone. We denote this critical distance as a solid black line. (E) We extend the hybrid zone along the distance field by a given half-width in each direction, indicated by dotted lines. This hybrid reconciliation zone between the dotted lines defines a zone where the DEM system will be coupled to the continuum system. We homogenize the velocity and stress (see §4.2) for use in step (G). (F) We delete all discrete grains that fall within the inner boundary of the reconciliation zone. (G) We run the "avoid-a-void" algorithm of Yue et al. [2015] from the outer boundary in to populate the region with material points. The material point states are determined using the homogenized velocity and stress computed in step (E) (see §4.2 for details).

Our oracle begins by computing the packing fraction of the discrete particle system on a uniform background grid. Note that only discrete grains are considered when computing the packing fraction, as continuum and hybrid regions are, by ansatz, considered sufficiently dense. From this implicit representation, we extract an isocontour corresponding to a critical, threshold packing fraction Φ_ρ (Fig. 4 (B)). We next compute the distance $\Phi_d(\boldsymbol{x})$ to this threshold isocontour on a second, uniform grid (C), and define an isocontour corresponding to a user specified distance ϕ_0 as the centerline of the reconciliation zone (D). We label a zone as hybrid if the distance from the centerline is within a given half-width r_h, i.e. $\phi_0 - r_h \leq \Phi_d(\boldsymbol{x}) \leq \phi_0 + r_h$. Similary, we label a zone as continuum if $\Phi_d(\boldsymbol{x}) > \phi_0 - r_h$ and a zone as discrete if $\Phi_d(\boldsymbol{x}) < \phi_0 + r_h$ (Alg. 7).

4.2 Homogenization and Enrichment

After updating the boundary between simulation domains, we are faced with four possible transition scenarios: a previously hybrid zone is now purely discrete, a previously hybrid zone is now purely continuum, a previously discrete zone is now hybrid, or a previously continuum zone is now hybrid. Note that after initialization, we do not permit direct transitions from continuum to discrete regions or vice versa. The transition away from a hybrid representation is quite simple. For a hybrid region transitioning to a purely discrete region, we simply delete all material points in the region. Similarly, for a hybrid region transitioning to a purely continuum region, we delete all discrete grains in the region. The transition to a hybrid representation is more involved, however. A region that previously contained only material points will require the insertion of discrete grains. Likewise, a previously discrete region will require the insertion of new material points. See Alg. 8.

The problem of adding samples to a dynamic simulation was addressed in the context of the material point method with the recently proposed avoid-a-void algorithm [Yue et al. 2015]. The avoid-a-void method applies Poisson disc sampling to maintain approximately constant material point distributions and to prevent the formation of non-physical voids within a simulated material. This technique is perfectly suited to our needs, where we need to insert material points or discrete grains in regions of high material density recently labeled as hybrid.

For discrete grains, the new position is determined by the Poisson disc sampling procedure, while we draw the radius of new grains from the same normal distribution used to generate the initial grain radii. The initial velocity of new discrete grains is computed by averaging the velocity of surrounding discrete grains and material points within a radius of 6 (mean) grain diameters, with an exponential falloff (Alg. 10). This window width is slightly above the minimal size that recovers continuum-like quantities in a "granular volume element" [Rycroft et al. 2009]. These nearest neighbor queries are accelerated with a uniform background grid.

We first determine the positions of new material points with Poisson disc sampling. Next, we compute the velocity, (normalized) strain, and deformation gradient magnitude of new points with *homogenization*. We utilize a regular grid to rasterize the velocity of discrete grains; after rasterizing each discrete grain's mass and momentum to the grid (using the same rasterization procedure as MPM), we divide the rasterized momentum by the rasterized mass to determine each grid node's velocity. Likewise, we estimate the *homogenized* stress at grid nodes using a modification of the Christoffersen formula [1981] that yields smooth stress fields via the MPM shape functions, as described in Alg. 11. After computing the homogenized velocities and stresses at nodes, we use the MPM shape functions to evaluate the velocity and stress at the newly generated material point positions. Finally, we convert the interpolated stress to strain: from the definition of the Kirchhoff stress, we can relate the Cauchy stress and the strain via $\boldsymbol{\sigma} = \frac{\kappa}{2}\frac{(J^2-1)}{J}\boldsymbol{I} + \frac{\mu}{J}\,\mathrm{dev}[\bar{\boldsymbol{b}}^e]$. By taking the trace of both sides, we obtain $\mathrm{Tr}[\boldsymbol{\sigma}] = \frac{3\kappa}{2}\frac{(J^2-1)}{J}$ (in 3D) and solve for J. Next, by taking the deviator of both sides, we obtain $\mathrm{dev}[\bar{\boldsymbol{b}}^e] = \frac{J}{\mu}\,\mathrm{dev}[\boldsymbol{\sigma}]$. Finally, $\bar{\boldsymbol{b}}^e$ is given in the form $\bar{\boldsymbol{b}}^e = \mathrm{dev}[\bar{\boldsymbol{b}}^e] + t\boldsymbol{I}$, and we solve for t with $det[\bar{\boldsymbol{b}}^e] = 1$.

We emphasize the importance of syncing the stress and strain during homogenization to preserve volume. Previously, we initialized new continuum elements to a stress-free state. This stress-free initialization caused the material to gain volume if the discrete grains were between a gaseous and liquid state while separating, however.

4.3 Layered Hybridization

We often found it beneficial to only use the discrete treatment for the topmost visible portion of a granular assembly. If the boundary treatment (at the side walls and floor) with the continuum elements is physically valid, and if the continuum elements are invisible from the outside, this *layered hybridization* approach yields additional speed improvements. We apply this approach to the bunny toss, excavator, bunny drill, and tire examples in Section 5.

Algorithm 7 `Identify_Hybrid_Zones`

$\Phi_\rho \leftarrow$ `Discrete_Packing_Fraction_Isocontours` $(\boldsymbol{q}_d)$
$\Phi_d \leftarrow$ `Distance_to_Density_Isocontours` (Φ_ρ)
`continuum_zone` $(\mathbf{x}) \leftarrow \Phi_d(\mathbf{x}) > \phi_0 - r_h$
`discrete_zone` $(\mathbf{x}) \leftarrow \Phi_d(\mathbf{x}) < \phi_0 + r_h$
`hybrid_zone` $(\mathbf{x}) \leftarrow \phi_0 - r_h \leq \Phi_d(\mathbf{x}) \leq \phi_0 + r_h$

Algorithm 8 `Update_Hybrid_Zones`

`Homogenize_Velocity_and_Stress(hybrid_zone)` ▷ Alg. 11
`Discrete_Avoid_a_Void(hybrid_zone)` ▷ Alg. 12
`Continuum_Avoid_a_Void(hybrid_zone)` ▷ Alg. 16
`Delete_Discrete_Grains(continuum_zone)` ▷ Alg. 17
`Delete_Continuum_Particles(discrete_zone)` ▷ Alg. 18

Algorithm 9 `Update_Hybrid_State`

`Identify_Hybrid_Zones`
`Update_Hybrid_Zones`

Algorithm 10 `Create_Discrete_Grain`

$\mathbf{x} \leftarrow$ `Position_from_Avoid_a_Void`
$r \leftarrow N(r_{mean}, r_{sigma})$ ▷ Normally distributed radii
$m \leftarrow \frac{4}{3}\pi r^3$
$\mathbf{v} \leftarrow 0$
$W \leftarrow 0$
for $i = 0 \ldots N_d$ **do** ▷ Iterate over discrete grains
  if $|\mathbf{x}_i - \mathbf{x}| < 12\, r_{mean}$ **then**
    $w \leftarrow e^{|\mathbf{x}_i - \mathbf{x}|^2 / 2 r_{mean}^2}$
    $\mathbf{v} \leftarrow \mathbf{v} + w\, \mathbf{v}_i$
    $W \leftarrow W + w$
  end if
end for
for $i = 0 \ldots N_c$ **do** ▷ Iterate over material points
  if $|\mathbf{x}_i - \mathbf{x}| < 12\, r_{mean}$ **then**
    $w \leftarrow e^{|\mathbf{x}_i - \mathbf{x}|^2 / 2 r_{mean}^2}$
    $\mathbf{v} \leftarrow \mathbf{v} + w\, \mathbf{v}_i$
    $W \leftarrow W + w$
  end if
end for
$\mathbf{v} \leftarrow \mathbf{v}/W$

Algorithm 11 `Homogenize_Velocity_and_Stress`

for each body ∈ `Discrete_Grains` **do**
  for node ∈ `Stencil(body)` **do**
    $w \leftarrow$ `Weight(body, node)`
    `node.m` += $w \cdot$ `body.m`
    `node.momentum` += $w \cdot$ `body.m` $\cdot$ `body.`$\boldsymbol{v}$
  end for
end for
for node ∈ `Grid_Nodes` **do**
  if `node.m` > 0 **then**
    `homogenized_velocity` $\leftarrow$ `node.momentum/node.m`
  end if
end for
for each c ∈ `collisions` **do**
  for node ∈ `Stencil(c)` **do**
    $w \leftarrow$ `Weight(c, node)`
    $\boldsymbol{f}_c \leftarrow$ `c.collision_force` ▷ Normal and friction forces
    $\boldsymbol{r}_c \leftarrow$ `c.arm_vector`
    $\boldsymbol{\sigma}_c \leftarrow \frac{1}{2}\left(\boldsymbol{f}_c \boldsymbol{r}_c^T + \boldsymbol{r}_c \boldsymbol{f}_c^T\right)$
    `homogenized_stress` += $w \cdot \boldsymbol{\sigma}_c$`/cell_volume`
  end for
end for

Algorithm 12 `Discrete_Avoid_a_Void`

for `cell` ∈ `Hybrid_Zone` **do**
  for $i = 0 \ldots$ `Max_Iters` **do**
    `Create_Discrete_Grain()` ▷ Alg. 10
  end for
end for

Algorithm 13 `Create_Continuum_Particle`

$\mathbf{x} \leftarrow$ `Position_from_Avoid_a_Void`

Algorithm 14 `Reassign_Hybrid_Continuum_Properties`

for `point` ∈ `Cell` **do**
  `point.m` $\leftarrow$ `mpm_mass_per_cell/(#points` ∈ `Cell)`
  `point.vol` $\leftarrow$ `cell_volume/(#points` ∈ `Cell)`
  `point.`$\boldsymbol{v}$ $\leftarrow$ `homogenized_velocity(point.`$\boldsymbol{x}$`)`
  $\boldsymbol{\sigma} \leftarrow$ `homogenized_stress(point.`$\boldsymbol{x}$`)`
  `point.strain` $\leftarrow$ `convert_stress_to_strain(`$\boldsymbol{\sigma}$`)`
end for

5 RESULTS

We now test our method against a number of model problems. In each of these examples, we employ uniform density partition weights of 1/2. Unless otherwise stated, most simulation runs have an MPM cell width to DEM mean grain diameter ratio of 2:1.

Granular column collapse. In Fig. 5, we simulate a collapsing column with pure DEM and with our hybrid approach. Note the correspondence between the shapes of both piles. Further observe that our hybrid method captures detailed "fly away" effects – individual grains separate from the overall bulk and roll away at the front of the collapse, a visually important effect that would be difficult to capture with a purely continuum model.

Encouraged by the agreement between the pure DEM approach and our hybrid approach, we validated our hybrid model against the power-law scaling of the run-out distance $\delta d = d_f - d_i$ reported in the literature, where d_f is the distance from the left wall (for a unilateral collapse like our study, or from the column center for a bilateral collapse) to the center of mass of the foremost grain that is connected to the main collection of grains, and d_i is the initial column width (for a unilateral collapse, or the initial half-width for

Algorithm 15 `Determine_New_Inner_Continuum_Properties`

```
for newly_sampled_point ∈ Cell do
    point.m ← gather_mass_of_extant_nghbr_pnts
    point.vol ← gather_vol_of_extant_nghbr_pnts
    point.v ← interp_vel_of_extant_nghbr_pnts
    point.strain ← interp_strn_of_extant_nghbr_pnts
end for
```

Algorithm 16 `Continuum_Avoid_a_Void`

```
for cell ∈ Hybrid_Zone do
    for i = 0...Max_Iters do
        Create_Continuum_Particle()                ▷ Alg. 13
    end for
    Reassign_Hybrid_Continuum_Properties           ▷ Alg. 14
end for
for cell ∈ Continuum_Zone do
    for i = 0...Max_Iters do
        Create_Continuum_Particle()                ▷ Alg. 13
    end for
    Determine_New_Inner_Continuum_Properties       ▷ Alg. 15
end for
```

Algorithm 17 `Delete_Discrete_Grains`

```
for body ∈ Discrete_Grains do
    if body ∈ Continuum_Zone then
        delete(body)
    end if
end for
```

Algorithm 18 `Delete_Continuum_Particles`

```
for point ∈ Material_Points do
    if point ∈ Discrete_Zone then
        delete(point)
    end if
end for
```

a bilateral one), as in Fig. 5. Granular run-out in a column follows a power law scaling as a function of the initial aspect ratio $AR = h_i/d_i$ in both experimental [Balmforth and Kerswell 2005; Lube et al. 2005] and numerical [Dunatunga and Kamrin 2015; Lagrée et al. 2011; Mast et al. 2015; Staron and Hinch 2005] tests, where h_i is the initial height of the column. Running a series of run-out simulations over a range of aspect ratios, we corroborate the previously reported power law scaling. Below a critical aspect ratio, we observe a linear run-out distance as a function of aspect ratio. Above this threshold, we observe a second power law scaling.

As evident in Fig. 6, a pure discrete simulation captures the expected runout profile. Encouragingly, our hybrid method captures a similar runout profile, with a clear turnover point. In the aforementioned experimental study by Lube, experiments showed that for AR < 1.8, the runout profile could be described by a simple linear relation: $\delta d/d_i = \alpha(AR)$. Lube found $\alpha = 1.2$ while our simulation data fits best with with $\alpha = 1.45$. In regimes with AR > 2.8, the runout distance was best described with a power law of the form $\delta d/d_i = \beta(AR)^\gamma$. Lube observed a best-fit with $\beta = 1.9$ and $\gamma = 0.67$.

Simulation Scene	DEM Cost	Hybrid Cost	Total Cost	Speed Boost	DEM Grains
Silo, DEM	0.41	N/A	0.41	N/A	100,000
Silo, Hybrid	0.25	0.03	0.28	1.47×	78,538
Toss, DEM	0.620	N/A	0.620	N/A	120,000
Toss, Hybrid	0.164	0.193	0.357	1.74×	28,289
Drill, DEM	1.84	N/A	1.84	N/A	360,000
Drill, Hybrid	0.125	0.145	0.27	6.82×	42,933
Tires, DEM	3.60	N/A	3.60	N/A	588,320
Tires, Hybrid	0.54	0.51	1.05	3.43×	114,35

Table 1. Simulation performance. Timings of our hybrid approach compared to a purely discrete approach for different scenarios. All reported costs have units of average seconds per time step. The hybrid cost includes both the cost of MPM time evolution and the coupling solves. We take a constant DEM time step of $dt = 10^{-6}$ and a constant MPM time step of $dt = 10^{-5}$. We gathered all performance statistics on an Intel 3.5 GHz Core i7-4770K with a single thread.

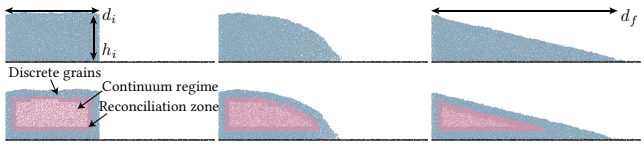


Fig. 5. Granular column: A collapsing column simulated with DEM (top) and with our hybrid approach (bottom). Observe the nice agreement in the final profile with our hybrid approach and the purely DEM approach.

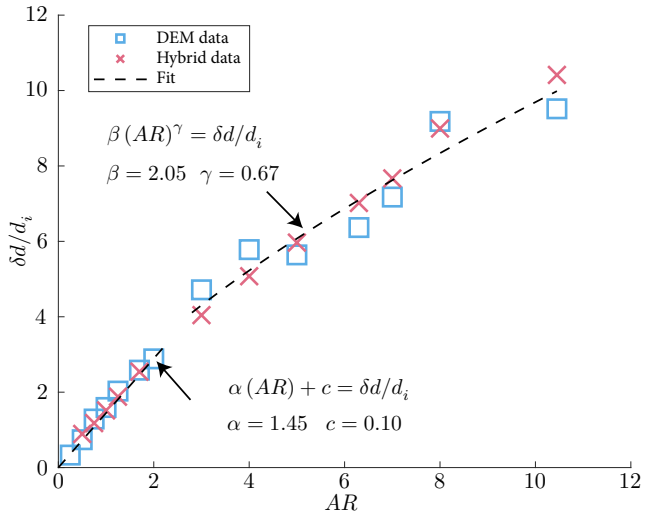


Fig. 6. Runout of a 2D granular column: Nondimensionalized runout distance, $\delta d/d_i = (d_f - d_i)/d_i$, vs aspect ratio, $AR = h_i/d_i$. Like DEM, our hybrid technique captures the two distinct regimes that Lube [2005] observed in experiments. We perform a linear fit in the low-AR regime, and a power-law fit in the high-AR regime.

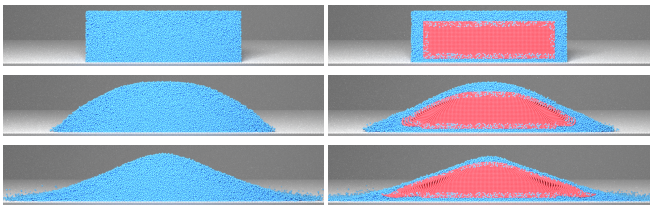


Fig. 7. 3D column collapse: a column collapse simulated with DEM (left) and with our hybrid method (right) at the same snapshots in time.

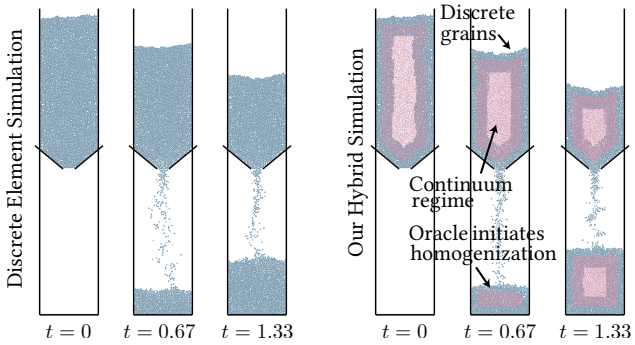


Fig. 8. Silo discharge: A silo discharges grains with a discrete method (left) and with our hybrid method (right).

In comparison, our data fits best with $\beta = 2.05$ and $\gamma = 0.67$. We thus obtain a good quantitative match to experimental results.

Extending to 3D, we also obtain a good qualitative matchup of the collapse in motion. Fig. 7 shows the motion of a discrete and a hybrid column collapse. Again note the grains at the edge of the runout, which the hybrid technique is able to capture.

Silo discharge. In Fig. 8, we simulate a silo that discharges grains using a purely discrete approach and our hybrid approach. With our approach, the oracle identifies the interior of the initial mass of grains as a continuum. As grains exit the silo and the continuum region falls towards the orifice, our method automatically converts the continuum material to discrete material. As grains form a pile on the ground, our method detects the formation of the sufficiently dense portions of the pile and automatically converts discrete grains to continuum material points in this area. The hybrid 2D hourglass has a slightly faster flow rate than the discrete only counterpart. We believe that the ability to control the coordination number for newly sampled DEM particles would reconcile these flow rates. Generating packings given constraints is an interesting avenue of future work.

We also simulate a hybrid silo discharge in 3D (Fig. 9). The hybrid approach is able to model ballistic motion and collisions after grains flowing from the silo enter a "gaseous" state. This ability to model contact is crucial for capturing the asymmetrical shape of the column, as well as the ballistic bounces when grains impact the container and the pile, both of which are observed in real-life hourglasses. See Fig. 10.

MPM fails at simulating fly away grains, as the Particle-in-Cell method handles collisions by homogenizing each Lagrangian particle's velocity onto a background Eulerian grid and then transferring back. This series of operations results in an effectively inelastic collision among Lagrangian particles, i.e. with restitution coefficient $e = 0$. Note that this inelastic nature is independent of the particle-grid transfer scheme, i.e. PIC, FLIP, or APIC.

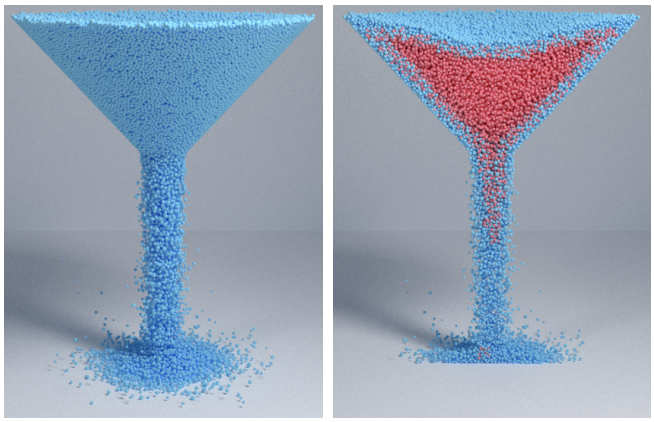


Fig. 9. 3D silo discharge simulated with our hybrid approach: Left is a full view of the discharging grains while right is a cutaway view.

Fig. 10. Hourglass experiment: Individual grains bounce off a sand pile, violating the continuum assumption and requiring a discrete model. ©MIT

In contrast, our hybrid approach handles the full range of restitution coefficients from 0 to 1. In Fig. 11, we show a comparison between $e = 0$ and $e = 0.5$. Notice that when $e = 0$, the hybrid approach also fails to capture the detailed bouncing effects and has a more uniform shape and flow profile.

Another advantage of our hybrid approach over a purely continuum method is the ability to frictionally jam due to so-called finite size effects. In Fig. 12, we simulate a silo discharge with a small orifice width using a purely DEM algorithm, our hybrid algorithm, and a purely continuum algorithm. Note that we use an MPM cell width to DEM mean grain diameter ratio of 1:1 to more accurately couple the hybrid region near the orifice. Our hybrid simulation, like the purely DEM simulation, jams with the small orifice width, as expected. On the contrary the continuum, regardless of the grid resolution, fails to capture these finite size effects. Extra non-local modeling is needed [Kamrin and Koval 2014].

Penetrometer insertion. Similar to Yan et al. [2010] and Wellmann and Wriggers [2012], we perform a hybrid simulation of a

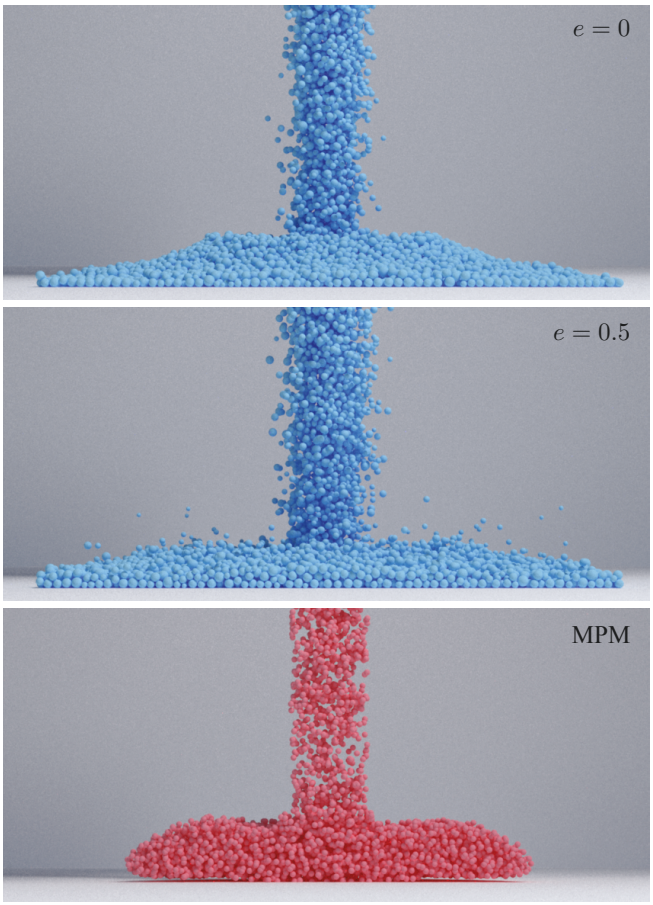


Fig. 11. Silo discharge (3D large orifice, $r = 1.8$): Our hybrid approach supports a range of restitution coefficients. Top: with 0 restitution, the flow looks uniform. Middle: with $e = 0.5$, the flow appears more energetic, with multiple visible fly away grains. Observe that the MPM version of this simulation (c) has both a less energetic flow, and fails to simulate grains bouncing away from the bulk whole. The MPM counterpart is simulated with sticky boundary conditions on the bottom.

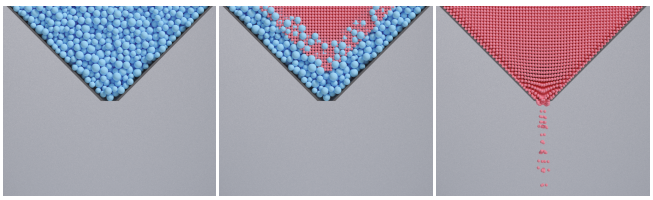


Fig. 12. Silo discharge (small orifice, $r = 0.2$): A silo discharges grains with a discrete method (left), our hybrid method (middle), and a continuum method (right). Both the discrete and hybrid approach capture size-dependent clogging effects, and all flow from the orifice halts. The continuum simulation, in contrast, flows nonphysically.

penetrometer insertion into a bed of grains (Fig. 13). These simulations are difficult to perform directly with standard continuum methods owing to the massive plastic shape changes observed around

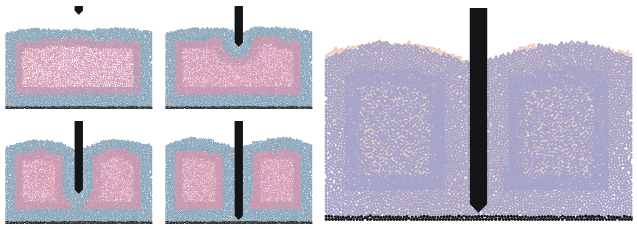


Fig. 13. Penetrometer insertion: We insert a penetrometer into a bed of grains with our hybrid algorithm (first four frames). As the penetrometer enters the bed, our hybrid oracle identifies the region around the tip as requiring a discrete treatment and enriches the simulation domain in this area. As the simulation progresses, the continuum region eventually experiences a topology change and splits in two. Examining an overlay of a hybrid simulation (purple) on a purely discrete simulation (peach), we find the resulting profiles to be in almost perfect agreement (rightmost frame).

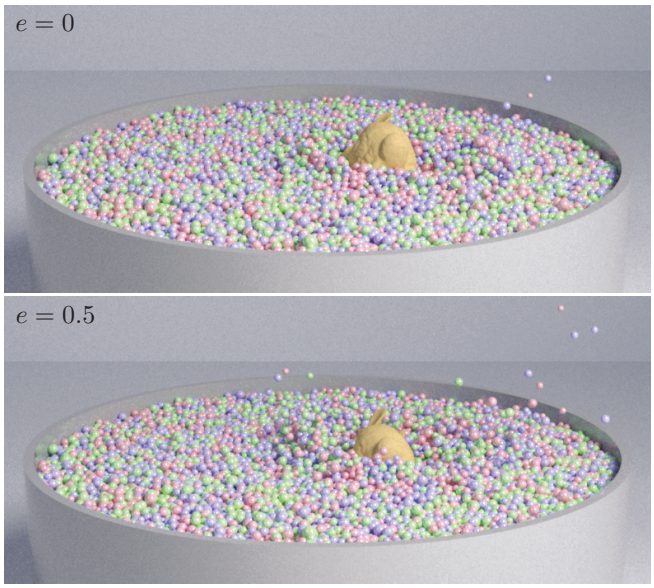


Fig. 14. The bunny crashes into a container of gumballs with different coefficients of restitution using our hybrid method.

the penetrometer tip. Unlike previous works, we do not specify the region to be treated with DEM a-priori. Instead, as the penetrometer advances into the bed of grains, our hybrid method is able to enrich the region surrounding the penetrometer, ensuring that it always interacts with the bed through discrete grains. As the penetrometer is fully inserted into the bed, the original single continuum region is split in two. Our hybrid approach gracefully treats this topological change with no extra machinery.

Bunny toss. We initialize a bunny with non-zero translational and angular velocity and simulate the resulting collision with a packing of gumballs. This high-speed bunny produces a splash upon impact with the gumballs before coming to rest. In Fig. 14, the top simulation has zero restitution, while the bottom simulation has restitution of

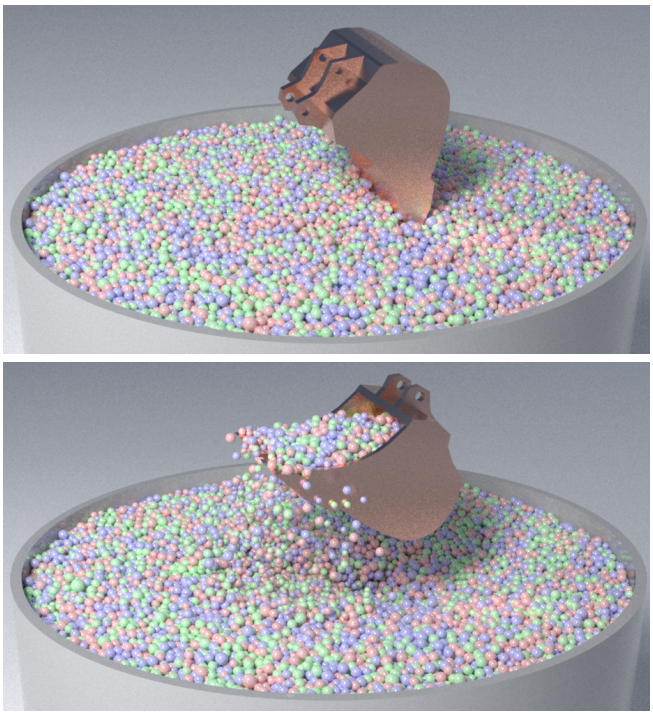


Fig. 15. We script an excavator to scoop gumballs out of a container.

$e = 0.5$. The larger coefficient of restitution leads to a simulation with a more dynamic splash.

While DEM uses 120,000 grains to simulate this scene, our hybrid approach only uses an average of 28,289 grains. In total, taking into account of the cost of MPM and the coupling computation (where again the MPM cell width is 2× the mean grain diameter), our method is 1.74× faster than DEM (Table 1).

Excavator. In Fig. 15, we script an excavator to scoop gumballs from the same container as the bunny toss. Our hybrid oracle robustly handles topology changes in the simulation domain induced by the excavator. As we employ the same gumball packing as the bunny toss in this test, we obtain a comparable speedup to that example.

Bunny drills. We aggressively insert and remove four scripted bunnies from a pile of 360,000 grains (Fig. 1). While initially only the top surface is represented as discrete grains, our method is able to dynamically enrich the interior around the bunnies in response to their motion. Our hybrid method obtains a similar visual result compared to the pure DEM simulation, yet it uses 88% fewer discrete grains and is thus 6.82× faster (See Table 1).

Tires on a gravel road. To test our method on a real world example, we simulate off-road tires traversing a gravel road. The tires are given a constant angular velocity around each of their axes, but are otherwise dynamically simulated. The hybrid method is able to capture multiple effects such as large splashes when fast rotating wheels collide with the grains, as well as tires sinking into the pile of grains due to a large density difference. While simulating this scene with a pure discrete method requires 822,956 grains, our hybrid approach allows us to simulate only a thin layer of discrete grains and the remainder is continuum (Fig. 16 (a)). Here, the MPM cell width is 1.75× the mean grain diameter. On average, the hybrid approach is 3.43× faster than a purely discrete method (Table 1) .

Spinning drum. An understanding of drum geometries is important in industrial applications (e.g. mills, tumblers) and in the study of free-surface flows [Groupement de Recherche Milieux Divisés (GDR MiDi) 2004]. To assess whether our algorithm is suitable for these geometries, we fill a drum with grains to half its area, and impose a rotation to the drum with a constant angular velocity. With a DEM simulation, we observe nearly rigid grains near the base of the drum, a steadily increasing flow towards the interior of the granular assembly, and loosely packed grains near the free surface. As the transient phase subsides, we observe the characteristic free-surface shape of these experiments. Comparing the purely discrete results to those from our hybrid algorithm (Fig. 17), we find the profiles to be in good agreement throughout the simulation. Because our hybrid algorithm treats regions near surfaces with discrete grains, we do not require any additional machinery to handle the drum boundary condition beyond that from the discrete simulation. Like the discrete simulation, our hybrid algorithm is also able to capture free flight fly away grains near the top of the domain.

Speedup study. We seek to quantify the speedup we are able to obtain from the hybrid method over a pure discrete simulation. In order to do this, a chute-flow geometry is taken, as seen in Fig. 18. Grains are initialized in a column and are then tilted at an angle θ relative to the horizontal, with gravity applied. Periodic boundary conditions are then enforced, allowing for a continual flow of grains down-slope. Three factors are adjusted: the initial total number of grains N_i before hybridization, the fraction of DEM left after the hybridization F, and the hybrid grid size (identical to the MPM grid size here) H. A parametric sweep adjusting these three variables allows for the construction of a phase plot, which shows for a given N_i, when a pure discrete simulation with N_i grains is faster or slower relative to a hybrid simulation initialized with N_i grains but with different F and H. Note that the geometry is kept fixed for all simulations, so that increasing or decreasing N_i means decreasing or increasing the average grain diameter.

N_i ranges from 1,000 grains up to 156,000 grains, F ranges from 0.07 to 0.89, and H ranges from 0.0025 to 0.000625. Cell width to mean grain diameter ratios thus range from 19:1 to 0.4:1. Fig. 19 displays phase plots over different values of H. As H decreases, more elements are hybridized, and so computational costs associated with hybridization increase. However, even for the most refined grid, a speedup can still be obtained with a reasonable F for a simulation requiring 40,000 grains or more. It can be seen from Fig. 19 that a speedup on the order of 12× can be obtained. While it may seem that increasing the resolution (thus decreasing H) results in a decay of the maximum speedup obtained, one can exploit a characteristic of a smaller H: with smaller H, a smaller F can be obtained.

Looking at the relationship between H and F from a different perspective, an analysis of the speedup for layered hybridization in 2D can be conducted from the geometry of the problem (see the supplemental material for a detailed derivation). Letting C_D be the computational cost of a discrete grain, C_E the cost of enrichment for a hybrid cell, C_C the cost for a continuum element, and A the

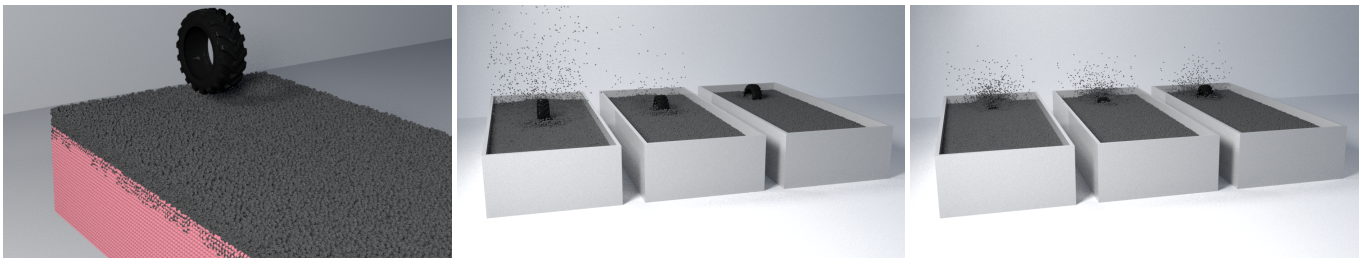


Fig. 16. Simulations of tires traversing a bed of gravel. Left: A render of the initial condition (boundary condition not shown), with a tire poised to race across the bed of grains. Notice the layered hybridization employed here. Center: Tires with different angular velocities but equal densities. Left to right the tires rotate at 1000 rad/s, 100 rad/s, and 10 rad/s. The 1000 rad/s tire produces a large granular splash while the 10 rad/s tire produces almost no splash. Right: Tires of different densities, but with the same angular velocities. Left to right the tires have 5×, 2×, and 1× the density of gravel. As the tires traverse the system, the larger density tires sink into the gravel.

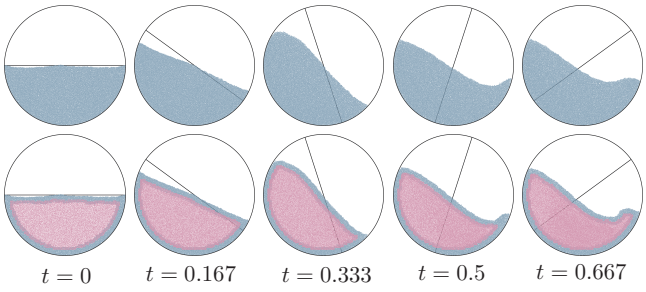


Fig. 17. Spinning drum: We rotate a drum filled halfway with grains using DEM (top row) and using our hybrid algorithm (bottom row). As the system evolves, observe that the shape of the free surface obtained with our hybrid method agrees with that of the purely discrete method.

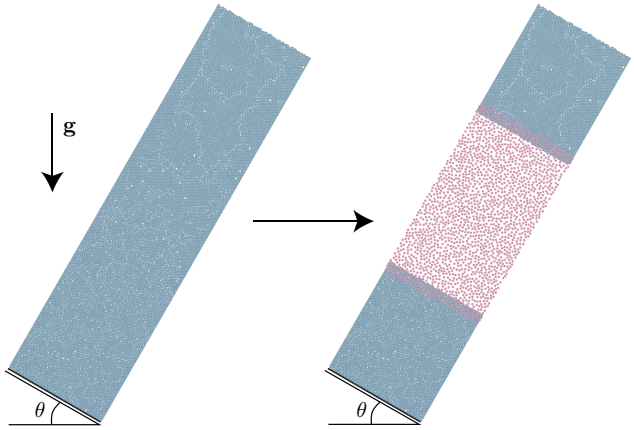


Fig. 18. Periodic chute flow: We set a granular packing at an angle θ with periodic boundary conditions to simulate flow down a chute.

effective total number of grains (average number of grains per cell multiplied with the number of cells containing grains), we obtain the following expression for the total time T_H for a complete hybrid iteration and total time T_C for a pure discrete simulation:

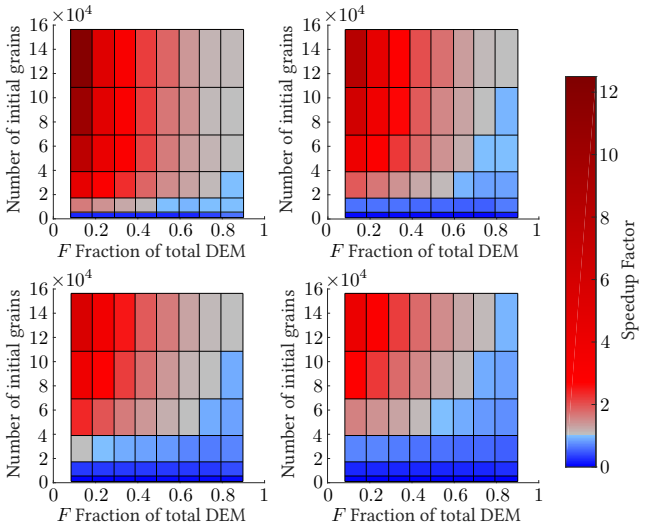


Fig. 19. Phase plots: Phase plots for $H = 0.0025$ (top left), $H = 0.00125$ (top right), $H = 0.00083$ (bottom left), and $H = 0.000625$ (bottom right). Red denotes regimes where our hybrid scheme is faster while blue denotes regimes where DEM is faster.

$$T_H = C_E hN^2 + C_D \frac{(N_D + N_H)N}{hN^2} + C_C(hN - N_D)N,$$
$$T_C = C_D A. \tag{21}$$

A reduction ratio R_A can then be defined between the computational time of a hybrid simulation and an equivalent discrete simulation of A grains:

$$R_A = \frac{T_H}{T_C} = \frac{C_E hN^2}{C_D A} + \frac{C_D(N_D + N_H)N}{hN} + \frac{C_C(hN - N_D)N}{C_D A}. \tag{22}$$

Minimizing R_A results in the largest speedup, and we can find the optimal N giving the largest speed up as:

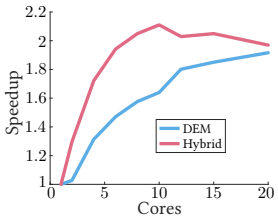
$$N = \left(\frac{C_D A}{h^2 (C_E + C_C)} \right)^{1/3}. \tag{23}$$

The key insight is that if N is chosen in the manner shown above in relation to A, then as $A \to \infty$, $R_A \to 0$, which means that increasing speedups can be had for increasing grain numbers. This is an extremely useful property, and serves to highlight the potential of our hybrid method to tackle problems bridging micro/mesoscale causes to macroscale effects.

For the sugar coated hybridization method in 3D described in Section 4.1 and the layered hybridization method in 3D described in Section 4.3, we obtain the best speedups if N scales with A according to $N \propto A^{1/4}$. Then as $A \to \infty$, $R_A \to 0$. See the supplemental material for the detailed derivation.

It is important to note that N scales with A in the power of 1/4 (1/3 in 2D), not 1/3 (1/2 in 2D). An intuitive explanation is that if we refine both the discrete and continuum elements equally (this corresponds to setting $N \propto A^{1/3}$ ($N \propto A^{1/2}$ in 2D)) while keeping the discrete layer thickness to a minimum, then the discrete computation time will scale as N^2 (N in 2D) whereas the continuum will scale as N^3 (N^2 in 2D), so eventually the continuum computation time will dominate, and we will hit a bound. However, if we refine them *differently* and maintain a balance between the two (i.e., setting $N \propto A^{1/4}$ in 3D and $N \propto A^{1/3}$ in 2D), then the acceleration continues.

Core scaling. To study how our code parallelizes, we conducted a test with the bunny drill. We ran both a pure DEM and a hybrid simulation with increasing core counts. For each core count, we ran each simulation for two hours of wall clock time and measured the number of completed time steps. As evident in the inset, both DEM and our hybrid method do parallelize, albeit with sublinear scaling. We note that at the time of this test, many routines in the code were not fully parallelized, and a complete refactoring with parallelization in mind would greatly improve scaling.



6 CONCLUSIONS AND FUTURE WORK

We have presented a theoretical framework for coupling a continuum model of a granular material to a discrete model. To develop this framework, we have proposed a method to partition a granular material into two distinct but overlapping systems. By modeling one system with a discrete model, and one system with a continuum model, we arrive at a hybrid treatment for a granular material. We then presented a discrete version of this framework able to accommodate a variety of integrators, and derived an explicit hybrid integrator that couples a material point simulation of a continuum to a discrete element simulation. We have demonstrated preliminary hybrid simulations with this integrator in settings that experience both large plastic shape changes and topology changes.

We are interested in extending our approach to treat both systems implicitly. The coupling framework naturally extends to other integrators, and a fully implicit approach promises significant performance improvements when integrating stiff systems, including granular materials. As contact dynamics models are implicit, an extension to implicit integrators would allow us to treat these models as well. Improvements are possible to our approach by constraining the angular momentum of the discrete system: if a grain with nonzero angular momentum and no active contacts is embedded in the hybrid zone, it will spin in place until it comes back into contact with the discrete system. While there is no notion of point-wise angular momentum in a continuum, certain discretizations, including the Affine Particle in Cell method, track angular momentum-like quantities, to which we could constrain the angular momentum. We are also interested in refining and exploring the limits of our methodology for detecting regions safe for homogenization and our methodology for resampling regions as the boundaries between continuum and discrete evolve. While we currently use uniform density weights of 1/2, we are interested in exploring how the use of smoothly varying weights might change the simulation results. In particular, our use of non-smooth weights could cause artifacts when analyzing wave propagation in granular systems. Currently, we do not explicitly control the coordination number of the newly inserted DEM particles in our enrichment. Generating packings with constraints is an interesting direction for future work.

While we are primarily concerned with discrete phenomena near the free surface of a system, it would be interesting to explore how to capture discrete effects in the interior. At present we are not able to capture internal shear banding, but extensions to our oracle could allow for enrichment in interior regions to produce these effects. Finally, our method does not model discrete force chains in the interior domain. While this is not a major approximation, as force chains homogenize into smooth stress fields over a window of a few grain diameters in 3D, it would be interesting to explore how to resolve these in the interior regions.

ACKNOWLEDGMENTS

We thank Henrique Teles Maia for his help in producing the video, Gabriel Cirio for creating the funnel mesh, as well as Sachith Dunatunga, Chenfanfu Jiang, Andre Pradhana, and Yun Fei for insightful discussions.

This work was supported in part by JSPS (Grant-in-Aid for Young Scientists (A) 17H04682), NSF (Grants CBET-17-06689, IIS-11-17257, IIS-12-17904, IIS-13-19483, IIS-14-09286, CMMI-11-29917, CMMI-13-31499), IARPA (Grant 2014-14071600012), Adobe, and Autodesk.

REFERENCES

Vincent Acary and Bernard Brogliato. 2008. *Numerical Methods for Nonsmooth Dynamical Systems: Applications in Mechanics and Electronics*. Vol. 35. Springer Science & Business Media.

Pierre Alart and Alain Curnier. 1991. A mixed formulation for frictional contact problems prone to Newton like solution methods. *Computer Methods in Applied Mechanics and Engineering* 92, 3 (Nov. 1991), 353–375.

Berni Julian Alder and Thomas Everett Wainwright. 1957. Phase transition for a hard sphere system. *The Journal of Chemical Physics* 27, 5 (1957), 1208.

Berni Julian Alder and Thomas Everett Wainwright. 1959. Studies in molecular dynamics. I. General method. *The Journal of Chemical Physics* 31, 2 (1959), 459–466.

Berni Julian Alder and Thomas Everett Wainwright. 1960. Studies in molecular dynamics. II. Behavior of a small number of elastic spheres. *The Journal of Chemical Physics* 33, 5 (1960), 1439–1451.

Iván Alduán and Miguel A. Otaduy. 2011. SPH Granular Flow with Friction and Cohesion. In *Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on*

Computer Animation (SCA '11). 25–32.
Iván Alduán, Angel Tena, and Miguel A. Otaduy. 2009. Simulation of High-Resolution Granular Media.. In *CEIG*. 11–18.
Christoph Ammann, Doug Bloom, Jonathan M. Cohen, John Courte, Lucio Flores, Sho Hasegawa, Nikos Kalaitzidis, Terrance Tornberg, Laurence Treweek, Bob Winter, and Chris Yang. 2007. The Birth of Sandman. In *ACM SIGGRAPH 2007 Sketches (SIGGRAPH '07)*. Article 26.
Mihai Anitescu and Gary D. Hart. 2004a. A constraint-stabilized time-stepping approach for rigid multibody dynamics with joints, contact and friction. *Internat. J. Numer. Methods Engrg.* 60, 14 (2004), 2335–2371.
Mihai Anitescu and Gary D. Hart. 2004b. A fixed-point Iteration approach for multibody dynamics with contact and small friction. *Mathematical Programming* 101, 1 (2004), 3–32.
Igor S. Aranson and Lev S. Tsimring. 2002. Continuum theory of partially fluidized granular flows. *Physical Review E* 65, 6 (2002), 061303.
Jan A. Åström and Hans Jürgen Herrmann. 1998. Fragmentation of grains in a two-dimensional packing. *The European Physical Journal B-Condensed Matter and Complex Systems* 5, 3 (1998), 551–554.
Neil J. Balmforth and Richard R. Kerswell. 2005. Granular collapse in two dimensions. *Journal of Fluid Mechanics* 538 (Sep 2005), 399 – 428.
David Baraff. 1989. Analytical Methods for Dynamic Simulation of Non-penetrating Rigid Bodies. *Computer Graphics* 23 (1989), 223–232.
Scott G. Bardenhagen and Edward M. Kober. 2004. The generalized interpolation material point method. *Computer Modeling in Engineering and Sciences* 5, 6 (2004), 477–496.
Nathan Bell, Yizhou Yu, and Peter J. Mucha. 2005. Particle-based simulation of granular materials. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*. ACM, 77–86.
Oded Ben-Nun, Itai Einav, and Antoinette Tordesillas. 2010. Force attractor in confined comminution of granular materials. *Physical Review Letters* 104, 10 (2010), 108001.
Miklós Bergou, Saurabh Mathur, Max Wardetzky, and Eitan Grinspun. 2007. TRACKS: Toward directable thin shells. *SIGGRAPH (ACM Transactions on Graphics)* 26, 3 (Jul 2007), 50.
Wim A. Beverloo, H. A. Leniger, and J. Van de Velde. 1961. The flow of granular solids through orifices. *Chemical Engineering Science* 15, 3 (1961), 260–269.
Olivier Bonnefon and Gilles Daviet. 2011. *Quartic formulation of Coulomb 3D frictional contact*. Technical Report RT-0400. INRIA.
Jean-Philippe Bouchaud, Michael E. Cates, Jagadeeshan Ravi Prakash, and Sam F. Edwards. 1994. A model for the dynamics of sandpile surfaces. *Journal de Physique I* 4, 10 (1994), 1383–1410.
Bernard Brogliato. 2012. *Nonsmooth mechanics: models, dynamics and control*. Springer Science & Business Media.
Eric Brown, Nicholas Rodenberg, John Amend, Annan Mozeika, Erik Steltz, Mitchell R. Zakin, Hod Lipson, and Heinrich M. Jaeger. 2010. Universal robotic gripper based on the jamming of granular material. *Proceedings of the National Academy of Sciences* 107, 44 (2010), 18809–18814.
Benoit Chanclou, Annie Luciani, and Arash Habibi. 1996. Physical models of loose soils dynamically marked by a moving object. In *Computer Animation'96. Proceedings*. 27–35.
Nuttapong Chentanez, Matthias Müller, and Tae-Yong Kim. 2015. Coupling 3D Eulerian, Heightfield and Particle Methods for Interactive Simulation of Large Scale Liquid Phenomena. *IEEE Transactions on Visualization and Computer Graphics* 21, 10 (Oct 2015), 1116–1128.
Jes Christoffersen, Morteza Monte Mehrabadi, and Sia Nemat-Nasser. 1981. A micromechanical description of granular material behavior. *Journal of applied mechanics* 48, 2 (1981), 339–344.
Michael B. Cline and Dinesh K. Pai. 2003. Post-stabilization for rigid body simulation with contact and constraints. In *Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on*, Vol. 3. IEEE, 3744–3751.
Peter A. Cundall and Otto D. L. Strack. 1979. A discrete numerical model for granular assemblies. *Géotechnique* 29, 1 (1979), 47–65.
Yannis F. Dafalias, Achilleas G. Papadimitriou, and Xiang S. Li. 2004. Sand plasticity model accounting for inherent fabric anisotropy. *Journal of Engineering Mechanics* 130, 11 (2004), 1319–1333.
Gilles Daviet and Florence Bertails-Descoubes. 2016. A Semi-implicit Material Point Method for the Continuum Simulation of Granular Materials. *ACM Trans. Graph.* 35, 4 (July 2016), 102:1–102:13.
Gilles Daviet, Florence Bertails-Descoubes, and Laurence Boissieux. 2011. A Hybrid Iterative Solver for Robustly Capturing Coulomb Friction in Hair Dynamics. *ACM Trans. Graph.* 30, 6 (2011), 139:1–139:12.
Hachmi Ben Dhia. 1998. Multiscale mechanical problems: the Arlequin method. *Comptes Rendus de l'Academie des Sciences Series IIB Mechanics Physics Astronomy* 12, 326 (1998), 899–904.
Joshua A Dijksman and Martin van Hecke. 2010. Granular flows in split-bottom geometries. *Soft Matter* 6, 13 (2010), 2901–2907.
Sachith Dunatunga and Ken Kamrin. 2015. Continuum modelling and simulation of granular flows through their many phases. *Journal of Fluid Mechanics* 779 (2015), 483–513.
Christian Duriez, Frederic Dubois, Abderrahmane Kheddar, and Claude Andriot. 2006. Realistic Haptic Rendering of Interacting Deformable Objects in Virtual Environments. *IEEE Transactions on Visualization and Computer Graphics* 12, 1 (Jan. 2006), 36–47.
Kenny Erleben. 2007. Velocity-based Shock Propagation for Multibody Dynamics Animation. *ACM Trans. Graph.* 26, 2, Article 12 (June 2007). https://doi.org/10.1145/1243980.1243986
Florian Ferstl, Ryoichi Ando, Chris Wojtan, Rüdiger Westermann, and Nils Thuerey. 2016. Narrow Band FLIP for Liquid Simulations. *Computer Graphics Forum* 35, 2 (2016), 225–232.
Daan Frenkel and Berend Smit. 2001. *Understanding Molecular Simulation: From Algorithms to Applications*. Vol. 1. Academic Press. https://doi.org/10.1016/B978-012267351-1/50000-6
Jason Alfredo Carlson Gallas, Hans Jürgen Herrmann, and Stefan Sokołowski. 1992. Convection cells in vibrating granular media. *Physical Review Letters* 69, 9 (1992), 1371.
Ming Gao, Andre Pradhana, Xuchen Han, Qi Guo, Grant Kot, Eftychios Sifakis, and Chenfanfu Jiang. 2018. Animating Fluid Sediment Mixture in Particle-Laden Flows. *ACM Transactions on Graphics (TOG)* 37, 4 (2018).
Ming Gao, Andre Pradhana Tampubolon, Chenfanfu Jiang, and Eftychios Sifakis. 2017. An adaptive generalized interpolation material point method for simulating elasto-plastic materials. *ACM Transactions on Graphics (TOG)* 36, 6 (2017), 223.
Abhinav Golas, Rahul Narain, Jason Sewall, Pavel Krajcevski, Pradeep Dubey, and Ming C. Lin. 2012. Large-scale Fluid Simulation Using Velocity-vorticity Domain Decomposition. *ACM Trans. Graph.* 31, 6, Article 148 (Nov. 2012), 9 pages.
Groupement de Recherche Milieux Divisés (GDR MiDi). 2004. On dense granular flows. *The European Physical Journal E* 14, 4 (2004), 341–365.
Peter K. Haff and Bradley T. Werner. 1986. Computer simulation of the mechanical sorting of grains. *Powder Technology* 48, 3 (1986), 239–245.
Seth R. Holladay. 2013. *Optimized Simulation of Granular Materials*. Ph.D. Dissertation. Brigham Young University.
Seth R. Holladay and Parris Egbert. 2012. Solid-state Culled Discrete Element Granular Systems. In *Eurographics 2012 - Short Papers*, Carlos Andujar and Enrico Puppo (Eds.). The Eurographics Association. https://doi.org/10.2312/conf/EG2012/short/065-068
William Graham Hoover. 1986. *Molecular Dynamics*. Springer-Verlag.
Shu-Wei Hsu and John Keyser. 2010. Piles of Objects. *ACM Trans. Graph.* 29, 6 (Dec. 2010), 155:1–155:6.
Yuanming Hu, Yu Fang, Ziheng Ge, Ziyin Qu, Yixin Zhu, Andre Pradhana, and Chenfanfu Jiang. 2018. A Moving Least Squares Material Point Method with Displacement Discontinuity and Two-Way Rigid Body Coupling. *ACM Transactions on Graphics (TOG)* 37, 4 (2018).
Markus Ihmsen, Arthur Wahl, and Matthias Teschner. 2013. A Lagrangian framework for simulating granular material with high detail. *Computers & Graphics* 37, 7 (2013), 800–808.
Heinrich M. Jaeger, Sidney R. Nagel, and Robert P. Behringer. 1996. Granular solids, liquids, and gases. *Rev. Mod. Phys.* 68 (Oct 1996), 1259–1273. Issue 4. https://doi.org/10.1103/RevModPhys.68.1259
Michel Jean. 1999. The Non-Smooth Contact Dynamics Method. *Computer Methods in Applied Mechanical Engineering* 177, 3-4 (1999), 235–257.
Michel Jean and Jean-Jacques Moreau. 1992. Unilaterality and dry friction in the dynamics of rigid body collections. In *Proceedings of Contact Mechanics International Symposium*, Vol. 1. 31–48.
Chenfanfu Jiang, Craig Schroeder, Andrew Selle, Joseph Teran, and Alexey Stomakhin. 2015. The affine particle-in-cell method. *ACM Transactions on Graphics (TOG)* 34, 4 (2015), 51.
Chenfanfu Jiang, Craig Schroeder, Joseph Teran, Alexey Stomakhin, and Andrew Selle. 2016. The Material Point Method for Simulating Continuum Materials. In *ACM SIGGRAPH 2016 Courses (SIGGRAPH '16)*. ACM, New York, NY, USA, Article 24, 52 pages. https://doi.org/10.1145/2897826.2927348
Pierre Jop, Yoël Forterre, and Olivier Pouliquen. 2006. A constitutive law for dense granular flows. *Nature* 441, 7094 (2006), 727–730.
Franck Jourdan, Pierre Alart, and Michel Jean. 1998. A Gauss-Seidel like algorithm to solve frictional contact problems. *Computer Methods in Applied Mechanics and Engineering* 155, 1 (March 1998), 31–47.
Ken Kamrin. 2008. *Stochastic and Deterministic Models for Dense Granular Flow*. Ph.D. Dissertation. Massachusetts Institute of Technology.
Ken Kamrin. 2010. Nonlinear elasto-plastic model for dense granular flow. *International Journal of Plasticity* 26, 2 (2010), 167–188.
Ken Kamrin and Georg Koval. 2012. Nonlocal constitutive relation for steady granular flow. *Physical Review Letters* 108, 17 (2012), 178301.
Ken Kamrin and Georg Koval. 2014. Effect of Particle Surface Friction on Nonlocal Constitutive Behavior of Flowing Granular Media. *Computational Particle Mechanics* 1, 2 (2014), 169–176.

Danny M. Kaufman, Shinjiro Sueda, Doug L. James, and Dinesh K. Pai. 2008. Staggered Projections for Frictional Contact in Multibody Systems. *ACM Trans. Graph.* 27, 5 (2008), 164:1–164:11.

Gergely Klár, Theodore Gast, Andre Pradhana, Chuyuan Fu, Craig Schroeder, Chenfanfu Jiang, and Joseph Teran. 2016. Drucker-prager Elastoplasticity for Sand Animation. *ACM Trans. Graph.* 35, 4, Article 103 (July 2016), 12 pages. https://doi.org/10.1145/2897824.2925906

Georg Koval, Jean-Noël Roux, Alain Corfdir, and François Chevoir. 2009. Annular shear of cohesionless granular materials: From the inertial to quasistatic regime. *Physical Review E* 79, 2 (2009), 021306.

Harald Kruggel-Emden, Erdem Simsek, Stefan Rickelt, Siegmar Wirtz, and Viktor Scherer. 2007. Review and extension of normal force models for the discrete element method. *Powder Technology* 171, 3 (2007), 157–173.

Pierre-Yves Lagrée, Lydie Staron, and Stéphane Popinet. 2011. The granular column collapse as a continuum: Validity of a two-dimensional Navier-Stokes model with a $\mu(I)$-rheology. *Journal of Fluid Mechanics* 686 (Nov. 2011), 378–408.

Cornelius Lanczos. 2012. *The variational principles of mechanics.*

Toon Lenaerts and Dutré Philip. 2009. Mixing Fluids and Granular Materials. *Computer Graphics Forum* 28, 2 (2009), 213–218.

Xin Li and J. Michael Moshell. 1993. Modeling Soil: Realtime Dynamic Models for Soil Slippage and Manipulation. In *Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '93)*. 361–368.

Gert Lube, Herbert E. Huppert, R. Stephen J. Sparks, and Armin Freundt. 2005. Collapses of two-dimensional granular columns. *Phys. Rev. E* 72 (Oct 2005), 10. Issue 4.

Annie Luciani, Arash Habibi, and Emmanuel Manzotti. 1995. A multi-scale physical model of granular materials. In *Graphics interface'95*. 136–146.

Miles Macklin, Matthias Müller, Nuttapong Chentanez, and Tae-Yong Kim. 2014. Unified Particle Physics for Real-time Applications. *ACM Trans. Graph.* 33, 4, Article 153 (July 2014), 12 pages.

Carter M. Mast, Pedro Arduino, Peter Mackenzie-Helnwein, and Gregory R. Miller. 2015. Simulating granular column collapse using the Material Point Method. *Acta Geotechnica* 10, 1 (2015), 101–116.

Hammad Mazhar, Toby Heyn, Dan Negrut, and Alessandro Tasora. 2015. Using Nesterov's Method to Accelerate Multibody Dynamics with Friction and Contact. *ACM Trans. Graph.* 34, 3, Article 32 (May 2015), 14 pages. https://doi.org/10.1145/2735627

Joseph J. McCarthy and Julio M. Ottino. 1998. Particle dynamics simulation: a hybrid technique applied to granular mixing. *Powder Technology* 97, 2 (1998), 91 – 99.

Gavin Miller and Andrew Pearce. 1989. Globular dynamics: A connected particle system for animating viscous fluids. *Computers & Graphics* 13, 3 (1989), 305 – 309.

Hiroshi Mio, Masatoshi Akashi, Atsuko Shimosaka, Yoshiyuki Shirakawa, Jusuke Hidaka, and Shinroku Matsuzaki. 2009. Speed-up of computing time for numerical analysis of particle charging process by using discrete element method. *Chemical Engineering Science* 64, 5 (2009), 1019 – 1026.

L. Srinivasa Mohan, K. Kesava Rao, and Prabhu R. Nott. 2002. A frictional Cosserat model for the slow shearing of granular materials. *Journal of Fluid Mechanics* 457 (2002), 377–409.

Jean-Jacques Moreau. 1983. *Standard Inelastic Shocks and the Dynamics of Unilateral Constraints.* Courses and Lectures, Vol. 288. International Centre for Mechanical Sciences, 173–221.

Jean-Jacques Moreau. 1988. Unilateral Contact and Dry Friction in Finite Freedom Dynamics. *Nonsmooth Mechanics and Applications, CISM Courses and Lectures* 302 (1988), 1–82.

Kevin W. Munns. 2015. *Gaseous Particulate Interaction in a 3-Phase Granular Simulation.* Master's thesis. Brigham Young University.

Rahul Narain, Abhinav Golas, and Ming C. Lin. 2010. Free-flowing Granular Materials with Two-way Solid Coupling. *ACM Trans. Graph.* 29, 6, Article 173 (Dec. 2010), 10 pages.

Duc-Hanh Nguyen, Emilien Azéma, Farhang Radjai, and Philippe Sornay. 2015. Numerical Modeling of Particle Breaking Process in Granular Materials: Compaction and Evolution of Size Distribution. In *Bifurcation and Degradation of Geomaterials in the New Millennium.* Springer, 161–167.

Koichi Onoue and Tomoyuki Nishita. 2003. Virtual sandbox. In *11th Pacific Conference onComputer Graphics and Applications, 2003. Proceedings.* 252–259.

Marta Pla-Castells, Ignacio García-Fernandez, and Rafael J. Martinez-Dura. 2008. Physically-Based Interactive Sand Simulation. In *Eurographics 2008 - Short Papers.*

Thorsten Pöschel and Thomas Schwager. 2005. *Computational Granular Dynamics: Models and Algorithms.* Springer Science & Business Media.

Olivier Pouliquen. 1999. Scaling laws in granular flows down rough inclined planes. *Physics of Fluids (1994-present)* 11, 3 (1999), 542–548.

Tobias Preclik. 2014. *Models and Algorithms for Ultrascale Simulations of Non-smooth Granular Dynamics.* Ph.D. Dissertation. Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU).

Dennis C. Rapaport. 2004. *The Art of Molecular Dynamics Simulation.* Cambridge University Press.

Patrick Richard, Mario Nicodemi, Renaud Delannay, Philippe Ribière, and Daniel Bideau. 2005. Slow relaxation and compaction of granular systems. *Nature Materials.* 4 (2005), 121–128.

Leo Rothenburg and Richard J. Bathurst. 1989. Analytical study of induced anisotropy in idealized granular materials. *Géotechnique* 39, 4 (1989), 601–614.

Witawat Rungjiratananon, Zoltan Szego, Yoshihiro Kanamori, and Tomoyuki Nishita. 2008. Real-time Animation of Sand-Water Interaction. In *Computer Graphics Forum*, Vol. 27. 1887–1893.

Chris H. Rycroft, Ken Kamrin, and Martin Z. Bazant. 2009. Assessing continuum postulates in simulations of granular flow. *Journal of the Mechanics and Physics of Solids* 57, 5 (2009), 828–839.

Andrew Schofield and Peter Wroth. 1968. *Critical State Soil Mechanics.* McGraw-Hill, London.

Ahmed A. Shabana. 2013. *Dynamics of multibody systems.* Cambridge university press.

J. Shäfer, S. Dippel, and D. E. Wolf. 1996. Force schemes in simulations of granular materials. *Journal de Physique I* 6, 1 (1996), 5–20.

Hannah G. Sheldon and Douglas J. Durian. 2010. Granular discharge and clogging for tilted hoppers. *Granular Matter* 12, 6 (2010), 579–585.

Tomotsugu Shimokawa, Toshiyasu Kinari, and S Shintaku. 2007. Interaction mechanism between edge dislocations and asymmetrical tilt grain boundaries investigated via quasicontinuum simulations. *Physical Review B* 75, 14 (2007), 144108.

Juan C. Simo and Thomas J. R. Hughes. 1998. *Computational Inelasticity.* Springer.

Breannan Smith, Danny M. Kaufman, Etienne Vouga, Rasmus Tamstorf, and Eitan Grinspun. 2012. Reflections on Simultaneous Impact. *ACM Trans. Graph.* 31, 4 (July 2012), 106:1–106:12.

Gregory S. Smith, Ellad B. Tadmor, Noam Bernstein, and Efthimios Kaxiras. 2001. Multiscale simulations of silicon nanoindentation. *Acta Materialia* 49, 19 (2001), 4089–4101.

Russell Smith and others. 2005. Open dynamics engine. (2005).

Lydie Staron and John E. Hinch. 2005. Study of the collapse of granular columns using two-dimensional discrete-grain simulation. *Journal of Fluid Mechanics* 545 (2005), 1–27.

Michael Steffen, Robert M. Kirby, and Martin Berzins. 2008. Analysis and reduction of quadrature errors in the material point method (MPM). *Internat. J. Numer. Methods Engrg.* 76, 6 (2008), 922–948. https://doi.org/10.1002/nme.2360

David E. Stewart. 2000. Rigid-body dynamics with friction and impact. *SIAM review* 42, 1 (2000), 3–39.

David E. Stewart. 2001. Finite-dimensional contact mechanics. *Phil. Trans. R. Soc. Lond. A* 359 (2001), 2467–2482.

David E. Stewart. 2011. *Dynamics with Inequalities: Impacts and Hard Constraints.* SIAM. https://doi.org/10.1137/1.9781611970715

David E. Stewart and Jeff C. Trinkle. 1996. An Implicit Time-Stepping Scheme for Rigid Body Dynamics with Coulomb Friction. *Internat. J. Numer. Methods Engrg.* 39, 15 (1996), 2673–2691.

Alexey Stomakhin, Craig Schroeder, Lawrence Chai, Joseph Teran, and Andrew Selle. 2013. A Material Point Method for Snow Simulation. *ACM Trans. Graph.* 32, 4, Article 102 (July 2013), 10 pages. https://doi.org/10.1145/2461912.2461948

Deborah Sulsky, Zhen Chen, and Howard L. Schreyer. 1994. A particle method for history-dependent materials. *Computer methods in applied mechanics and engineering* 118, 1 (1994), 179–196.

Robert W. Sumner, James F. O'Brien, and Jessica K. Hodgins. 1999. Animating Sand, Mud, and Snow. *Computer Graphics Forum* 18, 1 (1999), 17–26.

Ellad B. Tadmor, Rob Phillips, and Michael Ortiz. 1996. Mixed atomistic and continuum models of deformation in solids. *Langmuir* 12, 19 (1996), 4529–4534.

Andre Pradhana Tampubolon, Theodore Gast, Gergely Klár, Chuyuan Fu, Joseph Teran, Chenfanfu Jiang, and Ken Museth. 2017. Multi-species Simulation of Porous Sand and Water Mixtures. *ACM Trans. Graph.* 36, 4, Article 105 (July 2017), 11 pages.

Richard Tonge, Feodor Benevolenski, and Andrey Voroshilov. 2012. Mass Splitting for Jitter-free Parallel Rigid Body Simulation. *ACM Trans. Graph.* 31, 4, Article 105 (July 2012), 8 pages. https://doi.org/10.1145/2185520.2185601

Andrea Toselli and Olof Widlund. 2006. *Domain decomposition methods–algorithms and theory.* Vol. 34. Springer Science & Business Media.

Olivier Tsoungui, Denis Vallet, and Jean-Claude Charmet. 1999. Numerical model of crushing of grains inside two-dimensional granular materials. *Powder Technology* 105, 1 (1999), 190–198.

Thomas K. Uchida, Michael A. Sherman, and Scott L. Delp. 2015. Making a meaningful impact: Modelling simultaneous frictional collisions in spatial multibody systems. In *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, Vol. 471. The Royal Society, 20140859. https://doi.org/10.1098/rspa.2014.0859

Otis R. Walton and Robert L. Braun. 1986. Viscosity, granular-temperature, and stress calculations for shearing assemblies of inelastic, frictional disks. *Journal of Rheology* 30, 5 (1986), 949–980.

Chang-bo Wang, Qiang Zhang, Fan-long Kong, and Hong Qin. 2013. Hybrid particle–grid fluid animation with enhanced details. *The Visual Computer* 29, 9 (01 Sep 2013), 937–947.

Christian Wellmann and Peter Wriggers. 2012. A two-scale model of granular materials. *Computer Methods in Applied Mechanics and Engineering* 205-208, Supplement C

(2012), 46 – 58.
Zdzisław Więckowski. 2004. The material point method in large strain engineering problems. *Computer methods in applied mechanics and engineering* 193, 39-41 (2004), 4417–4438.
Beichuan Yan, Richard A. Regueiro, and Stein Sture. 2010. Three-dimensional ellipsoidal discrete element modeling of granular materials and its coupling with finite element facets. *Engineering Computations* 27, 4 (2010), 519–550.
Yonghao Yue, Breannan Smith, Christopher Batty, Changxi Zheng, and Eitan Grinspun. 2015. Continuum Foam: A Material Point Method for Shear-Dependent Flows. *ACM Trans. Graph.* 34, 5, Article 160 (Nov. 2015), 20 pages.
Zhennan Zhang and Xiurun Ge. 2005. A new quasi-continuum constitutive model for crack growth in an isotropic solid. *European Journal of Mechanics-A/Solids* 24, 2 (2005), 243–252.
Bo Zhu and Xubo Yang. 2010. Animating sand as a surface flow. *Eurographics 2010, Short Papers* (2010).
Yongning Zhu and Robert Bridson. 2005. Animating sand as a fluid. 24, 3 (2005), 965–972.
Olgierd Cecil Zienkiewicz and Robert L. Taylor. 2000. *The finite element method: The basis* (5 ed.). Vol. 1. Butterworth and Heinemann.

A DERIVATION OF THE PREDICTOR-CORRECTOR HYBRID TIME INTEGRATOR

Consider the generic (discretized) systems G_c and G_d coupled through constraints C. The state at the next time step (denoted by the 1 superscript) is given by parameters that satisfy:

$$G_c\left(\boldsymbol{v}_c^1, \boldsymbol{\lambda}^1\right) = 0, \tag{24}$$

$$G_d\left(\boldsymbol{v}_d^1, \boldsymbol{\lambda}^1\right) = 0, \tag{25}$$

$$C\left(\boldsymbol{v}_c^1, \boldsymbol{v}_d^1\right) = 0. \tag{26}$$

Note that while the G systems can be explicit or implicit, we presume that the constraints are implicit and thus evaluated at the end-of-step state. We omit $\boldsymbol{x}$ for clarity, noting that for typical integrators of interest $\boldsymbol{x}$ can be expressed in terms of $\boldsymbol{v}$.

A Newton algorithm for solving this system takes the form of Alg. 19. The linear system solved at each iteration is given by:

$$\begin{bmatrix} \frac{\partial G_c}{\partial \boldsymbol{v}_c^1} & 0 & \Gamma_c \\ 0 & \frac{\partial G_d}{\partial \boldsymbol{v}_d^1} & -\Gamma_d \\ \Gamma_c^T & -\Gamma_d^T & 0 \end{bmatrix} \begin{bmatrix} \Delta \boldsymbol{v}_c \\ \Delta \boldsymbol{v}_d \\ \Delta\lambda \end{bmatrix} = \begin{bmatrix} -G_c \\ -G_d \\ -C \end{bmatrix}. \tag{27}$$

To arrive at our hybrid integrator, first note that if both integrators are explicit, the nonlinear solve will reduce to a single linear solve, with $\Delta \boldsymbol{v}_c = \boldsymbol{v}_c^1 - \boldsymbol{v}_c^0$ and $\Delta \boldsymbol{v}_d = \boldsymbol{v}_d^1 - \boldsymbol{v}_d^0$. Next, observe that if both integrators are explicit, $\partial G_c / \partial \boldsymbol{v}_c^1 = M_c$ and $\partial G_d / \partial \boldsymbol{v}_d^1 = M_d$. Finally, we have the choice of initial iterate in Alg. 19. If we set $\boldsymbol{v}_{c,0} = 0$, $\boldsymbol{v}_{d,0} = 0$, and $\boldsymbol{\lambda}_0 = 0$, the linear system reduces to that of (20), revealing the predictor-corrector structure of the method.

B DERIVATION OF THE GRID VS. GRID HYBRID FORCE COMPUTATION

When constraining the velocities of co-located grids to match, the constraint matrices Γ_c and Γ_d in Eq. (20) reduce to identity, giving the system of equations:

$$W_c M_c \boldsymbol{v}_c^{n+1} + \lambda = W_c M_c \boldsymbol{v}_c^*, \tag{28}$$

$$W_d M_d \boldsymbol{v}_d^{n+1} - \lambda = W_d M_d \boldsymbol{v}_d^*, \tag{29}$$

$$\boldsymbol{v}_c^{n+1} = \boldsymbol{v}_d^{n+1}. \tag{30}$$

Algorithm 19 `Coupled_Step` $\left(\boldsymbol{v}_c^0, \boldsymbol{v}_d^0\right)$

$\left(\boldsymbol{v}_{c,0}, \boldsymbol{v}_{d,0}, \boldsymbol{\lambda}_0\right) \leftarrow$ `Generate_Initial_Iterate`
for $k = 0 \ldots$ `Max_Iterations` **do**
  $G_{c,k} \leftarrow G_c(\boldsymbol{v}_{c,k}, \boldsymbol{\lambda}_k)$ ▷ RHS
  $G_{d,k} \leftarrow G_d(\boldsymbol{v}_{d,k}, \boldsymbol{\lambda}_k)$
  $C_k \leftarrow C(\boldsymbol{v}_{c,k}, \boldsymbol{v}_{d,k})$
  `Residual` $\leftarrow \left|G_{c,k}, G_{d,k}, C_k\right|_\infty$ ▷ Termination check
  if `Residual` $\leq \epsilon$ **then**
    `Break`
  end if
  $\frac{\partial G_c}{\partial \boldsymbol{v}_c^1}_k \leftarrow \frac{\partial G_c}{\partial \boldsymbol{v}_c^1}(\boldsymbol{v}_{c,k})$ ▷ LHS
  $\frac{\partial G_d}{\partial \boldsymbol{v}_d^1}_k \leftarrow \frac{\partial G_d}{\partial \boldsymbol{v}_d^1}(\boldsymbol{v}_{d,k})$
  $\left(\Delta \boldsymbol{v}_c, \Delta \boldsymbol{v}_d, \Delta \boldsymbol{\lambda}\right) \leftarrow$ `Solve` $\left(G_{c,k}, G_{d,k}, C_k, \frac{\partial G_c}{\partial \boldsymbol{v}_c^1}_k, \frac{\partial G_d}{\partial \boldsymbol{v}_d^1}_k\right)$
  $\boldsymbol{v}_{c,k+1} \leftarrow \boldsymbol{v}_{c,k} + \Delta \boldsymbol{v}_c$
  $\boldsymbol{v}_{d,k+1} \leftarrow \boldsymbol{v}_{d,k} + \Delta \boldsymbol{v}_d$
  $\boldsymbol{\lambda}_{k+1} \leftarrow \boldsymbol{\lambda}_k + \Delta \boldsymbol{\lambda}$
end for

Substituting $\boldsymbol{v}_c^{n+1}$ for $\boldsymbol{v}_d^{n+1}$ in Eq. (28), adding Eq. (28) and Eq. (29), and solving for $\boldsymbol{v}_d^{n+1}$, we find that:

$$\boldsymbol{v}_c^{n+1} = \boldsymbol{v}_d^{n+1} = \left(W_c M_c + W_d M_d\right)^{-1} \left(W_c M_c \boldsymbol{v}_c^* + W_d M_d \boldsymbol{v}_d^*\right). \tag{31}$$

For diagonal mass matrices, each degree of freedom can be solved for independently, and the formula reduces to an inelastic impact between two particles in one dimension.

C MPM TIME STEP BREAKDOWN

Please see the supplement for detailed MPM pseudocode.

Algorithm 20 `MPM_Step_First_Phase`

`Rasterize_Mass_And_Momentum_To_Grid`
`Compute_Stress_At_Points`
`Compute_Forces_On_Grid`
`Update_Momentum_On_Grid`

Algorithm 21 `MPM_Step_Second_Phase`

`Lumped_Mass_Velocity_Update_On_Grid`
`Compute_Velocity_Gradient_At_Points`
`Elastic_Prediction_At_Points`
`Plastic_Correction_At_Points`
`Update_Velocities_At_Points`
`Update_Positions_At_Points`