FrictionalMonolith: A Monolithic Optimization-based Approach for Granular Flow with Contact-Aware Rigid-Body Coupling

TETSUYA TAKAHASHI, Adobe, USA
CHRISTOPHER BATTY, University of Waterloo, Canada

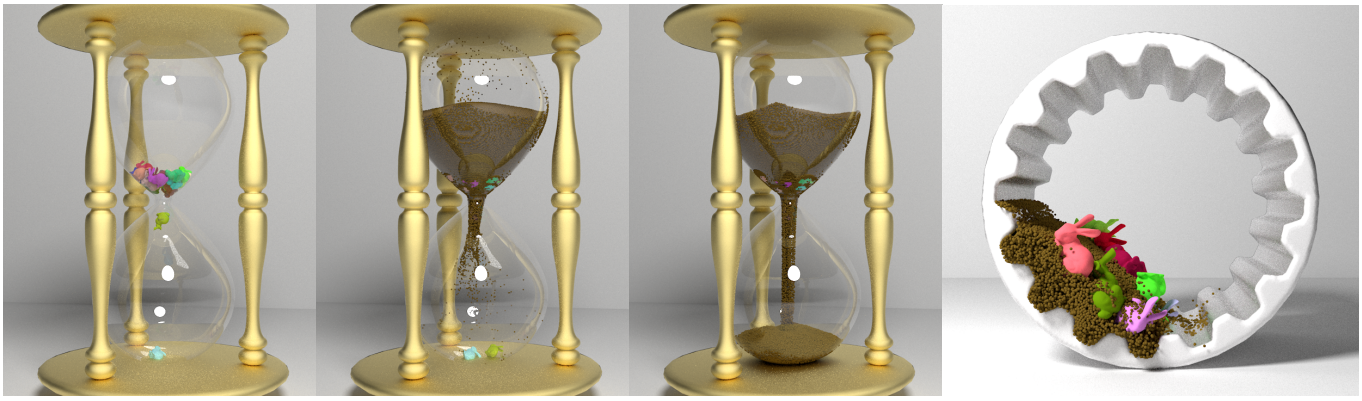


Fig. 1. Our unified solver, *FrictionalMonolith*, enables efficient and robust simulations of granular materials with strong, contact-aware, two-way coupling with rigid bodies. (Left) Granular materials flowing between several bunnies stuck in the middle of an hourglass due to frictional contacts. (Right) A rotating drum mixing granular materials with rigid bunnies.

We propose *FrictionalMonolith*, a monolithic pressure-friction-contact solver for more accurately, robustly, and efficiently simulating two-way interactions of rigid bodies with continuum granular materials or inviscid liquids. By carefully formulating the components of such systems within a single unified minimization problem, our solver can simultaneously handle unilateral incompressibility and implicit integration of friction for the interior of the continuum, frictional contact resolution among the rigid bodies, and mutual force exchanges between the continuum and rigid bodies. Our monolithic approach eliminates various problematic artifacts in existing weakly coupled approaches, including loss of volume in the continuum material, artificial drift and slip of the continuum at solid boundaries, interpenetrations of rigid bodies, and simulation instabilities. To efficiently handle this challenging monolithic minimization problem, we present a customized solver for the resulting quadratically constrained quadratic program that combines elements of staggered projections, augmented Lagrangian methods, inexact projected Newton, and active-set methods. We demonstrate the critical importance of a unified treatment and the effectiveness of our proposed solver in a range of practical scenarios.

CCS Concepts: • **Computing methodologies** → **Physically-based simulation**; **Monolithic coupling**.

Authors' addresses: Tetsuya Takahashi, Adobe, USA, ttakahas@adobe.com; Christopher Batty, University of Waterloo, Canada, christopher.batty@uwaterloo.ca.

0730-0301/2021/12-ART206 $15.00
https://doi.org/10.1145/3478513.3480539

Additional Key Words and Phrases: Fluid simulation, monolithic coupling, friction

ACM Reference Format:
Tetsuya Takahashi and Christopher Batty. 2021. FrictionalMonolith: A Monolithic Optimization-based Approach for Granular Flow with Contact-Aware Rigid-Body Coupling . *ACM Trans. Graph.* 40, 6, Article 206 (December 2021), 20 pages. https://doi.org/10.1145/3478513.3480539

1 INTRODUCTION

The dynamics of granular materials (e.g., sand, sugar, salt, rice, grains, etc.) and their interactions with other objects are critical to a huge range of common physical phenomena: consider sugar being scooped with a tea spoon, pack animals carrying their burdens across a vast desert, grain piling up in a silo, or children playing with toys on a sandy beach. The ability to simulate these phenomena both stably and efficiently is critical for practical applications in various industries, including animation, augmented and virtual reality, robotics, engineering, food processing, and medicine.

Because of the demand for simulating large-scale scenarios, a common approach in the literature is to treat the dynamics of granular materials at the macroscopic scale rather than attempting to resolve the myriad tiny individual grains that they are composed of. This is done by approximating the granular medium as a continuum, allowing one to significantly reduce computational cost and memory usage [Daviet and Bertails-Descoubes 2016; Ihmsen et al. 2013; Klár et al. 2016; Narain et al. 2010; Zhu and Bridson 2005]. Some of these prior approaches, based on this continuum perspective, offer mechanisms for two-way coupling between the granular material and individual rigid bodies, with mutual contact among rigid bodies handled in a distinct step. However, as was recently observed in

the context of viscous fluid / rigid body interaction [Takahashi and Batty 2020], such lagged or weakly coupled approaches can have disastrous effects on simulation quality and stability when the contact graph is non-trivial. For example, fluid-oblivious rigid-body contact resolution can induce severe fluid volume loss when a rigid body ends up overlapping the fluid; conversely, if fluid-solid coupling is computed while ignoring rigid-rigid contacts, rigid bodies can be pushed into each other, leading to numerical explosions. The same fundamental issue arises for granular media, but is made more difficult from a numerical perspective due to their comparatively complex constitutive and contact behavior.

To address this challenge, we propose a novel monolithic treatment of pressure, friction, and contact forces for complex granular-rigid interactions. Our formulation simultaneously considers: unilateral incompressibility and implicit integration of frictional stress for continuum granular media; frictional contact handling among rigid bodies; and pressure and frictional forces between the grains and rigid bodies. Our approach is variational in nature, relying on a unified kinetic energy minimization formulation wherein each physical component contributes terms to the system's total energy, leading to a correct and globally coupled exchange of forces. Furthermore, to overcome artificial anisotropy artifacts induced by the common pyramidal approximation of the friction cone, for both rigid bodies and granular media, we propose to enforce exact friction cone constraints; our monolithic formulation thus becomes a Quadratically Constrained Quadratic Program (QCQP). To efficiently solve this QCQP, we present a customized solver that combines staggered projections, augmented Lagrangian methods, inexact projected Newton, and active-set methods. In summary, our key contributions are

- **A unified pressure-friction solver** to accurately simulate continuum granular materials under unilateral incompressibility and exact friction cone constraints with sub-grid detail.
- **Strong two-way rigid-fluid coupling** to robustly capture two-way interactions of granular materials and individual rigid bodies within our unified pressure-friction solver.
- **A monolithic contact-aware rigid-rigid rigid-fluid coupling method** that improves robustness by simultaneously handling all momentum exchanges among fluids and solids due to pressure, frictional stress, and frictional contact forces.

2 RELATED WORK

2.1 Granular Materials

Given that granular materials consist of many tiny grains, a natural choice is to simulate them as individual particles. This leads to a range of approaches usually referred to as Discrete Element Methods (DEM), which often make use of penalty forces [Bell et al. 2005] or constraints [Macklin et al. 2014; Mazhar et al. 2015]. These approaches offer an impressive level of fidelity and detail, but the simulation costs can quickly become prohibitive as the material volume increases or the grain size shrinks. One partial solution is the use of secondary simulations which can improve visual quality and reduce simulation cost [Alduan et al. 2009]. Continuum-based approaches have instead been proposed to approximate granular flows at a coarser scale. Zhu and Bridson [2005] modeled sand as an incompressible continuum augmented with simplified friction handling on the interior and at boundaries. Narain et al. [2010] further adapted the continuum-based approach of Zhu and Bridson to simulate dry sand by enforcing friction pyramid constraints and unilateral incompressibility using Linear Complementarity Problem (LCP) formulations, allowing grains to freely separate from boundaries and one another. They also incorporated two-way rigid-body coupling, extending the fluid-solid coupling of Batty et al. [2007].

Our granular flow formulation is related to that of Narain et al. [2010], with several key differences. First, we enable exact friction *cone* constraints to eliminate artificial anisotropy induced by pyramidal friction cone approximations. Second, we derive a unified optimization formulation at the continuous level for the coupled pressure and frictional stress, whereas Narain et al. [2010] developed two discrete optimizations. Our formulation enables greater consistency, handling boundaries with both fluid/solid and liquid/air volume fractions (similar to Larionov et al. [2017] for viscous liquids) and yielding a more accurate discretization that prevents oscillation artifacts at free surfaces. Furthermore, our unified formulation enables two-way contact-aware coupling with multiple rigid bodies.

Recently, the material point method (MPM) has been widely used to simulate granular materials as a continuum, e.g., for dry sand [Daviet and Bertails-Descoubes 2016; Han et al. 2019; Jiang et al. 2017, 2020; Klár et al. 2016] and sand-fluid mixtures [Gao et al. 2018; Tampubolon et al. 2017]. The elastoplastic constitutive laws used in these MPM methods are effective in approximating granular flows [Klár et al. 2016; Yue et al. 2018], but they typically adopt a weakly compressible model and use explicit time integration, and thus can suffer from bouncing/compression artifacts and stability issues, in contrast to our strictly incompressible formulation. MPM schemes also often require larger quadratic or cubic stencils to avoid cell-crossing instabilities [Jiang et al. 2016], with some exceptions [Daviet and Bertails-Descoubes 2016; Yue et al. 2018]. To faithfully capture granular dynamics that depend critically on individual grains (e.g., clogging effects), Yue et al. [2018] presented a hybrid solver that combines MPM and DEM.

At the continuous level, our granular flow formulation has connections to the constraint-based MPM formulation of Daviet and Bertails-Descoubes [2016]. A key difference lies in the discretization approach. Their method is based on MPM using a purely colocated grid; the authors note that since they enforce constraints at discrete points (e.g., for unilateral incompressibility), this can lead to volume loss. By contrast, our method uses a hybrid grid that combines the traditional finite-volume-style staggered grid discretization with a colocated stress arrangement. Our approach can thus enforce incompressibility more naturally, without suffering from volume loss, in exchange for slightly larger stencils. Another notable difference is that Daviet and Bertails-Descoubes [2016] formulate the cone constraints as a complementarity problem using the modified Fischer-Burmeister function based on the Jordan algebras [Daviet et al. 2011]. Our method instead employs a variational minimization approach (whose Karush–Kuhn–Tucker (KKT) conditions are equivalent to the complementarity-based formulation [Anitescu and Tasora 2010]), directly addressing the cone constraints with our customized solver. Moreover, their approach does not consider the highly contact-dependent two-way coupled rigid body scenarios targeted by our monolithic framework.

Smoothed particle hydrodynamics (SPH) offers another possible treatment of continuum-based granular materials. For example, Lenaerts and Dutré [2009] adapted the model of Zhu and Bridson [2005] to SPH. Later, Alduán and Otaduy [2011] extended their method with unilateral incompressibility and friction constraints following Narain et al. [2010], and Ihmsen et al. [2013] incorporated a secondary simulation for finer details [Alduan et al. 2009].

Lastly, although they sacrifice intrinsically three-dimensional effects for efficiency, height field approaches have also been proposed for granular simulation, e.g., [Sumner et al. 1999; Zhu et al. 2021]. To improve visual quality with minimal increased cost, Onoue and Nishita [2003] introduced multi-valued height fields, and Zhu and Yang [2010] added 3D effects using surface DEM particles.

2.2 Contact Handling

Velocity-level LCP formulations have been extensively used for contact handling of rigid bodies [Stewart 2000]. While various numerical schemes can be used to solve the LCP, for increasingly complex scenarios with greater numbers of contacts, iterative solvers tend to be preferred because of their scalability and lower memory usage, as compared to direct solvers. In particular, Projected Gauss-Seidel (PGS) has been widely used due to its simplicity and flexibility despite its slow convergence. To improve the efficiency of PGS, various techniques have been presented, e.g., shock propagation [Erleben 2007], dimension reduction in Gauss-Seidel iterations [Silcowitz et al. 2010], mass splitting [Tonge et al. 2012], proximal algorithms [Erleben 2017], and long range constraints [Müller et al. 2017].

For greater efficiency and accuracy, researchers have investigated alternative reformulations and numerical solvers. Renouf and Alart [2005] reformulated the LCP as a QP and solved it using a conjugate projected gradient algorithm. Kaufman et al. [2008] presented the staggered projections approach, formulating the non-penetration and friction constraint problems as separate convex QPs and solving them in alternating fashion to convergence. Later, Le Lidec et al. [2021] solved QCQP friction problems within the staggered projections framework to handle the cone constraints, employing the Alternating Direction Method of Multipliers (ADMM) [Boyd et al. 2011]. Reformulation as a Nonlinear Complementarity Problem (NCP) using the Fischer-Burmeister formulation was also shown to accurately handle friction cone constraints; this form has been solved using a non-smooth Newton method [Silcowitz et al. 2009], non-smooth Gauss-Newton method [Todorov 2010], and non-smooth nonlinear conjugate gradient (NCG) [Silcowitz-Hansen et al. 2010]. Later, Todorov [2011] formulated the frictional contact problem as a constrained minimization problem and solved it using an interior point method with Newton's method. To solve a similar constrained minimization problem, Mazhar et al. [2015] proposed accelerated projected gradient descent (APGD), which exploits Nesterov's method, and demonstrated its efficiency over PGS. To evaluate efficiency, Melanz et al. [2017] and De et al. [2019] compared various first order methods, such as PGS and APGD, and second order methods, such as interior point methods with Newton's method.

Recently, these techniques have been augmented to handle frictional contacts of deformable solids, using the conjugate residual method with dynamic constraint updates [Verschoor and Jalba 2019], non-smooth Newton with geometric stiffness approximations [Macklin et al. 2019], the primal interior point method with incremental potentials [Li et al. 2020a,b], nodal contact assumptions for primal solvers [Li et al. 2018; Ly et al. 2020], ADMM to split elasticity and contact problems [Daviet 2020], and implicit surface representations of smooth contact geometry [Larionov et al. 2021].

2.3 Two-Way Fluid-Solid Coupling

Initial two-way coupling approaches for Eulerian fluids and Lagrangian solids employed weak two-way coupling (e.g., [Guendelman et al. 2005]), alternatingly solving the fluid and solid in isolation, with appropriate boundary condition updates; this approach is generally less robust than strong two-way coupling which simultaneously considers both fluid and solid dynamics (e.g., [Klingner et al. 2006]). While strong coupling is often formulated as a single unified system, iterating weak coupling to convergence is a valid form of strong coupling since it amounts to solving the unified system in block Gauss-Seidel fashion. This approach can converge slowly, although Akbay et al. [2018] presented a technique to accelerate it.

Batty et al. [2007] proposed a grid-based strong two-way pressure coupling method based on a variational principle, later augmented with friction [Narain et al. 2010] and viscosity [Takahashi and Lin 2019]. Robinson-Mosher et al. [2008] presented a different coupling approach for both rigid and deformable solids, including free-slip boundary conditions [Robinson-Mosher et al. 2009]. Later work reformulated this method to ensure symmetric positive definite (SPD) systems [Robinson-Mosher et al. 2011] and to incorporate drag on sub-grid solids [Hyde and Fedkiw 2019]. Takahashi and Batty [2020] developed a monolithic "contact-aware" coupling strategy, unifying rigid body frictional contact handling [Kaufman et al. 2008] with a Stokes-based solver for viscous liquids [Larionov et al. 2017].

Our two-way coupling strategy builds on the granular flow scheme of Narain et al. [2010] and the contact-aware coupling method of Takahashi and Batty [2020]. In contrast to Narain's work, we formulate a single unified formulation for implicit integration and fluid-solid force exchange of pressure and frictional stress under the true friction cone constraints, and further consider monolithic contact-aware coupling with rigid bodies. In contrast to Takahashi's method, we consider granular flow rather than viscous liquids, leading to a more challenging constrained optimization problem.

Variations on the immersed boundary method of Peskin [2002], which treats fluids and solids as a single incompressible medium, have also been used to animate two-way interactions, e.g., the rigid-fluid method [Carlson et al. 2004] or the purely Eulerian coupling method of Teng et al. [2016]. Recently, the immersed boundary method was augmented with a reduced elasticity solver to simulate fluid-solid coupling in real time [Brandt et al. 2019].

MPM schemes provide another natural mechanism for two-way coupling of different materials through the shared underlying grid structure [Ding and Schroeder 2020; Hu et al. 2018; Klár et al. 2016; Yue et al. 2018]. Recently, Fang et al. [2020] proposed using particle quadratures at the interface to handle free-slip boundary conditions. Similarly, a range of SPH methods have been derived to support

two-way coupling, in many cases exploiting a unified particle representation for fluids and solids [Akinci et al. 2012; Band et al. 2018a,b; Becker et al. 2009; Gissler et al. 2019; Koschier and Bender 2017].

3 FUNDAMENTALS

To establish important context and consistent notation, we summarize previous formulations, with some natural extensions, as the starting point for our work. We discuss granular material dynamics (§3.1), frictional contact handling for rigid bodies (§3.2), and strong two-way fluid-solid coupling in the absence of rigid-rigid contacts (§3.3). (For brevity we refer to our granular continuum as a "fluid".)

3.1 Granular Material Dynamics

We model granular materials as a fluid-like continuum based on the Cauchy momentum equation and continuity equation,

$$\rho \frac{D\mathbf{u}}{Dt} = \nabla \cdot \boldsymbol{\sigma} + \mathbf{f}_f \quad \text{and} \quad \frac{d\rho}{dt} + \nabla \cdot (\rho \mathbf{u}) = 0, \tag{1}$$

where t denotes time, $\frac{D}{Dt}$ the material derivative, $\mathbf{u}$ velocity, ρ density, $\boldsymbol{\sigma}$ the symmetric Cauchy stress tensor, and $\mathbf{f}_f$ any external forces. Cauchy stress $\boldsymbol{\sigma}$ can be decomposed into an isotropic mean stress given by the pressure p, and a traceless deviatoric component given by the frictional stress $\mathbf{s}$ ($\mathbf{s}_{xx}, \mathbf{s}_{xy}, \mathbf{s}_{xz}, \mathbf{s}_{yy}, \mathbf{s}_{yz}, \mathbf{s}_{zz}$) with $\mathbf{s}_{xx} + \mathbf{s}_{yy} + \mathbf{s}_{zz} = 0$, i.e., $\boldsymbol{\sigma} = -p\mathbf{I} + \mathbf{s}$ where $\mathbf{I}$ is the identity matrix.

We use the affine particle-in-cell (APIC) framework [Jiang et al. 2015] to handle advection of velocity and density (i.e., particle momentum and mass) and to advance the simulation. To update momentum on the grid, we first apply any external forces, and then apply pressure and friction forces using implicit time integration,

$$\rho^t \left(\frac{\mathbf{u}^{t+\Delta t} - \mathbf{u}^*}{\Delta t} \right) = -\nabla p^{t+\Delta t} + \nabla \cdot \mathbf{s}^{t+\Delta t}, \tag{2}$$

where $\mathbf{u}^*$ and $\mathbf{u}^{t+\Delta t}$ denote the velocities before and after implicit integration, respectively, and Δt denotes the time step size. This stage will be the primary focus of our work.

3.1.1 Staggered Projections. To address the integration of pressure and friction forces in (2), one can split this task into two subproblems, and combine them using Staggered Projections (SP) [Kaufman et al. 2008; Narain et al. 2010]. Specifically, one can first compute frictional stress $\mathbf{s}$ to update $\mathbf{u}^*$, then use the updated $\mathbf{u}^*$ to compute pressure $\mathbf{p}$ and update $\mathbf{u}^*$ again, and alternate these steps until convergence.

3.1.2 Pressure Formulation. The compression and expansion behaviors of non-cohesive dry granular flows (or splashing liquids with wall separation) can be modeled with three conditions relating pressure and density:

1. density ρ should not exceed the rest density ρ_0: $\rho \leq \rho_0$,
2. pressure is non-negative to prevent attraction: $0 \leq p$,
3. pressure acts (i.e., is non-zero) only when the density equals the rest density: $p(1 - \rho/\rho_0) = 0$.

These conditions can be encoded as a discrete LCP (or equivalent QP), as prior work has shown [Gerszewski and Bargteil 2013; Narain et al. 2010]. For consistency during discretization, we instead formulate a continuous version over the simulation domain Ω, introducing constraint stabilization (with a parameter β_f) that acts to correct densities that have exceeded ρ_0:

$$\mathbf{p} = \underset{0 \leq \mathbf{p}}{\arg\min} \int_\Omega \left(\frac{1}{2}\rho \left\| \mathbf{u}^* - \frac{\Delta t}{\rho} \nabla \mathbf{p} \right\|_2^2 + \Delta t \mathbf{p} \left(\frac{\beta_f}{\Delta t} \min\left(1 - \frac{\rho}{\rho_0}, 0\right) \right) \right) d\Omega. \tag{3}$$

The min operator ensures that the stabilization effect is active only when density $\rho > \rho_0$ (i.e., due to numerical advection error) since $\min(1 - \frac{\rho}{\rho_0}, 0)$ is non-positive. This yields a source term that induces (only) expansion where needed, analogous to rigid body stabilization that pushes overlapping solids apart [Anitescu and Hart 2004; Cline and Pai 2003]. Equation 3 extends the variational formulation of Batty et al. [2007] to take our three required conditions into account, along with stabilization, and naturally supports wall-separation.

3.1.3 Friction Formulation. The frictional stress must satisfy a yield condition. We choose the popular Drucker-Prager yield criterion [Klár et al. 2016; Narain et al. 2010],

$$\|\mathbf{s}\|_F \leq \alpha \mathbf{p}, \tag{4}$$

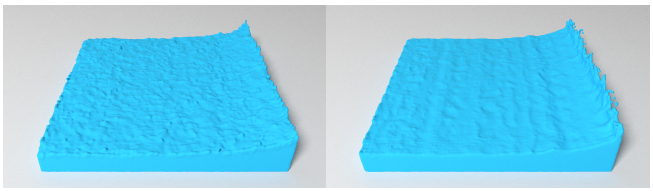
where $\|\cdot\|_F$ denotes the Frobenius norm and α denotes the friction coefficient, related to the angle of repose θ by $\alpha = \sqrt{2}\sin\theta$.

To model friction within our continuum granular material, one can formulate a minimization problem subject to the Drucker-Prager yield criterion (4) using the maximum dissipation principle:

$$\mathbf{s} = \underset{\|\mathbf{s}\|_F \leq \alpha \mathbf{p}}{\arg\min} \int_\Omega \left(\frac{1}{2}\rho \left\| \mathbf{u}^* + \frac{\Delta t}{\rho} \nabla \cdot \mathbf{s} \right\|_2^2 \right) d\Omega. \tag{5}$$

A possible simplification of this problem, suggested by Narain et al. [2010], is achieved by approximating the friction cone with a pyramid, applying pressure bounds to each frictional stress component separately as box constraints, i.e., $-\alpha\mathbf{p} \leq \mathbf{s}_e \leq \alpha\mathbf{p}$ (where $e \in xx, xy, xz, yy, yz, zz$) . This simplification further allows straightforward splitting of the problem into smaller componentwise subproblems using SP. In addition, due to the traceless property of $\mathbf{s}$, we can eliminate one subproblem on the diagonal components, e.g., via $\mathbf{s}_{zz} = -(\mathbf{s}_{xx} + \mathbf{s}_{yy})$; however, this introduces additional axis-dependent artificial anisotropy since the pressure bound cannot be enforced on the eliminated component, leading to $-2\alpha\mathbf{p} \leq \mathbf{s}_{zz} \leq 2\alpha\mathbf{p}$ if $\mathbf{s}_{zz}$ is eliminated.

3.1.4 Spatial Discretization with Sub-Grid Details. For spatial discretization, we employ a variational finite difference approach using the staggered grid [Goktekin et al. 2004] and volume fractions [Larionov et al. 2017]. Given a signed distance function to represent fluid/solid domains, we compute dimensionless volume fractions of fluids and solids as diagonal matrices $\mathbf{W}_F^u$ and $\mathbf{W}_S^u$, respectively, with a range of $[0, 1]$. Here, the superscript and subscript indicate weight matrices for velocity variables u, and fluid domain F or solid domain S, respectively. Similarly, we define $\mathbf{W}_F^p$ and $\mathbf{W}_S^p$ for pressure, and $\mathbf{W}_F^s$ and $\mathbf{W}_S^s$ for frictional stress. Likewise, using signed distance functions (approximated from particles [Zhu and Bridson 2005]) to represent liquid/air (L/A) domains, we define volume fractions for pressure $\mathbf{W}_L^p$ and $\mathbf{W}_A^p$, velocity $\mathbf{W}_L^u$ and $\mathbf{W}_A^u$, and frictional stress $\mathbf{W}_L^s$ and $\mathbf{W}_A^s$. With these volume fractions, the minimization



(a) Voxel-based (b) Volume-based (ours)

Fig. 2. An initially trapezoid-shaped inviscid liquid simulated with voxel- and volume-based computations for liquid/air volume fractions (see §5.1.2). The voxel-based approach (a) suffers from non-physical oscillatory behaviors due to voxelization, while our volume-based approach (b) generates smoother waves by more accurately resolving the surface position.

problems on pressure (3) and frictional stress (5) can be independently discretized for solid boundaries and free surfaces, leading to a combined form of

$$\mathbf{p} = \underset{0 \le \mathbf{p}}{\arg\min}\, E_{f,p}(\mathbf{p}) \quad \text{and} \quad \mathbf{s} = \underset{\|\mathbf{s}\|_F \le \alpha \mathbf{p}}{\arg\min}\, E_{f,s}(\mathbf{s})$$

$$E_{f,p}(\mathbf{p}) = \frac{1}{2}\left\|\mathbf{u}^* - \Delta t(\mathbf{P}^u\mathbf{W}_L^u)^{-1}\mathbf{G}\mathbf{W}_L^p\mathbf{p}\right\|_{\mathbf{M}_f}^2 + \Delta t\mathbf{p}^T\mathbf{W}^p\hat{\boldsymbol{\phi}}, \tag{6}$$

$$E_{f,s}(\mathbf{s}) = \frac{1}{2}\left\|\mathbf{u}^* - \Delta t(\mathbf{P}^u\mathbf{W}_L^u)^{-1}\mathbf{D}\mathbf{W}_L^s\mathbf{s}\right\|_{\mathbf{M}_f}^2, \tag{7}$$

where the diagonal matrix $\mathbf{P}$ denotes fluid density with superscripts to represent its grid location (u or p), $\mathbf{G}$ is the discrete (scalar) gradient operator, $\mathbf{D}$ is the discrete (tensor) divergence operator, and the matrix-weighted vector norm $\|\cdot\|_{\mathbf{W}}$ is defined such that $\|\mathbf{y}\|_{\mathbf{W}} = \sqrt{\mathbf{y}^T\mathbf{W}\mathbf{y}} = \left\|\mathbf{W}^{\frac{1}{2}}\mathbf{y}\right\|_2$. We have also made use of the following diagonal matrix definitions: $\mathbf{M}_f = \mathbf{P}^u\mathbf{W}^u$ (where the units of $\mathbf{M}_f$ are kg/m^3), $\mathbf{W}^u = \mathbf{W}_F^u\mathbf{W}_L^u$, and $\mathbf{W}^p = \mathbf{W}_F^p\mathbf{W}_L^p$. In addition, $\hat{\boldsymbol{\phi}}$ is a density constraint vector defined such that $\hat{\boldsymbol{\phi}}_i = \frac{\beta_f}{\Delta t}\min\left(1 - \mathbf{P}_i^p(\mathbf{P}_{0,i}^p)^{-1}, 0\right)$ for grid index i, and the diagonal matrix $\mathbf{P}_0^p$ denotes the fluid rest density with its grid location (p).

3.1.5 Discussion. The fundamental formulations summarized above build directly on the prior work [Narain et al. 2010], but we emphasize several key differences. First, while Narain et al. [2010] mitigated the ill-conditioned stress system due to a variable density formulation by rescaling the stress system with the occupation ratio of each cell, we only include voxels with fluid density exceeding a threshold in the optimization to maintain consistency for pressure and stress systems (see §4.1.1). Second, to more accurately account for free surface locations, we incorporate liquid/air volume fractions into our formulation (see Figure 2). Third, Narain et al. [2010] derived discretized QP formulations for pressure and stress whereas we derived analogous continuous QPs; this allows us to consistently discretize the formulations with both fluid/solid and liquid/air volume fractions, using variational finite differences [Larionov et al. 2017]. In addition, unlike the direct density correction of Narain et al. [2010], we will instead correct particle positions to adjust the density, ensuring consistency between the density and particle positions (see §4.1.2). Here, we address only the fluid compression case ($\rho_0 < \rho$) to emphasize the analogy of pressure for fluids in (3) and normal forces for rigid body contacts in (8).

3.2 Frictional Contact Dynamics for Rigid Bodies

For consistency with the fluid solver, we formulate the resolution of frictional contacts at the velocity-level [Stewart 2000; Tan et al. 2012]; this allows us to hold the position and orientation of rigid bodies fixed and avoid repeated evaluations of volume fractions within a time step. Given rigid bodies with generalized positions $\mathbf{x}$, generalized velocities $\mathbf{v}$ ($\mathbf{v}^*$ denotes velocities before the contact), mass matrix $\mathbf{M}_r$, contact basis matrix $\mathbf{B}(= [\mathbf{B}_{\boldsymbol{\lambda}}, \mathbf{B}_{\mathbf{z}}])$ of size $6N_r \times 3N_c$ in 3D (where N_r and N_c denote the number of rigid bodies and contacts, respectively), friction coefficient $\boldsymbol{\eta}$, and contact index j, resolving rigid body contacts requires satisfying the following constraints due to the Signorini-Coulomb contact law:

1. normal contact forces $\boldsymbol{\lambda}$ are non-negative: $0 \le \boldsymbol{\lambda}^{t+\Delta t}$,
2. relative velocities of colliding rigid bodies are non-penetrative: $-\mathbf{B}^T\mathbf{v}^* \le \mathbf{B}^T\mathbf{v}^{t+\Delta t}$,
3. tangential (frictional) contact forces $\mathbf{z}$ lie within their respective friction cones: $\left\|\mathbf{z}_j^{t+\Delta t}\right\|_2 \le \boldsymbol{\eta}_j\boldsymbol{\lambda}_j^{t+\Delta t}$.

Incorporating position-level stabilization with a collision constraint vector $\boldsymbol{\psi}(\mathbf{x}) \ge 0$ [Anitescu and Hart 2004; Cline and Pai 2003], we can formulate contact handling as a constrained minimization problem [Todorov 2011]:

$$\boldsymbol{\lambda}, \mathbf{z} = \underset{\substack{0 \le \boldsymbol{\lambda}, \\ \|\mathbf{z}_j\|_2 \le \boldsymbol{\eta}_j\boldsymbol{\lambda}_j}}{\arg\min} \left(\frac{1}{2}\left\|\mathbf{v}^* + \Delta t\mathbf{M}_r^{-1}\left(\mathbf{B}_{\boldsymbol{\lambda}}\boldsymbol{\lambda} + \mathbf{B}_{\mathbf{z}}\mathbf{z}\right)\right\|_{\mathbf{M}_r}^2 + \Delta t\boldsymbol{\lambda}^T\hat{\boldsymbol{\psi}}\right), \tag{8}$$

where $\hat{\boldsymbol{\psi}} = \frac{\beta_r}{\Delta t}\min(\boldsymbol{\psi}(\mathbf{x}), 0)$, and β_r is a stabilization parameter for position correction. The second (stabilization) term on the right hand side of (8) behaves in much the same manner as the density correction for fluids in (3). The minimization formulation (8) also accounts for the maximal dissipation principle, and supports the take-off, stick, and slip cases of frictional contact [Daviet et al. 2011; Li et al. 2018; Ly et al. 2020].

3.3 Two-Way Fluid-Solid Coupling

We review force exchange mechanisms for strong two-way fluid-solid coupling (without mutual rigid body contacts) via pressure [Batty et al. 2007] and frictional stress [Narain et al. 2010], including augmentations with both fluid/solid and liquid/air volume fractions [Larionov et al. 2017], mass scaling for dimensional consistency, and consistent force exchange formulations [Takahashi and Batty 2020].

3.3.1 Pressure. Considering the fluid pressure force on rigid bodies, we can update the rigid body velocities using volume fractions as

$$\mathbf{v}^{t+\Delta t} = \mathbf{v}^* + \Delta t\mathbf{M}_r^{-1}\mathbf{F}_p\mathbf{W}_L^p\mathbf{p}^{t+\Delta t}, \tag{9}$$

where $\mathbf{F}_p$ denotes a linear operator integrating the fluid pressure on the rigid body surface(s) [Bridson 2015]. Scaling the solid mass with $\tau = 1/(\Delta x)^3$ (Δx: cell width) to ensure consistent physical dimensions of the coupled systems [Takahashi and Batty 2020], we can extend the minimization problem of (6) to get:

$$\mathbf{p} = \underset{0 \le \mathbf{p}}{\arg\min}\left(E_{f,p}(\mathbf{p}) + \frac{1}{2}\left\|\mathbf{v}^* + \Delta t(\tau\mathbf{M}_r)^{-1}\mathbf{F}_p\mathbf{W}_L^p\mathbf{p}\right\|_{\tau\mathbf{M}_r}^2\right). \tag{10}$$

3.3.2 Friction. Similar to pressure forces, friction forces can be applied to rigid bodies using

$$\mathbf{v}^{t+\Delta t} = \mathbf{v}^* + \Delta t \mathbf{M}_r^{-1} \mathbf{F}_s \mathbf{W}_L^s \mathbf{s}^{t+\Delta t}, \tag{11}$$

where $\mathbf{F}_s$ denotes a linear function to integrate frictional stress (analogous to $\mathbf{F}_p$), ensuring consistent force exchange [Takahashi and Batty 2020], and one can formulate a corresponding minimization problem extending (7):

$$\mathbf{s} = \underset{\|\mathbf{s}\|_F \le \alpha \mathbf{p}}{\arg\min} \left(E_{f,s}(\mathbf{s}) + \frac{1}{2} \left\| \mathbf{v}^* + \Delta t (\tau \mathbf{M}_r)^{-1} \mathbf{F}_s \mathbf{W}_L^s \mathbf{s} \right\|_{\tau \mathbf{M}_r}^2 \right). \tag{12}$$

4 FRICTIONALMONOLITH

To eliminate a range of serious artifacts (e.g., loss of fluid volume, artificial drift of granular materials, and simulation instabilities), we propose a novel unified treatment of pressure, frictional stress, and frictional contact handling for fluids and rigid bodies by formulating a single monolithic constrained minimization problem:

$$\mathbf{s}, \mathbf{p}, \boldsymbol{\lambda}, \mathbf{z} = \underset{\substack{0 \le \mathbf{p},\ 0 \le \mathbf{c}(\mathbf{s},\mathbf{p}), \\ 0 \le \boldsymbol{\lambda},\ 0 \le \mathbf{d}(\boldsymbol{\lambda},\mathbf{z})}}{\arg\min} E(\mathbf{s}, \mathbf{p}, \boldsymbol{\lambda}, \mathbf{z}), \tag{13}$$

$$E(\mathbf{s}, \mathbf{p}, \boldsymbol{\lambda}, \mathbf{z}) = E_f(\mathbf{s}, \mathbf{p}) + E_r(\mathbf{s}, \mathbf{p}, \boldsymbol{\lambda}, \mathbf{z}). \tag{14}$$

The terms $E_f(\mathbf{s}, \mathbf{p})$ and $E_r(\mathbf{s}, \mathbf{p}, \boldsymbol{\lambda}, \mathbf{z})$ are the objective functions for fluids and rigid bodies, respectively, and $\mathbf{c}(\mathbf{s}, \mathbf{p})$ and $\mathbf{d}(\boldsymbol{\lambda}, \mathbf{z})$ denote the dimensionless friction cone constraint functions for fluids and rigid bodies, respectively. We will define each of these terms explicitly in the sections that follow, assembling and further extending the formulations described in §3.

4.1 Monolithic Granular Material Formulation

Given pressure and friction forces, the fluid velocity update is

$$\mathbf{u}^{t+\Delta t} = \mathbf{u}^* - \Delta t (\mathbf{P}^u \mathbf{W}_L^u)^{-1} \left(\hat{\mathbf{G}} \mathbf{p}^{t+\Delta t} + \hat{\mathbf{D}} \mathbf{s}^{t+\Delta t} \right), \tag{15}$$

where $\hat{\mathbf{G}} = \mathbf{G}\mathbf{W}_L^p$ and $\hat{\mathbf{D}} = \mathbf{D}\mathbf{W}_L^s$. Thus, we can formulate an objective function as

$$E_f(\mathbf{s}, \mathbf{p}) = \frac{1}{2} \left\| \mathbf{u}^* - \Delta t (\mathbf{P}^u \mathbf{W}_L^u)^{-1} \left(\hat{\mathbf{G}} \mathbf{p} + \hat{\mathbf{D}} \mathbf{s} \right) \right\|_{\mathbf{M}_f}^2. \tag{16}$$

This variational minimization formulation seeks pressure and frictional stress fields that maximally dissipate kinetic energy (similar to perfectly inelastic collisions [Bridson 2015]), and thus must be dissipative without increasing energy to ensure stability of the entire system. Since the density correction term $\Delta t \mathbf{p}^T \mathbf{W}^p \hat{\boldsymbol{\phi}}$ in (6) can inject energy into the system, causing stability issues [Kugelstadt et al. 2019], we instead exclude this term from (16), replacing it with particle position correction (§4.1.2).

4.1.1 Density Computation. To account for the material distributions, we use variable densities (along with the fluid/solid and liquid/air volume fractions) for the fluid density defined at the velocity locations $\mathbf{P}^u$ within the APIC framework [Jiang et al. 2015]. The variable density formulation is known to give rise to too small densities that cause simulation and numerical stability issues [Bridson 2015; Gerszewski and Bargteil 2013; Narain et al. 2010] because the density plays a role in the fluid velocity update (15) and the objective function (16). While Narain et al. [2010] rescaled the stress system using volume occupancy to mitigate the issue, this approach introduces inconsistency between the computations for pressure and stress. Thus, similar to [Bridson 2015; Gerszewski and Bargteil 2013], we take into account only pressure and stress samples where the computed density exceeds a threshold ρ_t (we set $\rho_t = 0.4\rho_0$).

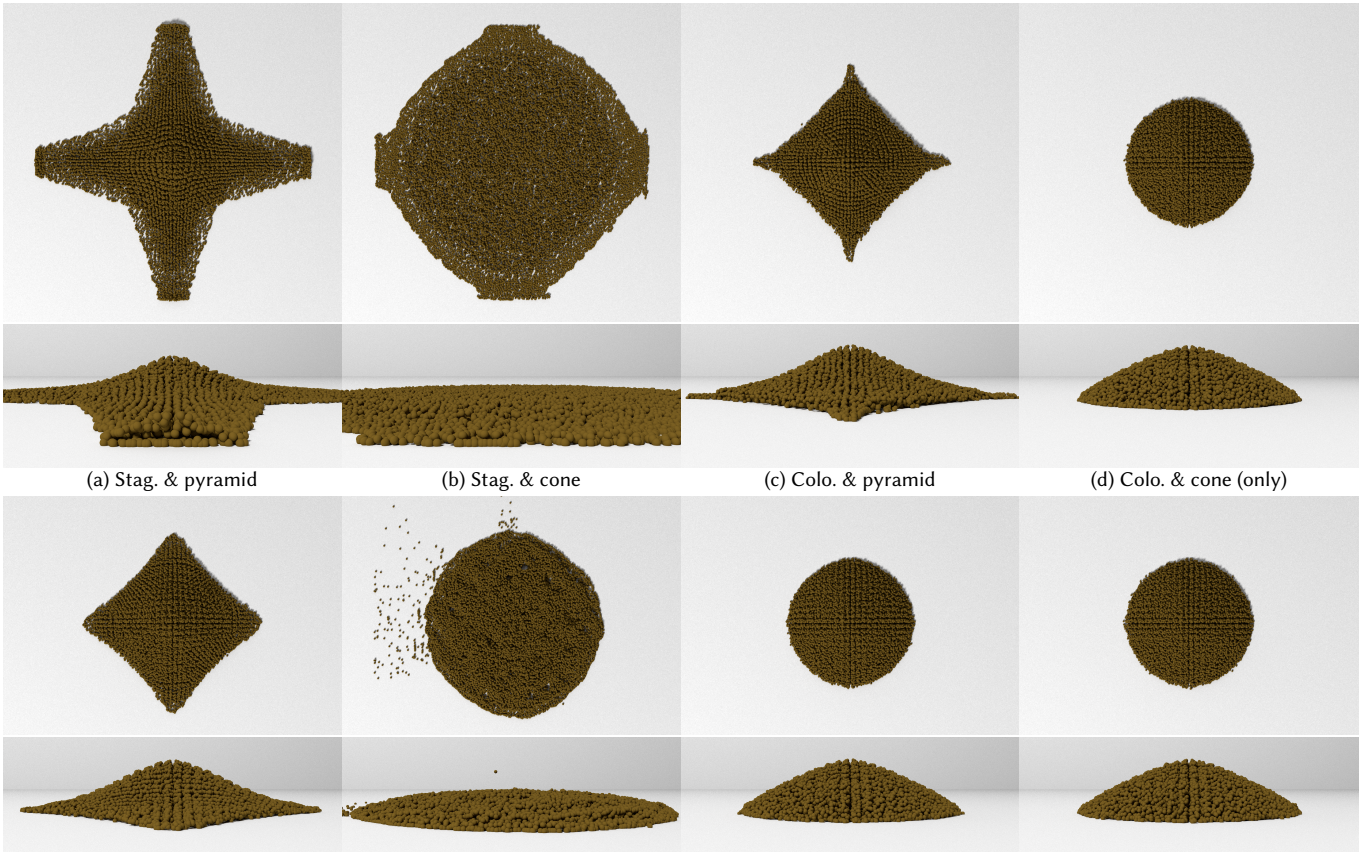
To evaluate the densities at pressure and stress samples, we compute and use the fluid densities defined at the cell centers $\mathbf{P}^p$, interpolating where necessary. Unlike $\mathbf{P}^u$ (on faces), since $\mathbf{P}^p$ is used to evaluate how much fluid occupies each cell, it is necessary to incorporate the density contributed by rigid bodies, distinguishing solid boundaries and free surfaces (particles cannot be both inside rigid bodies and in the air domain). Gerszewski and Bargteil [2013] addressed this issue with additional particles sampled within the solid objects at an extra cost. In our work, we instead reuse the already computed solid volume fraction $\mathbf{W}_S^p$, calculating the density contributed by the solid as $\mathbf{W}_S^p \mathbf{P}_0^p$.

4.1.2 Density Correction. Due to gradually accumulated numerical error, particle distributions can become non-uniform (i.e., violating the density constraints) after advection even under divergence-free velocities. Although Narain et al. [2010] and Gerszewski and Bargteil [2013] directly corrected the density with pseudo-pressures as a post-stabilization, this approach causes a discrepancy between the corrected density and the actual particle distribution. Taking a similar stabilization approach, Kugelstadt et al. [2019] corrected particle positions to fix the density, avoiding the discrepancy and stability issues. However, we experimentally observed that grid-based density correction cannot strictly enforce uniform particle distributions at a sub-grid level [Kugelstadt et al. 2019]. We also found that grid-based density correction introduces artificial drift for granular materials. Therefore, we perform particle-level density correction based on SPH at each frame [Macklin et al. 2014; Takahashi and Lin 2019].

4.1.3 Cone Constraints. While Narain et al. [2010] approximated the friction cone as a pyramid using box (or plane) constraints, this approach leads to artificial anisotropy [Daviet and Bertails-Descoubes 2016], as also observed for solid frictional contacts [Daviet et al. 2011; Le Lidec et al. 2021; Li et al. 2018]. Narain et al. [2010] suggested that adopting more plane constraints can improve the approximation accuracy, but achieving sufficient accuracy requires significantly increasing the number of plane constraints, with a correspondingly large increase in computational cost. As such, we prefer to employ the true cone constraints.

We impose the true cone constraints given by (4) at the center of each cell. While off-diagonal stress variables ($\mathbf{s}_{xy}, \mathbf{s}_{xz}, \mathbf{s}_{yz}$) have been traditionally defined on the cell edges based on finite differencing of face-based quantities [Goktekin et al. 2004; Narain et al. 2010], this staggered arrangement makes it unwieldy to enforce the cone constraints, unless we interpolate $\mathbf{s}_{xy}, \mathbf{s}_{xz}, \mathbf{s}_{yz}$ to the cell center. However, we have found that such an approach leads to smaller and smaller off-diagonal stress values because these stress variables are shared via interpolation and constrained independently by multiple distinct cone constraints; the result is an undesired reduction in the observed friction forces (see Figure 3(b)).

We instead take essentially the reverse approach. We colocate the off-diagonal stress components $\mathbf{s}_{xy}, \mathbf{s}_{xz}, \mathbf{s}_{yz}$ at cell centers, and for each cell with index i, define an associated dimensionless cone



(a) Stag. & pyramid (b) Stag. & cone (c) Colo. & pyramid (d) Colo. & cone (only)

(e) Colo. & cone (tight) (f) Colo. & cone (ls) (g) Colo. & cone (4iter) (h) Colo. & cone (ours)

Fig. 3. A sand column released on the ground, simulated with various solver options (see §5.2.2 for details). Stag. & pyramid generates granular flow with artificial anisotropy due to the pyramid constraints although the flow is stable. Stag. & cone fails to enforce the cone constraints, generating unstable and anisotropic behaviors. Colo. & pyramid generates stable but anisotropic flow due to the pyramid constraints. Colo. & cone (only) generates stable granular flow without suffering from the artificial anisotropy at convergence. Colo. & cone (tight) generates stable but anistropic behaviors due to the too tight pyramid constraints. Colo. & cone (ls) fails to generate stable granular flows due to failures of line search. Both Colo. & cone (4iter) and Colo. & cone (ours) generate stable granular flow without the artificial anisotropy while ours is 2.3× faster.

constraint $\mathbf{c}_i(\mathbf{s}, \mathbf{p})$. Thus each stress sample is touched by only one constraint. Our cone constraint is formed by squaring both sides of (4) and rearranging to get

$$\mathbf{c}_i(\mathbf{s}, \mathbf{p}) = \frac{1}{2}\left(1 - \frac{\mathbf{s}_{xx,i}^2 + \mathbf{s}_{yy,i}^2 + \mathbf{s}_{zz,i}^2 + 2\left(\mathbf{s}_{xy,i}^2 + \mathbf{s}_{xz,i}^2 + \mathbf{s}_{yz,i}^2\right)}{\alpha^2 \mathbf{p}_i^2}\right). \tag{17}$$

Then, to apply the frictional stress back onto the fluid, we first map the colocated off-diagonal stresses to the staggered arrangement before employing the traditional finite-difference approach. Specifically, we introduce a change-of-basis (or transition) matrix $\mathbf{T} = (\mathbf{I}^T, \mathbf{T}_{xy}^T, \mathbf{T}_{xz}^T, \mathbf{I}^T, \mathbf{T}_{yz}^T, \mathbf{I}^T)^T$ which interpolates the $\mathbf{s}_{xy}, \mathbf{s}_{xz}, \mathbf{s}_{yz}$ components defined at cell centers to their corresponding cell edge locations, treating inactive stress variables (e.g., outside of the liquid domain) as having a value of zero. The finite difference operator applying stress after mapping with $\mathbf{T}$ is then redefined as $\hat{\mathbf{D}} = \mathbf{D}\mathbf{W}_L^s\mathbf{T}$.

4.1.4 Stress Reduction. Using the traceless property of $\mathbf{s}$ under the divergence-free constraint, we can reduce the system size by eliminating one of the diagonal components in $\mathbf{s}$. In practice, we eliminate $\mathbf{s}_{yy}$ via $\mathbf{s}_{yy} = -(\mathbf{s}_{xx} + \mathbf{s}_{zz})$ and modify $\mathbf{W}_F^s, \mathbf{W}_L^s, \mathbf{D}, \mathbf{F}_s$, and $\mathbf{T}$ to account for the eliminated $\mathbf{s}_{yy}$. The cone constraint is thus redefined as

$$\mathbf{c}_i(\mathbf{s}, \mathbf{p}) = \frac{1}{2}\left(1 - 2\frac{\mathbf{s}_{xx,i}^2 + \mathbf{s}_{zz,i}^2 + \mathbf{s}_{xx,i}\mathbf{s}_{zz,i} + \mathbf{s}_{xy,i}^2 + \mathbf{s}_{xz,i}^2 + \mathbf{s}_{yz,i}^2}{\alpha^2 \mathbf{p}_i^2}\right). \tag{18}$$

This differs from the approach of Narain et al. [2010] since the reduced system itself needs to be reformulated taking into account the contributions of the removed stress. By contrast, Narain et al. simply substitute $\mathbf{s}_{zz} = -(\mathbf{s}_{xx} + \mathbf{s}_{yy})$ after $\mathbf{s}_{xx}$ and $\mathbf{s}_{yy}$ have been determined. Our stress reduction also differs from the prior work on viscous liquids [Larionov et al. 2017; Takahashi and Batty 2020]: since $\mathbf{s}$ is independent of $\mathbf{u}$ (while $\mathbf{u}$ depends on $\mathbf{s}$ in (2)), unlike the case of viscous fluids, we can embed the influence of the stress

reduction within $\mathbf{D}$ and $\mathbf{F}_s$ without introducing additional auxiliary matrices. An alternative, similar to the approach of Daviet and Bertails-Descoubes [2016], would be to replace $\mathbf{s}_{xx}, \mathbf{s}_{yy}, \mathbf{s}_{zz}$ with new stress variables $\mathbf{s}_a, \mathbf{s}_b$ via $\mathbf{s}_{xx} = \mathbf{s}_a + \mathbf{s}_b$, $\mathbf{s}_{yy} = \mathbf{s}_a - \mathbf{s}_b$, and $\mathbf{s}_{zz} = -2\mathbf{s}_a$, making the Hessian of the cone constraints slightly sparser. However, since this approach makes $\mathbf{W}_F^s, \mathbf{W}_L^s, \mathbf{D}$ and $\mathbf{F}_s$, much denser with significant modifications to the original system, we prefer our stress reduction method.

4.2 Monolithic Rigid Body Formulation

Considering all relevant forces for rigid bodies, their full velocity update can be written as

$$\mathbf{v}^{t+\Delta t} = \mathbf{v}^* + \Delta t(\tau \mathbf{M}_r)^{-1}\left(\hat{\mathbf{F}}_p \mathbf{p}^{t+\Delta t} + \hat{\mathbf{F}}_s \mathbf{s}^{t+\Delta t} + \mathbf{B}_{\boldsymbol{\lambda}} \tau \boldsymbol{\lambda}^{t+\Delta t} + \mathbf{B}_{\mathbf{z}} \tau \mathbf{z}^{t+\Delta t}\right), \tag{19}$$

where $\hat{\mathbf{F}}_p = \mathbf{F}_p \mathbf{W}_L^p$ and $\hat{\mathbf{F}}_s = \mathbf{F}_s \mathbf{W}_L^s \mathbf{T}$. Similarly, we can formulate an objective function for rigid bodies as

$$E_r(\mathbf{s}, \mathbf{p}, \boldsymbol{\lambda}, \mathbf{z}) = \frac{1}{2}\left\| \mathbf{v}^* + \Delta t(\tau \mathbf{M}_r)^{-1}\left(\hat{\mathbf{F}}_p \mathbf{p} + \hat{\mathbf{F}}_s \mathbf{s} + \mathbf{B}_{\boldsymbol{\lambda}} \tau \boldsymbol{\lambda} + \mathbf{B}_{\mathbf{z}} \tau \mathbf{z}\right)\right\|_{\tau \mathbf{M}_r}^2. \tag{20}$$

As we noted for fluids, position-level stabilization within the velocity-level formulation can inject extra energy into the system, causing stability issues (e.g., popping and jittering of rigid bodies) and negatively affecting the convergence of optimization solvers. We therefore exclude the stabilization term from (20). Instead, we employ post-stabilization [Cline and Pai 2003] to address position-level penetrations due to linearization and numerical errors.

Since each friction cone is independent of the other contacts, we can naturally define the cone constraint $\mathbf{d}_j(\boldsymbol{\lambda}, \mathbf{z})$ as

$$\mathbf{d}_j(\boldsymbol{\lambda}, \mathbf{z}) = \frac{1}{2}\left(1 - \frac{\|\mathbf{z}_j\|_2^2}{\eta_j^2 \boldsymbol{\lambda}_j^2}\right). \tag{21}$$

4.3 Minimization with Our Customized Solver

Having described our monolithic formulation and its discretization, we proceed to the design of an appropriate numerical solver. One could directly apply a general purpose off-the-shelf solver (e.g., sequential quadratic programming (SQP) or interior point methods) to the full QCQP (13) to find a local minimum. However, Kaufman et al. [2009; 2008] showed that the resulting local minima often violate either the Signorini conditions or the maximum dissipation principle, yielding unsatisfactory behavior; yet solvers possessing a particular fixed-point property *are* robust and can consistently produce superior solutions in a wide range of frictional scenarios. More generally, it is often more efficient to employ a specialized solver that utilizes knowledge of the problem, compared to general gradient-based optimizers (e.g., gradient descent and L-BFGS) applied to solve subproblems derived from (13). We therefore propose a customized optimization solver specifically designed to address our QCQP. Since there are multiple minimizers satisfying $\nabla E(\mathbf{s}, \mathbf{p}, \boldsymbol{\lambda}, \mathbf{z}) = 0$, we seek an appropriate local minimum.

4.3.1 Staggered Projections. The key to reasonable and consistent contact solutions is the fixed-point property described by Kaufman [2009], so we adapt the staggered projections approach [Kaufman et al. 2008; Narain et al. 2010]. Specifically, we decompose (13) into two subproblems, a QCQP on $(\mathbf{s}, \mathbf{z})$ with cone constraints and a QP on $(\mathbf{p}, \boldsymbol{\lambda})$ with simpler non-negativity (i.e., box) constraints:

$$\mathbf{s}^{k+1}, \mathbf{z}^{k+1} = \underset{0 \le \mathbf{c}(\mathbf{s}, \mathbf{p}^k),\ 0 \le \mathbf{d}(\boldsymbol{\lambda}^k, \mathbf{z})}{\arg\min} E(\mathbf{s}, \mathbf{p}^k, \boldsymbol{\lambda}^k, \mathbf{z}), \tag{22}$$

$$\mathbf{p}^{k+1}, \boldsymbol{\lambda}^{k+1} = \underset{0 \le \mathbf{p},\ 0 \le \boldsymbol{\lambda}}{\arg\min} E(\mathbf{s}^{k+1}, \mathbf{p}, \boldsymbol{\lambda}, \mathbf{z}^{k+1}). \tag{23}$$

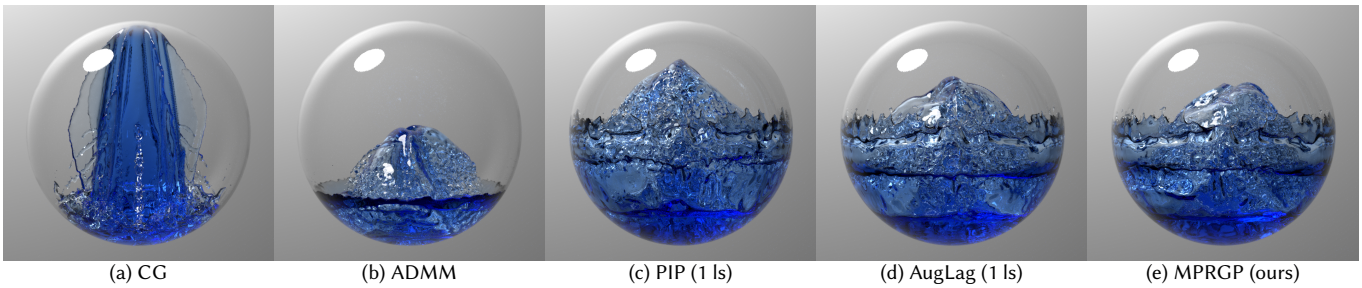
We alternate solving these two problems, with k denoting the iteration index for these staggered projections. We intentionally solve for $(\mathbf{s}, \mathbf{z})$ first within the cone constraints with fixed upper bounds dictated by $(\mathbf{p}^k, \boldsymbol{\lambda}^k)$, and conclude the iteration with updated $(\mathbf{p}^{k+1}, \boldsymbol{\lambda}^{k+1})$; we do so because $(\mathbf{p}, \boldsymbol{\lambda})$ are generally more critical for stability of the entire system, and all the residuals can be shifted to the less critical components $(\mathbf{s}, \mathbf{z})$ if early termination is desired. These two subproblems are non-expansive (typically contractive) projections, and thus alternating iterations of these projections converge to the solution of (13) [Kaufman et al. 2008; Narain et al. 2010]. In the following, we first explain how to address the box-constrained QP subproblem (23) before discussing solvers for the QCQP subproblem (22), since additional box-constrained QP subproblems also arise within the QCQP solver.

4.3.2 Box-Constrained QP Solver for $(\mathbf{p}, \boldsymbol{\lambda})$. The objective function for the box-constrained QP subproblem (23) can be rewritten as

$$E(\mathbf{s}^{k+1}, \mathbf{p}, \boldsymbol{\lambda}, \mathbf{z}^{k+1}) = \frac{1}{2}\begin{bmatrix}\mathbf{p}^T, \boldsymbol{\lambda}^T\end{bmatrix}\begin{bmatrix}\mathbf{A}_{\mathbf{pp}} & \mathbf{A}_{\mathbf{p}\boldsymbol{\lambda}} \\ (\mathbf{A}_{\mathbf{p}\boldsymbol{\lambda}})^T & \mathbf{A}_{\boldsymbol{\lambda}\boldsymbol{\lambda}}\end{bmatrix}\begin{bmatrix}\mathbf{p} \\ \boldsymbol{\lambda}\end{bmatrix} - \begin{bmatrix}(\mathbf{b}_{\mathbf{p}}^k)^T, (\mathbf{b}_{\boldsymbol{\lambda}}^k)^T\end{bmatrix}\begin{bmatrix}\mathbf{p} \\ \boldsymbol{\lambda}\end{bmatrix}. \tag{24}$$

We refer to Appendix A for details of $\mathbf{A}_{\mathbf{pp}}, \mathbf{A}_{\mathbf{p}\boldsymbol{\lambda}}, \mathbf{A}_{\boldsymbol{\lambda}\boldsymbol{\lambda}}, \mathbf{b}_{\mathbf{p}}^k, \mathbf{b}_{\boldsymbol{\lambda}}^k$. Although $\mathbf{A}_{\boldsymbol{\lambda}\boldsymbol{\lambda}}$ (which is structurally guaranteed to be symmetric positive semi-definite) can be slightly indefinite due to numerical error, we can correct it to be SPD by scaling the diagonal elements by $(1+\xi)$ (we use $\xi = 10^{-3}$) [Smith 2008; Tan et al. 2012]. The resulting system is a box-constrained convex QP (equivalent to an LCP [Takahashi and Batty 2020]) and has a unique solution. Takahashi and Batty [2020] demonstrated that solving this box-constrained convex QP as a whole is more efficient compared to using a block decomposition separating $\mathbf{p}$ and $\boldsymbol{\lambda}$ (i.e., block GS); however, we found that the block GS approach (we used 4 iterations) can actually be more efficient in our framework because it is unnecessary to strictly solve each of the box-constrained QPs (23) due to the surrounding global SP iterations, thereby allowing more efficient solves for the smaller block subcomponents.

One popular approach to solving the box-constrained convex QP (or its subcomponents) is MPRGP [Dostal and Schoberl 2005] (which is based on an active-set method and gradient projections) due to its relative efficiency in large-scale sparse problems [Gerszewski and Bargteil 2013; Narain et al. 2010]. However, we observed that the original MPRGP can be inefficient due to its conservative update of the active sets. This issue had led Kružík et al. [2020] to present an aggressive active-set expansion technique to improve the expansion efficiency, although we encountered convergence failure and stability issues with their approach. Fortunately, having noticed that the active-set update happens more frequently at early stages, we



(a) CG (b) ADMM (c) PIP (1 ls) (d) AugLag (1 ls) (e) MPRGP (ours)

Fig. 4. Inviscid liquid in contact with the top of a spherical container falls due to gravity, with various solver options applied for unilateral incompressibility and wall separation (see §5.1.1 for details). CG does not support wall separation while ADMM fails to preserve the liquid volume due to slow convergence. PIP (1 ls), AugLag (1 ls), and MPRGP (ours) generate comparable results, but MPRGP (ours) is 5.8× and 2.3× faster than PIP (1 ls) and AugLag (1 ls), respectively.

compromise by using their aggressive active-set expansion until the residual falls below 10ϵ (ϵ: termination residual), and using the conservative, original MPRGP near convergence.

Since the convergence rate of MPRGP can be accelerated with a preconditioner, similar to Conjugate Gradient (CG), we employ Modified Incomplete Cholesky (MIC) preconditioning. In addition, as MPRGP can be used without explicitly forming the system matrix, we handle the dense submatrices for $\mathbf{p}$ using the sequential multiplication technique [Bridson 2015; Takahashi and Batty 2020].

One issue that could arise is that the updated $\mathbf{p}$ and $\boldsymbol{\lambda}$ can more severely violate the cone constraints, thereby increasing the objective function value, since we do not consider the cone constraints when solving the box-constrained QP (23) nor use line search when updating $\mathbf{p}$ and $\boldsymbol{\lambda}$ to ensure the decrease of the objective function. Fortunately, each QCQP (22) and box-constrained QP (23) has a nearly unique global minimum, and they are (nearly) dissipative and contractive projections [Kaufman 2009; Narain et al. 2010]. Consequently, the update of $\mathbf{p}$ and $\boldsymbol{\lambda}$ did not introduce any stability issues in practice. This stable behavior is not surprising, considering that the extreme cases for continuum (i.e., inviscid liquids with pressure only and rigid materials with stress only) are still stable, and thus materials with blended pressure and stress are stable as well. In our experiments, we have also tested solving (23) taking the cone constraints into account and using line search when updating $\mathbf{p}$ and $\boldsymbol{\lambda}$. However, we found that the line search frequently failed leading to volume loss and stability issues since the incompressibility and non-penetration constraints could not be satisfied.

Other constrained optimization solvers, such as the augmented Lagrangian method [Nocedal and Wright 2006] and the primal interior point (PIP) method [Forsgren et al. 2002] (also known as a log-barrier method [Nocedal and Wright 2006]), can also be used to solve the box-constrained convex QP for $(\mathbf{p}, \boldsymbol{\lambda})$. However, we found that these approaches are less ideal because the augmented Lagrangian method would converge to a minimum slightly different from the global minimum unless the penalty parameter is infinitely large, and PIP can be unstable depending on parameters. In addition, these approaches are slower to converge and require more iterations to satisfy the box constraints since these constraints are satisfied only at convergence. By contrast, MPRGP can exactly satisfy these constraints at each iteration, allowing for early termination if desired, and thus we prefer our custom MPRGP. Figure 4 compares various box-constrained QP solvers in a wall-separation scenario.

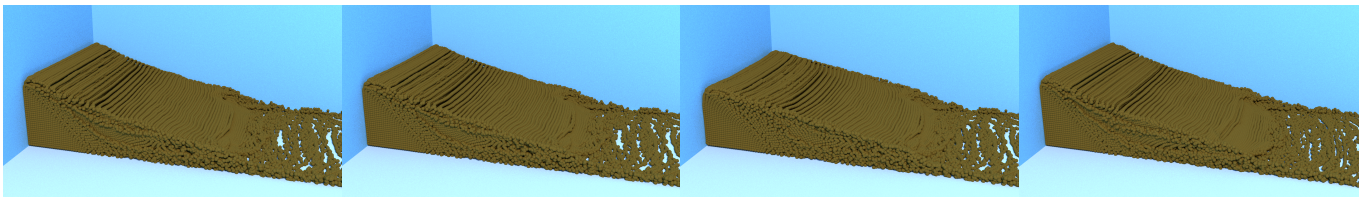
4.3.3 Componentwise Staggered Projections. Because of the staggered projections formulation applied to (13), our method possesses the desired fixed point property. Hence, one could apply various optimizers to solve the QCQP subproblem (22). However, this QCQP subproblem involves all of the stress variables $(\mathbf{s}_{xx}, \mathbf{s}_{xy}, \mathbf{s}_{xz}, \mathbf{s}_{yz}, \mathbf{s}_{zz})$, and thus it is numerically challenging to efficiently solve them all simultaneously due to the slow convergence of iterative solvers. In addition, due to numerical errors (similar to rigid body contact formulations [Smith 2008; Tan et al. 2012]), the resulting system can be indefinite even though the system is structurally guaranteed to be SPD. While this indefinite system can be fixed to be SPD by increasing the diagonal elements (as done for rigid body contacts), this corresponds to introducing a regularization term to stress (analogous to variational viscosity treatments [Larionov et al. 2017]), thus generating viscous fluid-like behaviors, i.e., artificial melting of granular piles (see Figure 5). To address these issues, we further apply staggered projections to (22) to obtain smaller componentwise subproblems on each stress variable, similar to Narain et al. [2010].

There are multiple ways one could split the QCQP subproblem (22). We employ a block Jacobi-like decomposition, which independently processes the componentwise subproblems, achieving perfect symmetry among the frictional stress $\mathbf{s}$. We call this approach Jacobi componentwise SP (JCSP). By first addressing the friction forces using staggered projections for simplicity, JCSP can be written as

$$\mathbf{z}^{k+1} = \underset{0 \le \mathbf{d}(\boldsymbol{\lambda}^k, \mathbf{z})}{\arg\min}\ E(\mathbf{s}^k, \mathbf{p}^k, \boldsymbol{\lambda}^k, \mathbf{z}), \tag{25}$$

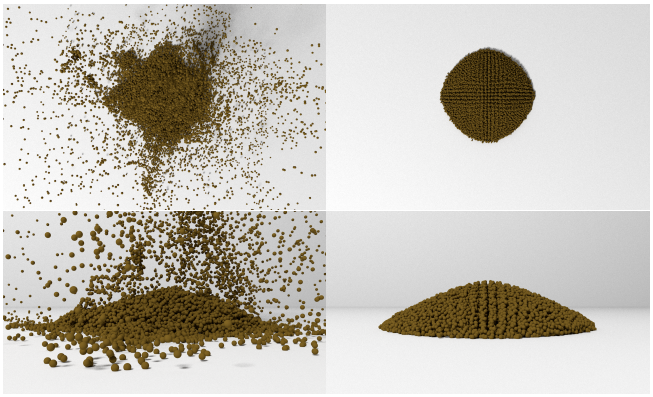
$$\mathbf{s}_e^{k+1} = \underset{0 \le \mathbf{c}(\bar{\mathbf{s}}_e^k, \mathbf{p})}{\arg\min}\ E(\bar{\mathbf{s}}_e^k, \mathbf{p}^k, \boldsymbol{\lambda}^k, \mathbf{z}^{k+1}), \tag{26}$$

where $e \in (xx, xy, xz, yz, zz)$, and $\bar{\mathbf{s}}_{xx}^k = (\mathbf{s}_{xx}, \mathbf{s}_{xy}^k, \mathbf{s}_{xz}^k, \mathbf{s}_{yz}^k, \mathbf{s}_{zz}^k)$. Each of the resulting new QCQP subproblems is much smaller than the original QCQP subproblem (22) and thus can be more efficiently solved. Surprisingly, the convergence rate of JCSP in terms of iteration counts is comparable to an approach that directly solves the QCQP subproblem (22) (with all the stress variables $\mathbf{s}_{xx}, \mathbf{s}_{xy}, \mathbf{s}_{xz}, \mathbf{s}_{yz}, \mathbf{s}_{zz}$ together), giving a significant performance benefit to JCSP. In addition, the decomposed subproblems in JCSP do not suffer from numerical issues in the system entries and thus



(a) SP (b) JCSP (ours)

Fig. 5. An initially box-shaped granular volume being released from rest, simulated using staggered stresses and pyramid constraints with different pressure-friction solvers (see §5.2.1 for details). Due to the regularization term, Staggered Projections (SP) suffers from artificial melting effects so the flow does not come to rest. Our Jacobi Componentwise Staggered Projections (JCSP) approach stably simulates the granular flow and is 3.7× faster than SP.



(a) Without pyramid (b) With pyramid (ours)

Fig. 6. A column of granular material released on the ground, simulated using only 2 SP iterations. Without the auxiliary pyramid constraints used in our method, the simulation explodes; using the auxiliary pyramid enables stable granular flows that do not suffer from artificial anisotropy.

are strictly SPD, making extra regularization unnecessary. Furthermore, since the subproblems in JCSP are projection operations, JCSP maintains the fixed-point property [Narain et al. 2010].

4.3.4 Augmented Lagrangian Method for Cone Constraints. Considering that the unique source of the quadratic constraints is the set of friction cones for contacts and stress in (25) and (26), we aim to handle these cone constraints using the augmented Lagrangian method [Nocedal and Wright 2006], which typically converts solving the constrained QCQP problem into the iterative solution of a series of unconstrained subproblems involving appropriate penalty terms for the cone constraints. However, we found that when treating these subproblems as unconstrained, the solutions can significantly deviate and cause convergence failure, especially with inaccurate initial guesses (see Figure 6). Hence, we introduce auxiliary box (pyramid) constraints on the subproblems that fully *encompass* the cone constraints (by treating other stress variables as 0) to avoid divergent solutions. In practice, since we found that overly tight box constraints can still cause artificial anisotropy (see Figure 3(e)), we employ loose box constraints here. Accordingly, the QCQP subproblems (25) and (26) are modified with a penalty parameter μ (we use $\mu = 10^6$), Lagrange multipliers $\mathbf{g}$ and $\mathbf{h}$ for $\mathbf{c}(\mathbf{s}, \mathbf{p})$ and $\mathbf{d}(\boldsymbol{\lambda}, \mathbf{z})$, respectively, loosening parameter κ (we use $\kappa = 3.0$), and iteration index for the augmented Lagrangian l, into the following box-constrained minimization problem [Narain et al. 2012; Nocedal and Wright 2006]:

$$\mathbf{z}^{k+1} = \underset{-\kappa\boldsymbol{\eta}\boldsymbol{\lambda}^k \le \mathbf{z} \le \kappa\boldsymbol{\eta}\boldsymbol{\lambda}^k}{\arg\min} \left(E_r(\mathbf{s}^k, \mathbf{p}^k, \boldsymbol{\lambda}^k, \mathbf{z}) + E_d(\boldsymbol{\lambda}^k, \mathbf{z}, \mathbf{h}^l) \right), \tag{27}$$

$$E_d(\boldsymbol{\lambda}^k, \mathbf{z}, \mathbf{h}^l) = \frac{1}{2} \left(\mu \left\| \tilde{\mathbf{d}}(\boldsymbol{\lambda}^k, \mathbf{z}, \mathbf{h}^l) \right\|_2^2 - \frac{\left\| \mathbf{h}^l \right\|_2^2}{\mu} \right), \tag{28}$$

$$\mathbf{s}_e^{k+1} = \underset{-\frac{1}{\sqrt{2}}\kappa\alpha\mathbf{p}^k \le \mathbf{s}_e \le \frac{1}{\sqrt{2}}\kappa\alpha\mathbf{p}^k}{\arg\min} \left(E_f(\bar{\mathbf{s}}_e^k, \mathbf{p}^k) + E_r(\bar{\mathbf{s}}_e^k, \mathbf{p}^k, \boldsymbol{\lambda}^k, \mathbf{z}^{k+1}) + E_c(\bar{\mathbf{s}}_e^k, \mathbf{p}^k, \mathbf{g}^l) \right), \tag{29}$$

$$E_c(\bar{\mathbf{s}}_e^k, \mathbf{p}^k, \mathbf{g}^l) = \frac{1}{2} \left(\mu \left\| \tilde{\mathbf{c}}(\bar{\mathbf{s}}_e^k, \mathbf{p}^k, \mathbf{g}^l) \right\|_2^2 - \frac{\left\| \mathbf{g}^l \right\|_2^2}{\mu} \right), \tag{30}$$

where we define $\tilde{\mathbf{c}}(\bar{\mathbf{s}}_e^k, \mathbf{p}^k, \mathbf{g}^l) = \min(\mathbf{c}(\bar{\mathbf{s}}_e^k, \mathbf{p}^k) - \frac{\mathbf{g}^l}{\mu}, 0)$ and $\tilde{\mathbf{d}}(\boldsymbol{\lambda}^k, \mathbf{z}, \mathbf{h}^l) = \min(\mathbf{d}(\boldsymbol{\lambda}^k, \mathbf{z}) - \frac{\mathbf{h}^l}{\mu}, 0)$. The update rule for the Lagrange multipliers becomes

$$\mathbf{g}^{l+1} = -\mu\tilde{\mathbf{c}}(\mathbf{s}^{l+1}, \mathbf{p}^k, \mathbf{g}^l), \text{ and } \mathbf{h}^{l+1} = -\mu\tilde{\mathbf{d}}(\boldsymbol{\lambda}^k, \mathbf{z}^{l+1}, \mathbf{h}^l). \tag{31}$$

While PIP could also be used to handle the cone constraints, we found that the augmented Lagrangian method is more efficient and also convenient since it does not require that the constraints be strictly satisfied at all times. By contrast, PIP requires constraint projections for violated constraints [Forsgren et al. 2002], e.g., after the update of $\mathbf{s}, \mathbf{z}$, and $\mathbf{p}, \boldsymbol{\lambda}$ as well.

4.3.5 Inexact Projected Newton Method. To address the subproblems (27) and (29) of the augmented Lagrangian method, it is necessary to iteratively solve box-constrained minimization problems. While various solvers can be applied, given the quadratic property, we prefer to employ a Newton-type method for its associated quadratic convergence rate. Since the subproblems must be updated for the constraint Hessian and Lagrange multipliers in each Newton iteration, it is not necessary to exactly solve each subproblem; performing more Newton steps with less accurate descent directions is typically more efficient than spending more time finding better descent directions [Yan et al. 2018]. As such, we use the Inexact Projected Newton (IPN) method [Nocedal and Wright 2006], taking the box constraints into account.

Considering the update of an optimization variable $\mathbf{x}$ with the Newton step $\mathbf{x}^{l+1} = \mathbf{x}^l + \gamma\Delta\mathbf{x}^{l+1}$ (where $\Delta\mathbf{x}$ denotes the step direction, l the iteration index, and γ the step size), our first goal is to obtain $\Delta\mathbf{x}^{l+1}$ by solving $\nabla^2 E(\mathbf{x}^l)\Delta\mathbf{x}^{l+1} = -\nabla E(\mathbf{x}^l)$, or equivalently by minimizing the quadratic approximation of the objective $\frac{1}{2}(\Delta\mathbf{x}^{l+1})^T\mathbf{A}^l\Delta\mathbf{x}^{l+1} - (\mathbf{b}^l)^T\Delta\mathbf{x}^{l+1}$ where $\mathbf{A}^l = \nabla^2 E(\mathbf{x}^l)$ and $\mathbf{b}^l = -\nabla E(\mathbf{x}^l)$, respecting the box constraints. These hold for each of $\mathbf{s}_{xx}, \mathbf{s}_{xy}, \mathbf{s}_{xz}, \mathbf{s}_{yz}, \mathbf{s}_{zz}, \mathbf{z}$; see Appendix B for details of $\mathbf{A}^l$ and $\mathbf{b}^l$.

Hessian projection. To ensure $\Delta\mathbf{x}$ is a descent direction to decrease $E(\mathbf{x})$, $\mathbf{A}^l$ needs to be SPD. Since the Hessian of the cone constraint can be indefinite, following Teran et al. [2005], we project each of the constraint Hessians to be SPD. Specifically, since the Hessians of the cone constraints for $\mathbf{s}_{xx}, \mathbf{s}_{xy}, \mathbf{s}_{xz}, \mathbf{s}_{yz}, \mathbf{s}_{zz}$ are one-dimensional due to our componentwise application of SP, we simply clamp negative values to zero. For the solid frictional contacts, the constraint Hessian is 2×2, and we project it to be SPD via eigendecomposition. In addition, since $\mathbf{A}_{\mathbf{zz}}$ can be indefinite due to numerical error, we also adjust it to be SPD as done for $\mathbf{A}_{\boldsymbol{\lambda\lambda}}$.

Box constraints. Given the lower bound $\mathbf{x}_{\min}$ and upper bound $\mathbf{x}_{\max}$ of the box constraint for $\mathbf{x}$ ($\mathbf{x}_{\min} \le \mathbf{x} \le \mathbf{x}_{\max}$), the ideal step size ($\gamma = 1$) gives $\mathbf{x}_{\min} \le \mathbf{x}^l + \Delta\mathbf{x}^{l+1} \le \mathbf{x}_{\max}$. Thus, we can define the box constraints for $\Delta\mathbf{x}^{l+1}$ ($\Delta\mathbf{x}^{l+1}_{\min} \le \Delta\mathbf{x}^{l+1} \le \Delta\mathbf{x}^{l+1}_{\max}$) as $\Delta\mathbf{x}^{l+1}_{\min} = \mathbf{x}_{\min} - \mathbf{x}^l$ and $\Delta\mathbf{x}^{l+1}_{\max} = \mathbf{x}_{\max} - \mathbf{x}^l$.

Box-constrained convex QP solver. Having applied the Hessian projection and box constraint update in the Newton iteration, we have reduced finding the descent direction to solving a box-constrained convex QP, i.e., the same type of problem as for $\mathbf{p}, \boldsymbol{\lambda}$. As such, we again employ our custom MPRGP method (§4.3.2) for $\mathbf{s}, \mathbf{z}$.

Step size. While the ideal step size is $\gamma = 1$ when updating $\mathbf{s}$, we found that, due to the block Jacobi decomposition and cone constraints with the Hessian projection, it is necessary to use a smaller step size. Although backtracking line search has been traditionally used for Newton's method [Nocedal and Wright 2006], in our formulation, line search attempts failed very frequently, leading to less frictional behaviors (see Figure 3(f)), and this procedure was too costly. As such, we employ damped Newton (which uses a fixed step size) with $\gamma = 0.5$. Compared to $\mathbf{s}$, the influence of the Hessian projection is more significant for $\mathbf{z}$, and it was typically necessary to use much smaller step sizes to avoid stability issues. Thus, we use backtracking line search for $\mathbf{z}$.

4.3.6 Choice of Termination Criteria. As a termination criterion, the norm of ∇E has been widely used. However, we found that this norm is usually too strict since reasonable solutions can be obtained even if this norm is not so small [Zhu et al. 2018]. In addition, the penalty parameter for the cone constraints makes it difficult to achieve convergence with this termination criterion, and the box constraints are not considered within ∇E since they are addressed with the active-set method. While the objective function value itself has also been used as a termination criterion [Kaufman et al. 2008], we found that it can stagnate, causing termination with insufficient iterations [Zhu et al. 2018]. Instead, we use the infinity norm of the Newton step direction $\|\Delta\mathbf{x}\|_\infty$ as a termination criterion (similar to [Li et al. 2020a]) since $\Delta\mathbf{x}$ is computed based on ∇E and respects box and cone constraints, and we were able to reliably obtain converged visual results in our experiments.

4.3.7 Warm Start. To accelerate convergence of the staggered projections, we warm start using the pressure and frictional stress from the previous time step (or iterations) as their initial values. Since the number of rigid body contacts is much smaller than the number of degrees of freedoms (DOFs) for fluid pressure and stress, there was no clear benefit for warm starting contact forces. As such, we initialize them to zero. In our experiments, warm starting accelerated the time to convergence by up to two orders of magnitude.

4.3.8 Algorithm of Our Customized Solver. Algorithm 1 summarizes the steps of our tailor-made solver for the QCQP (13).

Algorithm 1 Our customized solver

$k = 0, E_{\text{curr}} = E(\mathbf{s}^k, \mathbf{p}^k, \boldsymbol{\lambda}^k, \mathbf{z}^k)$
do
  $\mathbf{z}^{k+1} = \arg\min_{0 \le \mathbf{d}(\boldsymbol{\lambda}^k, \mathbf{z})} E(\mathbf{s}^k, \mathbf{p}^k, \boldsymbol{\lambda}^k, \mathbf{z})$
  $\mathbf{s}_e^{k+1} = \arg\min_{0 \le \mathbf{c}(\bar{\mathbf{s}}_e^k, \mathbf{p})} E(\bar{\mathbf{s}}_e^k, \mathbf{p}^k, \boldsymbol{\lambda}^k, \mathbf{z}^{k+1})$
  $\mathbf{p}^{k+1}, \boldsymbol{\lambda}^{k+1} = \arg\min_{0 \le \mathbf{p},\ 0 \le \boldsymbol{\lambda}} E(\mathbf{s}^{k+1}, \mathbf{p}, \boldsymbol{\lambda}, \mathbf{z}^{k+1})$.
  $E_{\text{curr}} = E(\mathbf{s}^{k+1}, \mathbf{p}^{k+1}, \boldsymbol{\lambda}^{k+1}, \mathbf{z}^{k+1})$
  $\Delta\mathbf{x}^{k+1} = (\Delta\mathbf{s}^{k+1}, \Delta\mathbf{p}^{k+1}, \Delta\boldsymbol{\lambda}^{k+1}, \Delta\mathbf{z}^{k+1})$
  $k = k + 1$
while $\epsilon < \left\|\Delta\mathbf{x}^k\right\|_\infty$
return $\mathbf{s}^k, \mathbf{p}^k, \boldsymbol{\lambda}^k, \mathbf{z}^k$

4.4 Algorithm

Algorithm 2 summarizes our complete solver.

Algorithm 2 FrictionalMonolith solver

1: Map velocity and density from particles to grid
2: Add external forces to grid and rigid bodies
3: Compute fluid/solid and liquid/air domains
4: Evaluate volume fractions
5: Detect collisions between rigid bodies
6: Assemble the system
7: Solve the system with our customized solver (Algorithm 1)
8: Apply forces to fluids and solids
9: Update solid positions and signed distance using stabilization
10: Advect particles with position corrections

5 RESULTS AND DISCUSSION

Our method is implemented in C++ and parallelized using OpenMP. All examples used adaptive timestepping with CFL numbers between 1.0 and 3.0, with 50 frames per second. For inviscid liquid simulations except for Figure 2, we used cut-cell-based area weighting for $\mathbf{W}_F^u$ and $\mathbf{W}_S^u$ [Ng et al. 2009] and the ghost-fluid method for $\mathbf{W}_L^p$ and $\mathbf{W}_L^u$ [Gibou et al. 2002] (instead of volume weights) for better accuracy, and enforced non-negative pressures to allow wall

separation [Gerszewski and Bargteil 2013] while using constant density. We did not use preconditioning for the independent normal and friction force problems (due to their relatively small DOF counts) but we applied MIC preconditioning to CG and MPRGP for larger problems, except where specifically mentioned. We used a termination residual ϵ between 10^{-3} and 10^{-5}. In addition to the termination criteria mentioned in §4.3.6, we also set the maximum SP iteration count to either 20 or 30 depending on the scenario. For each friction force problem within SP, we use up to 10 iterations to accurately solve it while we use 1 iteration to solve the frictional stress problem for efficiency, except Figure 5 (where we use 10 iterations for accuracy). We executed all our simulations using "e2-standard-8" (8 cores with 32GB RAM) provided by Google Compute Engine. All reported statistics are averaged over the effective simulation length. In the Tables below, p_{outer}, SP_{iter}, s_{inner}, p_{inner}, pn_{inner}, n_{inner}, and t_{inner} denote the total number of outer pressure, SP, inner stress, inner pressure, inner pressure-normal, inner normal, and inner tangent solve iterations per frame, respectively. s_{dof}, p_{dof}, n_{dof}, and t_{dof} denote the number of stress, pressure, normal force, and tangent force DOFs, respectively. The number of substeps per frame is denoted by n. T_p, T_c, T_{pf}, T_{pc}, and T_{pfc}denote the total time in seconds for the pressure, contact, pressure-friction, pressure-contact, and pressure-friction-contact handling phases, respectively. Gray rows indicate early-terminated simulations due to failures.

5.1 Inviscid Liquids

5.1.1 Wall Separation. Since efficiently and robustly enforcing non-negativity constraints for pressure is an essential component in our framework, we first isolate this aspect by evaluating various box-constrained convex QP solvers and techniques in an inviscid liquid wall separation scenario (Figure 4) with 1.2M particles and a grid resolution of $128 \times 128 \times 128$. We compared the following solvers:

(1) CG: reference linear solver without non-negativity constraints;
(2) ADMM;
(3) PIP (1 ls): PIP with up to 1 line search (ls);
(4) PIP (5 ls);
(5) AugLag (1 ls): Augmented Lagrangian method;
(6) AugLag (5 ls);
(7) MPRGP (original): [Dostal and Schoberl 2005];
(8) MPRGP (ours): our custom MPRGP (explained in §4.3.2).

As the inner loop solver for the unconstrained minimization within ADMM, PIP, and AugLag, we employ a CG solver. Since the box-constrained convex QP has a unique solution, it is possible to take larger steps (without using multiple line searches), as long as the constraints are satisfied when necessary. Since PIP requires that constraints always be satisfied, we clamp negative pressures to zero if the constraint is violated. While we have also tested clamping for ADMM and AugLag, it resulted in these solvers stagnating, and thus we do not apply clamping to them. We used up to 5,000 iterations for CG and MPRGP and 100 outer iterations for ADMM, PIP, and AugLag. Performance numbers are given in Table 1, and profiles of total time for the pressure solve are compared in Figure 7.

Due to the non-negativity constraints for pressure, wall separating behaviors are observed for ADMM, PIP, AugLag, and MPRGP, in contrast to CG. However, ADMM failed to satisfy the unilateral incompressibility constraints due to its slow convergence [Boyd et al. 2011]. By contrast, since this problem has a unique solution, PIP, AugLag, and MPRGP generated qualitatively consistent results, albeit with minor differences due to solver tolerances/residuals. For PIP and AugLag, trying multiple line searches slows down convergence due to the smaller step size for the Newton iterations. Because of the more efficient active-set expansion we used, our custom MPRGP is more than one order of magnitude faster than the original MPRGP [Dostal and Schoberl 2005], and is 5.8× and 2.3× faster than PIP (1 ls) and AugLag (1 ls), respectively.

Table 1. Performance numbers for Figure 4. F denotes the convergence failure rate of the solver.

Solver	p_{outer}	p_{inner}	p_{dof}	n	T_p	F (%)
CG		0.3k	293.4k	5.1	3.9	0.0
ADMM	361.5	1.1k	91.7k	3.9	14.5	85.4
PIP (1 ls)	113.4	1.1k	231.0k	3.7	20.8	1.0
PIP (5 ls)	383.7	10.5k	234.0k	3.9	128.2	95.0
AugLag (1 ls)	43.9	0.5k	229.5k	3.7	8.4	0.0
AugLag (5 ls)	190.8	2.4k	227.1k	3.9	38.0	0.1
MPRGP (original)		5.5k	240.6k	3.8	58.1	1.3
MPRGP (ours)		0.3k	232.8k	3.8	3.6	0.0

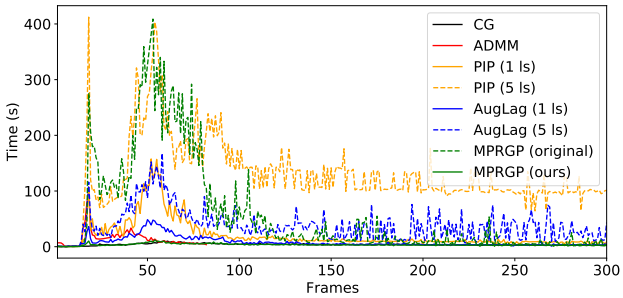


Fig. 7. Profiles of the total time for the entire pressure handling phase per frame for Figure 4. The performance of MPRGP (ours) is comparable to the reference CG, and at least 2× faster than the others.

5.1.2 Sub-Grid Details for Free Surfaces. To demonstrate the effectiveness of our free surface handling with liquid/air volume fractions for sub-grid details, we compare voxel-based (binary weights) free-surface handling (e.g., as used by Narain et al. [2010]) against our approach, using an initially trapezoid-shaped liquid volume, as shown in Figure 2. Here, we applied only minimal surface smoothing to better expose the surface motion. Since the voxel-based approach cannot accurately resolve the location of the free surface, the liquid surface suffers from significant oscillations. By contrast, our volume-based approach considers the free surface location at a sub-grid level and hence generates smoother flows.

5.2 Granular Materials

5.2.1 Unified Pressure-Friction Formulation. To demonstrate the efficiency and importance of our JCSP approach over SP in the unified pressure-friction formulation, we compare these two schemes by

Table 2. Performance numbers for Figure 5.

Solver	SP_{iter}	s_{inner}	p_{inner}	s_{dof}	p_{dof}	n	T_{pf}
SP	30.7	9.8k	1.2k	117.8k	22.3k	1.5	165.95
JCSP (ours)	19.6	54.9k	1.0k	118.2k	22.4k	1.5	45.34

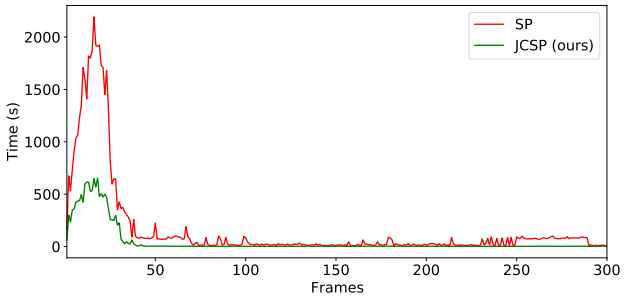


Fig. 8. Profiles of the total time for the entire pressure-friction handling phase per frame for Figure 5. Our JCSP is 3.7× faster than SP.

releasing two initially box-shaped granular volumes under gravity, as shown in Figure 5, with a grid resolution of 96×96×96 and 138.6k particles. For this scene, to avoid the complicating factors of the true friction cone, we use the pyramidal approximation with staggered stress arrangement. Since the friction problem for SP considers all the stress variables at once and we found MIC preconditioning less effective [Takahashi and Batty 2020], we employ Successive-Over-Relaxation (SOR) preconditioning for SP. In addition, the friction problem for SP can be indefinite due to small numerical errors (similar to the rigid body contact problem), so we correct the system to be SPD by slightly increasing diagonal entries [Smith 2008; Tan et al. 2012]. Table 2 summarizes performance numbers and Figure 8 profiles the total time for the pressure-friction handling phase.

SP and JCSP generate comparable results in the initial frames. However, we observe that the increased diagonal entries needed to ensure positive definiteness are equivalent to the regularization term in variational viscosity formulations [Larionov et al. 2017]. The result is that particles simulated with SP exhibit an artificial melting effect as time goes on and fail to come to rest. Moreover, solving the stress system is more costly for SP due to its 5× larger system than that of JCSP, and the convergence of staggered projections for SP was inefficient compared to JCSP. Consequently, JCSP was 3.7× faster than SP. We have also tested higher resolutions to evaluate the scalability and found that the cost of SP becomes superlinearly more expensive (due to less effective SOR preconditioning); SP took more than 20 minutes for 128 × 128 × 128 while JCSP took less than 2 minutes, demonstrating its significant advantage in scalability.

5.2.2 Friction Cone. To evaluate our friction cone formulation, we considered various possible schemes:

(1) Stag. & pyramid: staggered stress with the friction pyramid;
(2) Stag. & cone: staggered stress with the friction cone using $\mathbf{s}_{xy}, \mathbf{s}_{xz}, \mathbf{s}_{yz}$ interpolated to the cell center;
(3) Colo. & pyramid: colocated stress with the friction pyramid;

Table 3. Performance numbers for Figure 3.

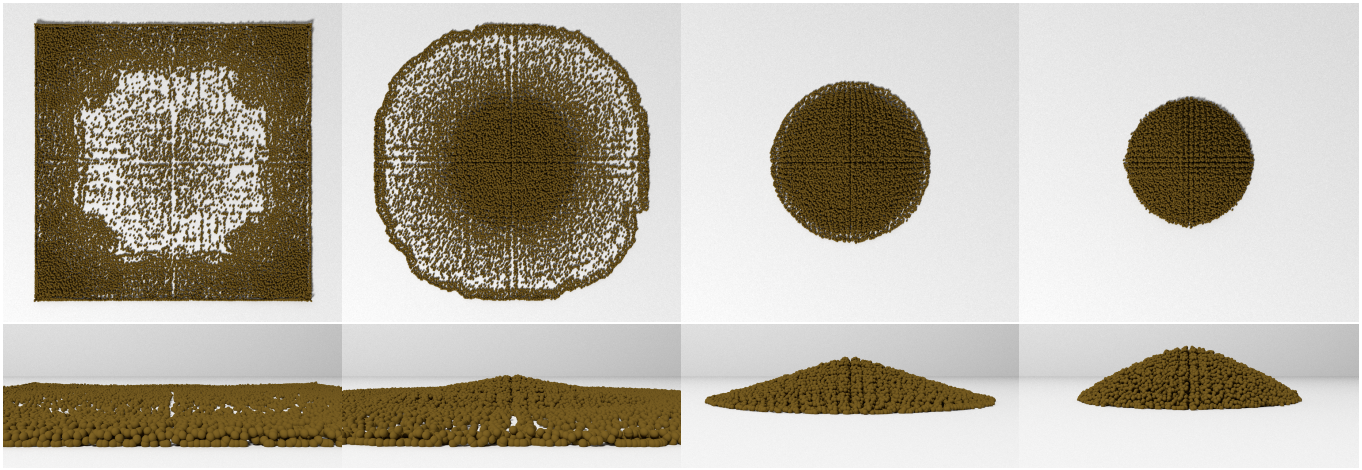
Solver	SP_{iter}	s_{inner}	p_{inner}	s_{dof}	p_{dof}	n	T_{pf}
Stag. & pyramid	6.1	0.4k	59.0	23.6k	5.6k	1.0	0.23
Stag. & cone	34.6	6.7k	328.6	26.3k	6.5k	1.2	2.68
Colo. & pyramid	5.3	2.5k	66.5	22.6k	4.5k	1.0	1.16
Colo. & cone (only)	9.9	0.7k	107.1	20.8k	4.2k	1.0	0.73
Colo. & cone (tight)	6.5	2.0k	74.6	22.5k	4.5k	1.0	1.20
Colo. & cone (ls)	36.1	2.6k	635.9	29.1k	5.8k	1.2	3.02
Colo. & cone (4iter)	6.3	2.7k	62.5	20.8k	4.2k	1.0	1.62
Colo. & cone (ours)	7.3	0.8k	79.9	20.9k	4.2k	1.0	0.69

(4) Colo. & cone (only): colocated stress with the friction cone, without the auxiliary pyramid;
(5) Colo. & cone (tight): colocated stress with the friction cone, with a tight auxiliary pyramid ($\kappa = 1.0$);
(6) Colo. & cone (ls): colocated stress with the cone and auxiliary pyramid, using line search-based update;
(7) Colo. & cone (4iter): colocated stress with the cone and auxiliary pyramid, using 4 iterations for each friction solve;
(8) Colo. & cone (ours): colocated stress with the cone and auxiliary pyramid (1 iteration per friction solve).

In this experiment, we release a column of granular material on the ground, as shown in Figure 3, with a grid resolution of 80×80×80 and 26.0k particles. We used up to 30 SP iterations. Performance numbers are summarized in Table 3, and profiles of total time for the full pressure-friction handling phase are compared in Figure 10.

While Stag. & pyramid efficiently generated stable granular flows, the flows exhibit artificial anisotropy due to the pyramid constraints. Unfortunately simply interpolating $\mathbf{s}_{xy}, \mathbf{s}_{xz}, \mathbf{s}_{yz}$ to the cell center did not resolve the issue, and Stag. & cone failed to avoid artificial anisotropy because the cone constraints could not be sufficiently satisfied with the interpolated values. In addition, since off-diagonal stresses are restricted by multiple, independently defined cone constraints, these stress values are likely to be smaller, generating less-frictional behaviors. Colo. & pyramid generated stable granular piles, albeit with artificial anisotropy due to the pyramid constraints. With sufficiently many SP iterations, Colo. & cone (only) successfully generated stable piles without artificial anisotropy. Colo. & cone (tight) used a tight auxiliary pyramid constraint to improve stability. However, because of the overly tight pyramid constraints, this approach also suffered from artificial anisotropy. Colo. & cone (ls) employs line search to ensure a decrease of the objective function. However, this approach failed to effectively satisfy the cone constraints since line search was almost always unsuccessful, leading to unstable and less-frictional behaviors. While Colo. & cone (4iter) (which performs 4 iterations for each friction solve to ensure a more accurate solution) successfully generated stable granular flows without artificial anisotropy, Colo. & cone (ours) (with just 1 iteration for each friction solve) was able to generate comparable results, validating that extra iterations for the friction solve are unnecessary due to the outer SP iterations; thus it is 2.3× faster than Colo. & cone (4iter) for the same quality.

Next, to demonstrate the necessity of our auxiliary pyramid constraints, we compare the method with and without them in Figure 6 using only 2 SP iterations. In this experiment, granular material



(a) $\alpha = 0.1$ (b) $\alpha = 0.3$ (c) $\alpha = 0.5$ (d) $\alpha = 0.7$

Fig. 9. A column of granular materials released on the ground, simulated with different friction coefficients α.

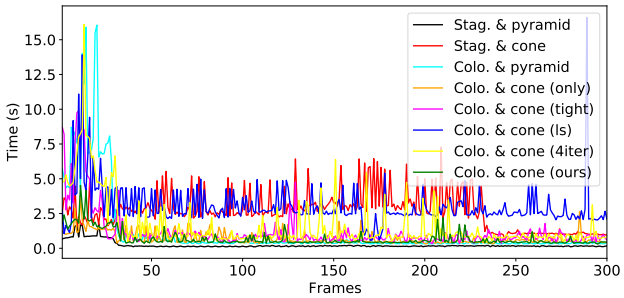


Fig. 10. Profiles of the total time for the entire pressure-friction handling phase per frame for Figure 3. Colo. & cone (ours, in green) eliminates the friction pyramid's artificial anisotropy artifact at a cost of around 3.0× Stag. & pyramid (in black).

simulated without the auxiliary pyramid scattered unstably since the frictional stress can significantly deviate from the desired solution; to prevent this issue without the auxiliary pyramid, up to 30 SP iterations are required to achieve sufficient convergence (Figure 3(d)). By contrast, our use of the auxiliary pyramid enabled stable granular flows, preventing the explosion, without suffering from artificial anisotropy (which is typically introduced by overly tight pyramid constraints), even with just 2 SP iterations.

We also evaluated different friction coefficients α to demonstrate varying frictional behaviors of granular materials, as shown in Figure 9. With smaller α, the material flows and scatters more smoothly, similar to inviscid liquids. As α increases, the material forms stable circular piles of increasing height.

5.2.3 JCSP Convergence. To evaluate the convergence behavior of JCSP for strictly granular scenes, we experimented with different arrangements (staggered with the friction pyramid and colocated with the friction cone) and SP iterations, using a bunny-shaped granular volume (density 1,600 kg/m^3) dropped onto the ground with a grid resolution of $80 \times 80 \times 80$ and 176.5k particles, as shown

Table 4. Performance numbers for Figure 11.

Solver	SP_{iter}	s_{inner}	p_{inner}	s_{dof}	p_{dof}	n	T_{pf}
Stag. (2iter)	1.9	181.2	26.1	132.1k	28.8k	1.1	0.95
Stag. (20iter)	5.8	746.5	101.7	128.3k	27.5k	1.1	1.63
Colo. (2iter)	2.8	430.0	59.3	145.4k	29.1k	1.4	4.75
Colo. (20iter)	8.1	1,243.1	169.5	141.9k	28.4k	1.0	6.60

in Figure 11. Performance numbers are summarized in Table 4, and timing profiles are compared in Figure 12.

While Stag. (2iter) generated stable granular flows overall, some particles are scattered due to the early-terminated pressure-friction computation. Stag. (20iter) successfully generated a stable pile without particle scattering due to the additional SP iterations. Similarly, Colo. (2iter) suffered from particle scattering due to an insufficiently accurate pressure-friction force balance while Colo. (20iter) generated a stable and tall pile. In this example, performing 20 iterations increased cost only slightly (around 1.5×).

We additionally evaluated the critical importance of performing enough SP iterations for accurate pressure-friction solutions in two-way coupling scenarios involving granular material (density 1,600 kg/m^3) and a (single) dynamic rigid body (density 2,000 kg/m^3). We considered both the staggered stress (Figure 13) and colocated stress (Figure 14) arrangements with a grid resolution of $64 \times 64 \times 64$ with 173.7k particles. Performance numbers are listed in Table 5.

With Stag. (2iter) the rigid bunny dropped onto the pile jitters, while Stag. (20iter) enables the lighter pile to stably support the denser bunny. Granular material simulated with Colo. (2iter) exhibits artificial melting, and the rigid block gradually (and erroneously) sinks into the pile. By contrast, granular material with Colo. (20iter) stably supports the rigid block. In each case, the increased cost for the additional iterations is relatively modest (~1.4× on average).

5.3 Contact-Aware Coupling with Inviscid Liquids

To demonstrate the benefits of our monolithic coupling framework even in the absence of frictional (fluid) stress, we experiment with

(a) Stag. (2iter) (b) Stag. (20iter) (c) Colo. (2iter) (d) Colo. (20iter)

Fig. 11. An initially bunny-shaped volume of granular material dropped onto the ground, simulated with different stress arrangements and maximum SP iterations. Stag. (2iter) generates stable granular flows with some scattered particles due to non-converged frictional stress while Stag. (20iter) produce more stable and consistent piles of granular materials. Colo. (2iter) suffers from particle scattering due to the early-terminated stress computations while Colo. (20iter) generates a stable and tall pile of granular materials.

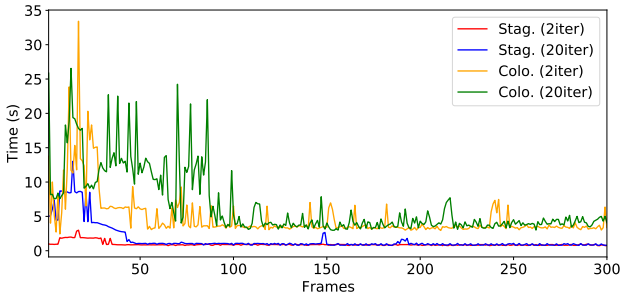
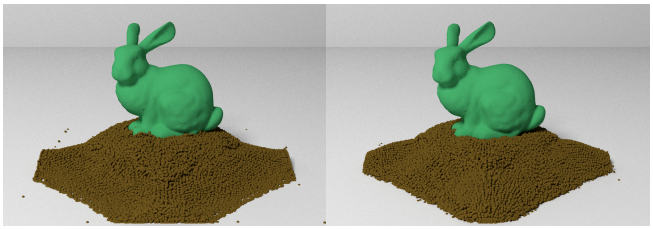


Fig. 12. Profiles of the total time for the entire pressure-friction handling phase per frame for Figure 11.



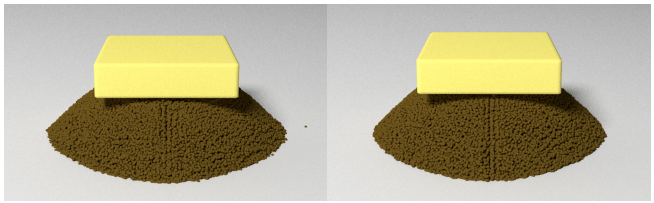
(a) Stag. (2iter) (b) Stag. (20iter)

Fig. 13. A hemisphere of granular material on the ground is released from rest and a rigid bunny is dropped onto the resulting granular pile, simulated with the staggered stress arrangement and different maximum SP iterations. The rigid bunny on the pile with Stag. (2iter) jitters due to inaccurate pressure-friction coupling while Stag. (20iter) generates stable behavior for both the grains and the bunny.

Table 5. Performance numbers for Figures 13 and 14.

Solver	SP_{iter}	s_{inner}	p_{inner}	s_{dof}	p_{dof}	n	T_{pf}
Stag. (2iter)	4.6	888.9	100.3	124.5k	26.5k	2.3	2.57
Stag. (20iter)	11.9	1868.5	240.5	122.6k	25.8k	1.2	3.08
Colo. (2iter)	3.3	285.6	73.1	128.2k	25.6k	1.7	4.76
Colo. (20iter)	16.5	1,416.5	338.2	126.4k	25.3k	1.1	7.95

interactions of inviscid liquids and rigid bodies involving frictional contacts. Concretely, in the case of inviscid liquids the differences between our method (involving a QCQP) and the method of Takahashi and Batty [2020] (involving only an LCP) are that our rigid body contacts consider the true friction cone and our liquid supports



(a) Colo. (2iter) (b) Colo. (20iter)

Fig. 14. A hemisphere of granular material on the ground is released from rest and a two-way coupled rigid block is dropped onto the resulting pile, simulated with the colocated stress arrangement and different maximum SP iterations. With Colo. (2iter) the flow exhibits artificial melting effects due to the less accurate pressure-friction coupling; with Colo. (20iter) the grains correctly (and stably) support the block.

wall-separation via unilateral incompressibility. We evaluate four pressure-contact coupling schemes:

(1) P-C: pressure solve followed by contact solve (once);
(2) C-P: contact solve followed by pressure solve (once);
(3) T-PN unified: staggered projections of friction (tangential) force and unified pressure-normal force solve;
(4) T-P-N iter: staggered projections of tangential force and iterative block GS pressure and normal force solves (ours).

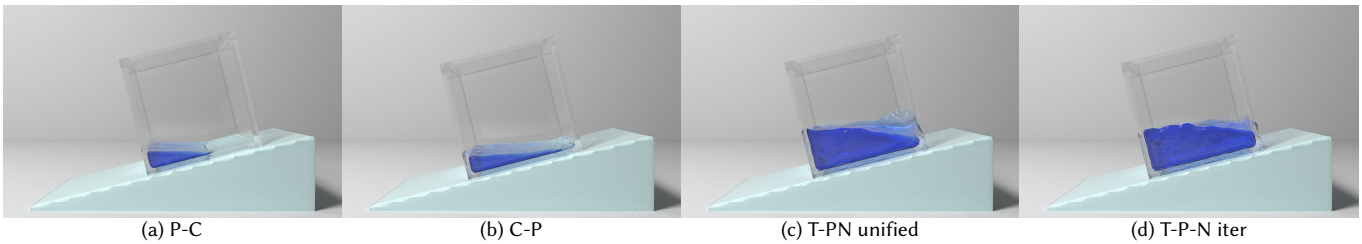
In Figure 15, we dropped a hollow glass cube (density 500 kg/m^3) containing a bunny-shaped volume of inviscid liquid (density 1,000 kg/m^3) onto a slope, using a grid resolution of $192 \times 144 \times 96$ and 177.2k particles. We summarize performance numbers in Table 6 and compare profiles of total time for the entire pressure and frictional contact handling phase in Figure 16.

For P-C, while contacts between the slope and glass cube are properly addressed by the concluding contact solve, the liquid suffers from significant volume loss due to the neglected unilateral incompressibility constraints. For C-P, due to the neglected contact handling, the cube slips more on the slope, and inconsistent solid velocities cause some liquid volume loss. By contrast, both T-PN unified and T-P-N iter correctly enforce the unilateral incompressibility constraints while appropriately handling the frictional contacts, with similar computational costs.

5.4 Contact-Aware Coupling with Granular Materials

We next demonstrate our full monolithic formulation for simultaneous coupling of granular materials and multiple rigid bodies with frictional contacts. We evaluate four possible schemes:

(1) PF-C: pressure-friction solve followed by contact solve (once);



(a) P-C (b) C-P (c) T-PN unified (d) T-P-N iter

Fig. 15. A hollow glass cube containing initially bunny-shaped inviscid liquid dropped onto a slope, with various solver options (see §5.3 for details). P-C fails to enforce the fluid incompressibility losing the volume. C-P also fails to preserve the volume due to the inconsistent solid velocity and causes more slips of the cube due to neglected contact handling. Both T-PN unified and T-P-N iter correctly preserve fluid volume and handle frictional contacts.

Table 6. Performance numbers for Figure 15.

Solver	SP_{iter}	p_{inner}	pn_{inner}	n_{inner}	t_{inner}	p_{dof}	n_{dof}	t_{dof}	n	T_p	T_c	T_{pc}
P-C	26.2	82.1		112.8	10.3k	18.6k	4.0	7.9	5.3	0.28	0.38	0.66
C-P	26.7	86.0		111.1	1.6k	26.8k	4.0	7.9	5.1	0.33	0.08	0.42
T-PN unified	58.7		3,267.9		1.6k	47.8k	4.0	7.9	6.2			50.4
T-P-N iter	73.0	7,208.5		2,091.9	28.4k	48.8k	4.0	7.9	8.3			49.5

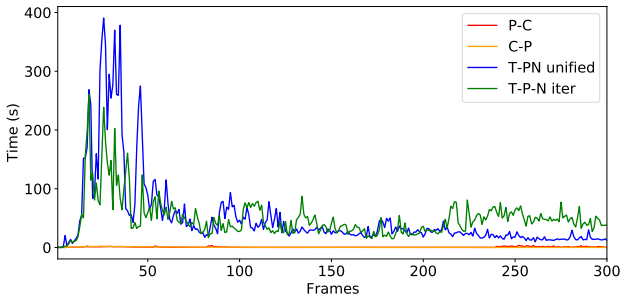


Fig. 16. Profiles of the total time for the entire pressure-contact handling phase per frame for Figure 15.

(2) C-PF: contact solve followed by pressure-friction solve (once);
(3) T-F-PN unified: SP of tangential force, friction, and unified pressure-normal force solves;
(4) T-F-P-N iter: SP of tangential force, friction, and iterative block GS pressure and normal force solves (ours).

In Figure 17, we released a hemisphere of granular material (density 1,600 kg/m^3) on the ground, dropped a two-way coupled block (density 500 kg/m^3) onto the pile, and a rigid bunny (density 500 kg/m^3) on top of the block. We used a grid resolution of $64 \times 64 \times 64$ and 90.2k particles. We summarize performance numbers in Table 7 and compare profiles of total time for the entire pressure-friction-contact handling phase in Figure 18.

Due to the neglected unilateral incompressibility constraints with PF-C, the rigid bunny incorrectly pushes the board into the granular materials. C-PF neglects the contacts between the bunny and the board, causing the simulation to explode. T-F-PN unified and T-F-P-N iter generate plausible and comparable granular flows satisfying both the unilateral incompressibility and friction cone constraints simultaneously with rigid body frictional contacts.

5.5 Complex Examples

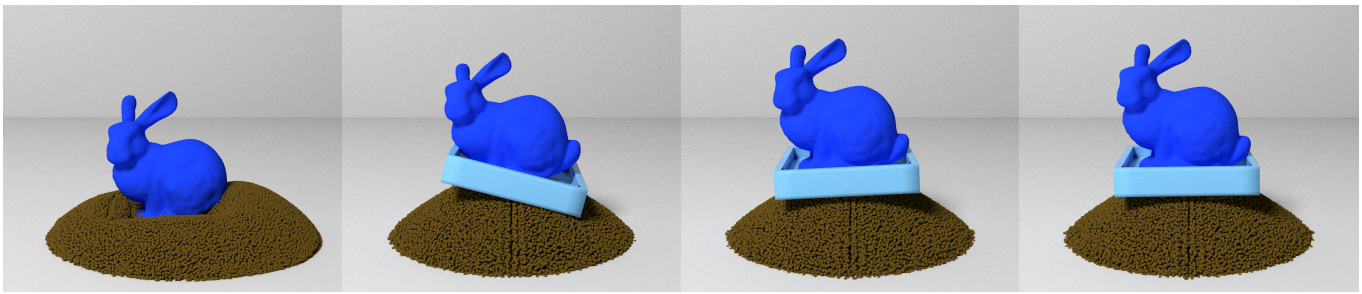
The following more complex examples further demonstrate the capabilities of our method for friction-critical scenarios. Figure 19 demonstrates a hollow glass cube containing a bunny-shaped inviscid liquid volume pinched by a static block (left) and a two-way coupled block (right) with a constant leftward body force applied on the right block. We compare simulations with $\eta = 0.05$ and $\eta = 0.20$ at a grid resolution of $128 \times 128 \times 64$ with 144.0k particles; the total simulation times per frame are 35.40 s and 48.22 s, respectively. With $\eta = 0.05$ the cube is initially slowed when pinched, but ultimately continues sliding downward due to the weight of the splashing liquid inside. By contrast, with $\eta = 0.20$ the stronger frictional contact forces successfully halt the cube and keep it suspended in the air.

Figure 1 (left) simulates granular material flowing through a collection of bunnies that are stuck in the middle of an hourglass due to frictional contacts. This simulation used a grid resolution of $80 \times 160 \times 80$ with 463.4k particles. In this example, we employed the staggered stress arrangement with friction pyramid constraints to demonstrate that our monolithic formulation is effective in this setting. The total simulation time per frame is 368.95 s.

Lastly, in Figure 1 (right), we simulate a rotating drum mixing multiple bunnies and granular material together. This simulation used a grid resolution of $64 \times 64 \times 64$ with 70.2k particles, employing the colocated stress arrangement with friction cone constraints. The total simulation time per frame is 161.92s.

5.6 Discussion

In our experiments, we found that performing more iterations with appropriate granularity of the subproblems is typically more efficient (e.g., block GS for the pressure-normal force problem vs. its unified variant) since it is unnecessary to strictly solve each subproblem due to the outer SP iterations. However, if early termination of SP is desired, one would prefer accurately solving each subproblem



(a) PF-C (b) C-PF (c) T-F-PN unified (d) T-F-P-N iter

Fig. 17. A hemisphere of granular material on the ground is released from rest and a two-way coupled board and a rigid bunny are dropped onto the resting pile, simulated with various solver options (see §5.4 for details). PF-C fails to preserve the volume of granular materials while neglected contacts with C-PF cause a explosion at frame 69. By contrast, both T-F-PN unified and T-F-P-N iter generate plausible behaviors of the granular materials, board, and bunny.

Table 7. Performance numbers for Figure 17. SP_{iter} denotes the number of SP iterations for granular materials/contacts for PF-C and C-PF.

Solver	SP_{iter}	s_{inner}	p_{inner}	pn_{inner}	n_{inner}	t_{inner}	s_{dof}	p_{dof}	n_{dof}	t_{dof}	n	T_{pf}	T_c	T_{pfc}
PF-C	78.2/9.8	3.2k	2.6k		74.5	768.9	60.6k	12.0k	4.3	8.6	2.6	10.8	0.03	10.9
C-PF	37.0/2.4	3.0k	0.5k		5.8	26.6	67.0k	13.4k	0.7	1.4	1.3	6.96	0.0	6.96
T-F-PN unified	39.9	3.3k		0.7k		1,542.2	68.8k	13.8k	2.5	5.1	1.3			28.6
T-F-P-N iter	33.9	3.0k	1.6k		297.9	1,325.8	68.6k	13.7k	2.5	5.1	1.1			25.2

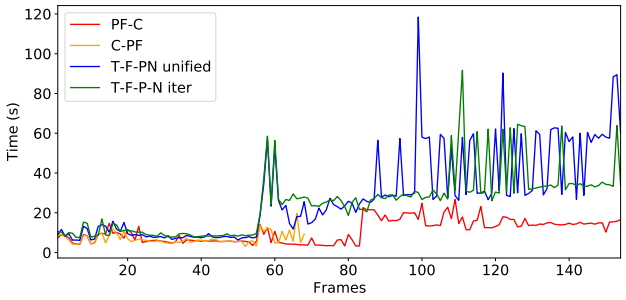


Fig. 18. Profiles of the total time for the entire pressure-friction-contact handling phase per frame for Figure 17.

for numerical stability; thus investigating optimal subproblem granularity (and computational ordering) along with solver parameters would be beneficial.

While our formulation for friction cone constraints with the colocated stress arrangement removes the artificial anisotropy effect, as shown in Figure 3, employing cone constraints increases the required number of SP iterations for sufficient convergence if one is to avoid stability issues (as compared to pyramid constraints, addressed with MPRGP without the augmented Lagrangian). The colocated stress arrangement enlarges the stencil size for $\mathbf{s}_{xy}, \mathbf{s}_{xz}, \mathbf{s}_{yz}$ leading to a denser system with more nonzeros. In addition, the transition matrix that is necessary to combine standard finite differencing with our hybrid grid introduces numerical averaging in the frictional stress, resulting in slightly smoothed velocity fields.

Since MPRGP supports preconditioning, and given recent multigrid treatments for fluid pressure LCPs [Chentanez and Mueller-Fischer 2012; Lai et al. 2020] and viscosity linear systems [Aanjaneya et al. 2019], it could be valuable to develop a multigrid preconditioner specialized to further accelerate our granular flow formulation.

To visualize the granular flow, we directly rendered the underlying APIC particles. To further improve the visual quality, it would be worthwhile exploring secondary simulations [Ihmsen et al. 2013] or split/merge operations to resolve finer details [Daviet and Bertails-Descoubes 2016; Narain et al. 2010].

6 CONCLUSIONS

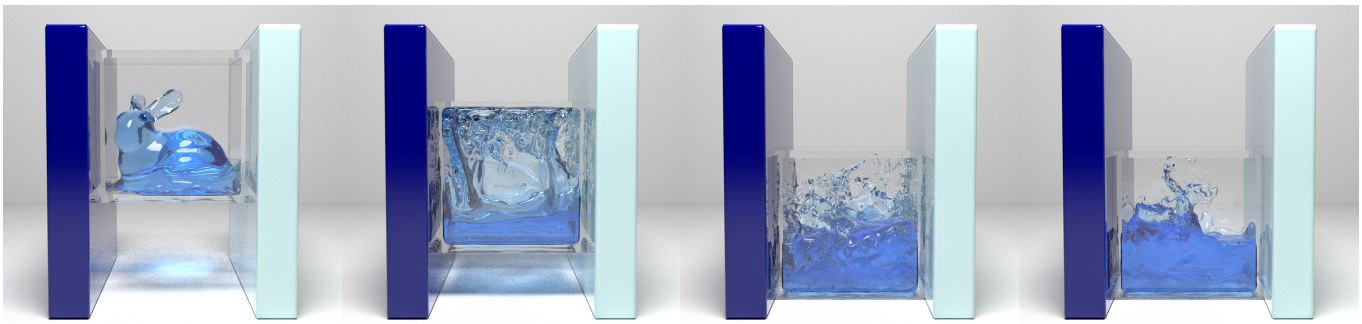
We have proposed FrictionalMonolith, a monolithic framework for challenging pressure-friction-contact problems that can treat, in a unified way, unilateral fluid incompressibility constraints, implicit granular friction integration under friction cone constraints, frictional contact resolution among rigid bodies, and the interactions among liquid/granular materials and rigid bodies. To efficiently and robustly solve the QCQP resulting from this formulation, we presented our customized solver and evaluated its performance. Finally, we demonstrated the efficacy of our monolithic solver on a range of difficult granular/rigid interaction scenarios in which it eliminates artifacts and surpasses the capabilities of prior or alternative schemes.

ACKNOWLEDGMENTS

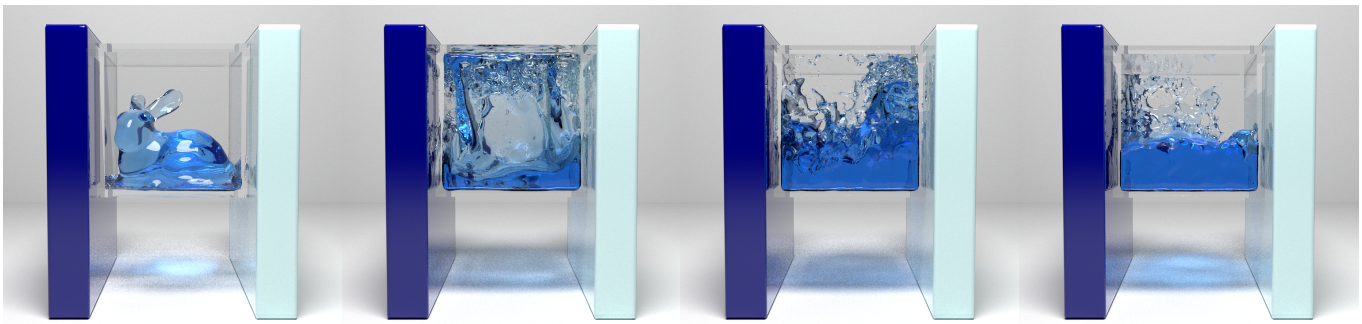
We would like to thank the anonymous reviewers for their valuable suggestions and comments. This work was supported in part by the Natural Sciences and Engineering Research Council of Canada (Grant RGPIN-2021-02524).

REFERENCES

Mridul Aanjaneya, Chengguizi Han, Ryan Goldade, and Christopher Batty. 2019. An Efficient Geometric Multigrid Solver for Viscous Liquids. *Proc. ACM Comput. Graph. Interact. Tech.* 2, 2, Article 14 (July 2019), 21 pages.



(a) $\eta = 0.05$



(b) $\eta = 0.20$

Fig. 19. A hollow glass cube containing a bunny-shaped volume of inviscid liquid is dropped and pinched between a static navy block and a two-way coupled ice blue rigid body block, with a constant leftward body force applied on the right block. With $\eta = 0.05$ (top) the cube continues sliding down while with $\eta = 0.20$ (bottom) frictional contact forces bringing it quickly to rest, still suspended in the air.

Muzaffer Akbay, Nicholas Nobles, Victor Zordan, and Tamar Shinar. 2018. An extended partitioned method for conservative solid-fluid coupling. *ACM Transactions on Graphics* 37 (2018), 1–12.

Nadir Akinci, Markus Ihmsen, Gizem Akinci, Barbara Solenthaler, and Matthias Teschner. 2012. Versatile Rigid-fluid Coupling for Incompressible SPH. *ACM Transactions on Graphics* 31, 4, Article 62 (2012), 62:1–62:8 pages.

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*. 25–32.

Iván Alduan, Ángel Tena, and Miguel A. Otaduy. 2009. Simulation of High-Resolution Granular Media. In *CEIG 09 - Congreso Espanol de Informatica Grafica*.

Mihai Anitescu and Gary D. Hart. 2004. 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 Alessandro Tasora. 2010. An iterative approach for cone complementarity problems for nonsmooth dynamics. *Computational Optimization and Applications* 47 (10 2010), 207–235.

Stefan Band, Christoph Gissler, Markus Ihmsen, Jens Cornelis, Andreas Peer, and Matthias Teschner. 2018a. Pressure Boundaries for Implicit Incompressible SPH. *ACM Trans. Graph.* 37, 2, Article 14 (2018).

Stefan Band, Christoph Gissler, Andreas Peer, and Matthias Teschner. 2018b. MLS pressure boundaries for divergence-free and viscous SPH fluids. *Computers & Graphics* 76 (2018), 37 – 46.

Christopher Batty, Florence Bertails, and Robert Bridson. 2007. A Fast Variational Framework for Accurate Solid-fluid Coupling. *ACM Trans. Graph.* 26, 3, Article 100 (2007).

Markus Becker, Hendrik Tessendorf, and Matthias Teschner. 2009. Direct Forcing for Lagrangian Rigid-Fluid Coupling. *IEEE Transactions on Visualization and Computer Graphics* 15, 3 (2009), 493–503.

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*. 77–86.

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. 2011. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Found. Trends Mach. Learn.* 3, 1 (Jan. 2011), 1–122.

Christopher Brandt, Leonardo Scandolo, Elmar Eisemann, and Klaus Hildebrandt. 2019. The reduced immersed method for real-time fluid-elastic solid interaction and contact simulation. *ACM Transactions on Graphics* 38 (2019).

Robert Bridson. 2015. *Fluid Simulation for Computer Graphics*. A K Peters/CRC Press.

Mark Carlson, Peter Mucha, and Greg Turk. 2004. Rigid Fluid: Animating the Interplay Between Rigid Bodies and Fluid. *ACM Transactions on Graphics* 23 (2004).

Nuttapong Chentanez and Matthias Mueller-Fischer. 2012. A Multigrid Fluid Pressure Solver Handling Separating Solid Boundary Conditions. *IEEE Transactions on Visualization and Computer Graphics* 18, 8 (2012), 1191–1201.

M. B. Cline and D. K. Pai. 2003. Post-stabilization for rigid body simulation with contact and constraints. In *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, Vol. 3. 3744–3751 vol.3.

Gilles Daviet. 2020. Simple and Scalable Frictional Contacts for Thin Nodal Objects. *ACM Trans. Graph.* 39, 4, Article 61 (July 2020), 16 pages.

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, Article 102 (2016), 102:1–102:13 pages.

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 (Dec. 2011), 1–12.

Saibal De, Eduardo Corona, Paramsothy Jayakumar, and Shravan Veerapaneni. 2019. Scalable Solvers for Cone Complementarity Problems in Frictional Multibody Dynamics. In *2019 IEEE High Performance Extreme Computing Conference (HPEC)*. 1–7.

Ounan Ding and Craig Schroeder. 2020. Penalty Force for Coupling Materials with Coulomb Friction. *IEEE Transactions on Visualization and Computer Graphics* 26, 7 (2020), 2443–2455.

Zdenek Dostal and Joachim Schoberl. 2005. Minimizing Quadratic Functions Subject to Bound Constraints with the Rate of Convergence and Finite Termination. *Computational Optimization and Applications* 30, 1 (2005), 23–43.

Kenny Erleben. 2007. Velocity-Based Shock Propagation for Multibody Dynamics Animation. *ACM Trans. Graph.* 26, 2 (June 2007), 12–es.

Kenny Erleben. 2017. Rigid Body Contact Problems Using Proximal Operators. In *Proceedings of the ACM SIGGRAPH / Eurographics Symposium on Computer Animation (SCA '17)*. Association for Computing Machinery, Article 13, 12 pages.

Yu Fang, Ziyin Qu, Minchen Li, Xinxin Zhang, Yixin Zhu, Mridul Aanjaneya, and Chenfanfu Jiang. 2020. IQ-MPM: An Interface Quadrature Material Point Method for Non-Sticky Strongly Two-Way Coupled Nonlinear Solids and Fluids. *ACM Trans. Graph.* 39, 4, Article 51 (2020).

Anders Forsgren, Philip E. Gill, and Margaret H. Wright. 2002. Interior Methods for Nonlinear Optimization. *SIAM Rev.* 44, 4 (2002), 525–597.

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 Trans. Graph.* 37, 4, Article 149 (July 2018).

Dan Gerszewski and Adam W. Bargteil. 2013. Physics-based Animation of Large-scale Splashing Liquids. *ACM Transactions on Graphics* 32, 6, Article 185 (2013), 6 pages.

Frederic Gibou, Ronald P. Fedkiw, Li-Tien Cheng, and Myungjoo Kang. 2002. A Second-Order-Accurate Symmetric Discretization of the Poisson Equation on Irregular Domains. *J. Comput. Phys.* 176, 1 (2002), 205 – 227.

Christoph Gissler, Andreas Peer, Stefan Band, Jan Bender, and Matthias Teschner. 2019. Interlinked SPH Pressure Solvers for Strong Fluid-Rigid Coupling. *ACM Transactions on Graphics* 38 (2019), 1–13.

Tolga G. Goktekin, Adam W. Bargteil, and James F. O'Brien. 2004. A Method for Animating Viscoelastic Fluids. *ACM Trans. Graph.* 23, 3 (2004), 463–468.

Eran Guendelman, Andrew Selle, Frank Losasso, and Ronald Fedkiw. 2005. Coupling water and smoke to thin deformable and rigid shells. *ACM Trans. Graph.* 24 (2005), 973–981.

Xuchen Han, Theodore F. Gast, Qi Guo, Stephanie Wang, Chenfanfu Jiang, and Joseph Teran. 2019. A Hybrid Material Point Method for Frictional Contact with Diverse Materials. *Proc. ACM Comput. Graph. Interact. Tech.* 2, 2, Article 17 (2019).

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 Trans. Graph.* 37, 4, Article 150 (2018).

David A.B. Hyde and Ronald Fedkiw. 2019. A unified approach to monolithic solid-fluid coupling of sub-grid and more resolved solids. *J. Comput. Phys.* 390 (2019), 490 – 526.

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.

Chenfanfu Jiang, Theodore Gast, and Joseph Teran. 2017. Anisotropic elastoplasticity for cloth, knit and hair frictional contact. *ACM Transactions on Graphics* 36 (07 2017), 1–14.

Chenfanfu Jiang, Craig Schroeder, Andrew Selle, Joseph Teran, and Alexey Stomakhin. 2015. The Affine Particle-in-cell Method. *ACM Trans. Graph.* 34, 4, Article 51 (2015), 51:1–51:10 pages.

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*. 1–52.

Yupeng Jiang, Minchen Li, Chenfanfu Jiang, and Fernando Alonso-Marroquin. 2020. A hybrid material-point spheropolygon-element method for solid and granular material interaction. *Internat. J. Numer. Methods Engrg.* (2020).

Danny M. Kaufman. 2009. Coupled Principles for Computational Frictional Contact Mechanics.

M. Danny Kaufman, Shinjiro Sueda, L. Doug James, and K. Dinesh Pai. 2008. Staggered projections for frictional contact in multibody systems. *ACM Trans. Graph.* (2008), 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 (2016), 103:1–103:12 pages.

Bryan Klingner, Bryan Feldman, Nuttapong Chentanez, and James O'Brien. 2006. Fluid animation with dynamic meshes. *ACM Trans. Graph.* 25 (2006), 820–825.

Dan Koschier and Jan Bender. 2017. Density Maps for Improved SPH Boundary Handling. In *Proceedings of the ACM SIGGRAPH / Eurographics Symposium on Computer Animation*. Article 1, 10 pages.

J. Kružík, D. Horák, M. Čermák, L. Pospíšil, and M. Pecha. 2020. Active set expansion strategies in MPRGP algorithm. *Advances in Engineering Software* 149 (2020), 102895.

Tassilo Kugelstadt, Andreas Longva, Nils Thurey, and Jan Bender. 2019. Implicit Density Projection for Volume Conserving Liquids. *IEEE Transactions on Visualization and Computer Graphics* (2019), 1–1.

Junyu Lai, Yangang Chen, Yu Gu, Christopher Batty, and Justin W.L. Wan. 2020. Fast and Scalable Solvers for the Fluid Pressure Equations with Separating Solid Boundary Conditions. *Computer Graphics Forum* 39, 2 (2020), 23–33.

Egor Larionov, Christopher Batty, and Robert Bridson. 2017. Variational Stokes: A Unified Pressure-viscosity Solver for Accurate Viscous Liquids. *ACM Trans. Graph.* 36, 4, Article 101 (July 2017), 101:1–101:11 pages.

Egor Larionov, Ye Fan, and Dinesh K. Pai. 2021. Frictional Contact on Smooth Elastic Solids. *ACM Trans. Graph.* 40, 2, Article 15 (April 2021), 17 pages.

Quentin Le Lidec, Igor Kalevatykh, Ivan Laptev, Cordelia Schmid, and Justin Carpentier. 2021. Differentiable simulation for physical system identification. *IEEE Robotics and Automation Letters* (2021). https://hal.archives-ouvertes.fr/hal-03025616

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

Jie Li, Gilles Daviet, Rahul Narain, Florence Bertails-Descoubes, Matthew Overby, George E. Brown, and Laurence Boissieux. 2018. An Implicit Frictional Contact Solver for Adaptive Cloth Simulation. *ACM Trans. Graph.* 37, 4, Article 52 (July 2018), 15 pages.

Minchen Li, Zachary Ferguson, Teseo Schneider, Timothy Langlois, Denis Zorin, Daniele Panozzo, Chenfanfu Jiang, and Danny M. Kaufman. 2020a. Incremental Potential Contact: Intersection-and Inversion-Free, Large-Deformation Dynamics. *ACM Trans. Graph.* 39, 4, Article 49 (July 2020), 20 pages.

Minchen Li, Danny M. Kaufman, and Chenfanfu Jiang. 2020b. Codimensional Incremental Potential Contact. arXiv:cs.GR/2012.04457

Mickaël Ly, Jean Jouve, Laurence Boissieux, and Florence Bertails-Descoubes. 2020. Projective Dynamics with Dry Frictional Contact. *ACM Trans. Graph.* 39, 4, Article 57 (July 2020), 8 pages.

Miles Macklin, Kenny Erleben, Matthias Müller, Nuttapong Chentanez, Stefan Jeschke, and Viktor Makoviychuk. 2019. Non-Smooth Newton Methods for Deformable Multi-Body Dynamics. *ACM Trans. Graph.* 38, 5, Article 140 (Oct. 2019), 20 pages.

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

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.

Daniel Melanz, Luning Fang, Paramsothy Jayakumar, and Dan Negrut. 2017. A comparison of numerical methods for solving multibody dynamics problems with frictional contact modeled via differential variational inequalities. *Computer Methods in Applied Mechanics and Engineering* 320 (2017), 668 – 693.

Matthias Müller, Nuttapong Chentanez, Miles Macklin, and Stefan Jeschke. 2017. Long Range Constraints for Rigid Body Simulations. In *Proceedings of the ACM SIGGRAPH / Eurographics Symposium on Computer Animation (SCA '17)*. Association for Computing Machinery, Article 14, 10 pages.

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

Rahul Narain, Armin Samii, and James F. O'Brien. 2012. Adaptive Anisotropic Remeshing for Cloth Simulation. *ACM Trans. Graph.* 31, 6, Article 152 (2012), 10 pages.

Yen Ting Ng, Chohong Min, and Frédéric Gibou. 2009. An efficient fluid–solid coupling algorithm for single-phase flows. *J. Comput. Phys.* 228, 23 (2009), 8807 – 8829.

Jorge Nocedal and Stephen J. Wright. 2006. *Numerical Optimization* (second ed.). Springer, New York, NY, USA.

Koichi Onoue and Tomoyuki Nishita. 2003. Virtual Sandbox. In *Proceedings of the 11th Pacific Conference on Computer Graphics and Applications (PG '03)*. 252.

Charles Peskin . 2002. Peskin, C.S.: The immersed boundary method. Acta Numerica 11, 479-517. *Acta Numerica* 11 (01 2002), 479 – 517.

Mathieu Renouf and Pierre Alart. 2005. Conjugate gradient type algorithms for frictional multi-contact problems: applications to granular materials. *Computer Methods in Applied Mechanics and Engineering* 194, 18 (2005), 2019 – 2041.

Avi Robinson-Mosher, R. Elliot English, and Ronald Fedkiw. 2009. Accurate Tangential Velocities for Solid Fluid Coupling. In *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '09)*. 227–236.

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

Avi Robinson-Mosher, Tamar Shinar, Jon Gretarsson, Jonathan Su, and Ronald Fedkiw. 2008. Two-way Coupling of Fluids to Rigid and Deformable Solids and Shells. *ACM Trans. Graph.* 27, 3, Article 46 (2008), 9 pages.

Morten Silcowitz, Sarah Niebe, and Kenny Erleben. 2009. Nonsmooth Newton Method for Fischer Function Reformulation of Contact Force Problems for Interactive Rigid Body Simulation. *VRIPHYS 2009 - 6th Workshop on Virtual Reality Interactions and Physical Simulations*, 105–114.

Morten Silcowitz, Sarah Niebe, and Kenny Erleben. 2010. Projected Gauss-Seidel Subspace Minimization Method for Interactive Rigid Body Dynamics - Improving Animation Quality using a Projected Gauss-Seidel Subspace Minimization Method. *GRAPP 2010 - Proceedings of the International Conference on Computer Graphics Theory and Applications* 229, 38–45.

Morten Silcowitz-Hansen, Sarah Niebe, and Kenny Erleben. 2010. A nonsmooth nonlinear conjugate gradient method for interactive contact force problems. *The Visual Computer* 26 (2010), 893–901.

Russell Smith. 2008. Open Dynamics Engine. http://www.ode.org/

David E. Stewart. 2000. Rigid-Body Dynamics with Friction and Impact. *SIAM Rev.* 42, 1 (2000), 3–39.

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.

Tetsuya Takahashi and Christopher Batty. 2020. Monolith: a monolithic pressure-viscosity-contact solver for strong two-way rigid-rigid rigid-fluid coupling. *ACM Transactions on Graphics (TOG)* 39, 6 (2020), 1–16.

Tetsuya Takahashi and Ming C. Lin. 2019. A Geometrically Consistent Viscous Fluid Solver with Two-Way Fluid-Solid Coupling. *Computer Graphics Forum* 38, 2 (2019), 49–58.

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 (2017), 105:1–105:11 pages.
Jie Tan, Kristin Siu, and C. Karen Liu. 2012. *Contact Handling for Articulated Rigid Bodies Using LCP.* Technical Report GIT-GVU-15-01-2. Georgia Institute of Technology, School of Interactive Computing.
Yun Teng, David I. W. Levin, and Theodore Kim. 2016. Eulerian Solid-Fluid Coupling. *ACM Trans. Graph.* 35, 6, Article 200 (2016), 8 pages.
Joseph Teran, Eftychios Sifakis, Geoffrey Irving, and Ronald Fedkiw. 2005. Robust Quasistatic Finite Elements and Flesh Simulation. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '05).* 181–190.
Emanuel Todorov. 2010. Implicit nonlinear complementarity: A new approach to contact dynamics. *Proceedings - IEEE International Conference on Robotics and Automation,* 2322 – 2329.
Emanuel Todorov. 2011. A convex, smooth and invertible contact model for trajectory optimization. 1071–1076.
Richard Tonge, Feodor Benevolenski, and Andrey Voroshilov. 2012. Mass Splitting for Jitter-Free Parallel Rigid Body Simulation. *ACM Transactions on Graphics* 31 (2012).
Mickeal Verschoor and Andrei C. Jalba. 2019. Efficient and Accurate Collision Response for Elastically Deformable Models. *ACM Trans. Graph.* 38, 2, Article 17 (March 2019), 20 pages.
Guowei Yan, Wei Li, Ruigang Yang, and Huamin Wang. 2018. Inexact Descent Methods for Elastic Parameter Optimization. In *SIGGRAPH Asia 2018 Technical Papers (SIGGRAPH Asia '18).* Article 253, 14 pages.
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 (2018), 19 pages.
Bo Zhu and Xubo Yang. 2010. Animating Sand as a Surface Flow. In *Eurographics 2010 - Short Papers.*
Kuixin Zhu, Xiaowei He, Sheng Li, Hongan Wang, and Guoping Wang. 2021. Shallow Sand Equations: Real-Time Height Field Simulation of Dry Granular Flows. *IEEE Transactions on Visualization and Computer Graphics* 27, 3 (2021), 2073–2084.
Yongning Zhu and Robert Bridson. 2005. Animating Sand As a Fluid. *ACM Trans. Graph.* 24, 3 (2005), 965–972.
Yufeng Zhu, Robert Bridson, and Danny M. Kaufman. 2018. Blended Cured Quasi-Newton for Distortion Optimization. *ACM Trans. Graph.* 37, 4, Article 40 (July 2018), 14 pages.

A DETAILS OF $\mathbf{A}_{\mathbf{pp}}, \mathbf{A}_{\mathbf{p}\boldsymbol{\lambda}}, \mathbf{A}_{\boldsymbol{\lambda}\boldsymbol{\lambda}}, \mathbf{b}_{\mathbf{p}}^k, \mathbf{b}_{\boldsymbol{\lambda}}^k$

$\mathbf{A}_{\mathbf{pp}}, \mathbf{A}_{\mathbf{p}\boldsymbol{\lambda}}, \mathbf{A}_{\boldsymbol{\lambda}\boldsymbol{\lambda}}, \mathbf{b}_{\mathbf{p}}^k, \mathbf{b}_{\boldsymbol{\lambda}}^k$ in (24) can be written as

$$\mathbf{A}_{\mathbf{pp}} = \Delta t \hat{\mathbf{G}}^T \mathbf{W}_F^u (\mathbf{P}^u \mathbf{W}_L^u)^{-1} \hat{\mathbf{G}} + \frac{\Delta t}{\tau} \hat{\mathbf{F}}_p^T \mathbf{M}_r^{-1} \hat{\mathbf{F}}_p, \tag{32}$$

$$\mathbf{A}_{\mathbf{p}\boldsymbol{\lambda}} = \Delta t \hat{\mathbf{F}}_p^T \mathbf{M}_r^{-1} \mathbf{B}_{\boldsymbol{\lambda}}, \tag{33}$$

$$\mathbf{A}_{\boldsymbol{\lambda}\boldsymbol{\lambda}} = \Delta t \tau \mathbf{B}_{\boldsymbol{\lambda}}^T \mathbf{M}_r^{-1} \mathbf{B}_{\boldsymbol{\lambda}}, \tag{34}$$

$$\mathbf{b}_{\mathbf{p}}^k = \hat{\mathbf{G}}^T \mathbf{W}_F^u \mathbf{u}^* - \hat{\mathbf{F}}_p^T \mathbf{v}^* - \mathbf{A}_{\mathbf{sp}}^T \mathbf{s}^{k+1} - \mathbf{A}_{\mathbf{pz}} \mathbf{z}^{k+1}, \tag{35}$$

$$\mathbf{b}_{\boldsymbol{\lambda}}^k = -\tau \mathbf{B}_{\boldsymbol{\lambda}}^T \mathbf{v}^* - \mathbf{A}_{\mathbf{s}\boldsymbol{\lambda}}^T \mathbf{s}^{k+1} - \mathbf{A}_{\boldsymbol{\lambda}\mathbf{z}} \mathbf{z}^{k+1}. \tag{36}$$

We note that while $\mathbf{A}_{\mathbf{pp}}, \mathbf{A}_{\mathbf{p}\boldsymbol{\lambda}}, \mathbf{A}_{\boldsymbol{\lambda}\boldsymbol{\lambda}}$ are constant within the staggered projections at each time step, $\mathbf{b}_{\mathbf{p}}^k$ and $\mathbf{b}_{\boldsymbol{\lambda}}^k$ need to be updated with the already computed $\mathbf{s}^{k+1}$ and $\mathbf{z}^{k+1}$.

B DETAILS OF $\mathbf{A}^l$ AND $\mathbf{b}^l$

To obtain the descent direction for $\mathbf{z}$ within the Newton iteration, we first need to assemble $\mathbf{A}_{\mathbf{zz}}^l$ and $\mathbf{b}_{\mathbf{z}}^l$, and these can be computed by

$$\mathbf{A}_{\mathbf{zz}}^l = \Delta t \tau \mathbf{B}_{\mathbf{z}}^T \mathbf{M}_r^{-1} \mathbf{B}_{\mathbf{z}} + \nabla^2 E_d(\boldsymbol{\lambda}^k, \mathbf{z}^l, \mathbf{h}^l), \tag{37}$$

$$\mathbf{b}_{\mathbf{z}}^l = -\tau \mathbf{B}_{\mathbf{z}}^T \mathbf{v}^* - \mathbf{A}_{\mathbf{sz},0}^T \mathbf{s}^k - \mathbf{A}_{\mathbf{pz},0}^T \mathbf{p}^k - \mathbf{A}_{\boldsymbol{\lambda}\mathbf{z},0}^T \boldsymbol{\lambda}^k - \mathbf{A}_{\mathbf{zz},0} \mathbf{z}^l - \nabla E_d(\boldsymbol{\lambda}^k, \mathbf{z}^l, \mathbf{h}^l). \tag{38}$$

Similarly, we can assemble $\mathbf{A}_{\mathbf{s}_{xx}\mathbf{s}_{xx}}^l$ and $\mathbf{b}_{\mathbf{s}_{xx}}^l$ as

$$\mathbf{A}_{\mathbf{s}_{xx}\mathbf{s}_{xx}}^l = \mathbf{A}_{\mathbf{s}_{xx}\mathbf{s}_{xx},0} + \nabla^2 E_c(\mathbf{s}_{xx}^l, \mathbf{p}^k, \mathbf{g}^l), \tag{39}$$

$$\mathbf{b}_{\mathbf{s}_{xx}}^l = \mathbf{b}_{\mathbf{s}_{xx},0} - \sum_e \mathbf{A}_{\mathbf{s}_{xx}\mathbf{s}_e,0} \mathbf{s}_e^l - \mathbf{A}_{\mathbf{s}_{xx}\mathbf{p},0} \mathbf{p}^k - \mathbf{A}_{\mathbf{s}_{xx}\boldsymbol{\lambda},0} \boldsymbol{\lambda}^k - \mathbf{A}_{\mathbf{s}_{xx}\mathbf{z},0} \mathbf{z}^{k+1} - \nabla E_c(\mathbf{s}_{xx}^l, \mathbf{p}^k, \mathbf{g}^l), \tag{40}$$

and we can compute $\mathbf{A}^l$ and $\mathbf{b}^l$ for $\mathbf{s}_{xy}, \mathbf{s}_{xz}, \mathbf{s}_{yz}, \mathbf{s}_{zz}$ in the same way. Here, componentwise submatrices are defined as

$$\mathbf{A}_{\mathbf{ss},0} = \begin{bmatrix} \mathbf{A}_{\mathbf{s}_{xx}\mathbf{s}_{xx},0} & \mathbf{A}_{\mathbf{s}_{xx}\mathbf{s}_{xy},0} & \mathbf{A}_{\mathbf{s}_{xx}\mathbf{s}_{xz},0} & \mathbf{A}_{\mathbf{s}_{xx}\mathbf{s}_{yz},0} & \mathbf{A}_{\mathbf{s}_{xx}\mathbf{s}_{zz},0} \\ \mathbf{A}_{\mathbf{s}_{xx}\mathbf{s}_{xy},0}^T & \mathbf{A}_{\mathbf{s}_{xy}\mathbf{s}_{xy},0} & \mathbf{A}_{\mathbf{s}_{xy}\mathbf{s}_{xz},0} & \mathbf{A}_{\mathbf{s}_{xy}\mathbf{s}_{yz},0} & \mathbf{A}_{\mathbf{s}_{xy}\mathbf{s}_{zz},0} \\ \mathbf{A}_{\mathbf{s}_{xx}\mathbf{s}_{xz},0}^T & \mathbf{A}_{\mathbf{s}_{xy}\mathbf{s}_{xz},0}^T & \mathbf{A}_{\mathbf{s}_{xz}\mathbf{s}_{xz},0} & \mathbf{A}_{\mathbf{s}_{xz}\mathbf{s}_{yz},0} & \mathbf{A}_{\mathbf{s}_{xz}\mathbf{s}_{zz},0} \\ \mathbf{A}_{\mathbf{s}_{xx}\mathbf{s}_{yz},0}^T & \mathbf{A}_{\mathbf{s}_{xy}\mathbf{s}_{yz},0}^T & \mathbf{A}_{\mathbf{s}_{xz}\mathbf{s}_{yz},0}^T & \mathbf{A}_{\mathbf{s}_{yz}\mathbf{s}_{yz},0} & \mathbf{A}_{\mathbf{s}_{yz}\mathbf{s}_{zz},0} \\ \mathbf{A}_{\mathbf{s}_{xx}\mathbf{s}_{zz},0}^T & \mathbf{A}_{\mathbf{s}_{xy}\mathbf{s}_{zz},0}^T & \mathbf{A}_{\mathbf{s}_{xz}\mathbf{s}_{zz},0}^T & \mathbf{A}_{\mathbf{s}_{yz}\mathbf{s}_{zz},0}^T & \mathbf{A}_{\mathbf{s}_{zz}\mathbf{s}_{zz},0} \end{bmatrix}, \tag{41}$$

$$\mathbf{A}_{\mathbf{sp},0} = \begin{bmatrix} \mathbf{A}_{\mathbf{s}_{xx}\mathbf{p},0} \\ \mathbf{A}_{\mathbf{s}_{xy}\mathbf{p},0} \\ \mathbf{A}_{\mathbf{s}_{xz}\mathbf{p},0} \\ \mathbf{A}_{\mathbf{s}_{yz}\mathbf{p},0} \\ \mathbf{A}_{\mathbf{s}_{zz}\mathbf{p},0} \end{bmatrix}, \mathbf{A}_{\mathbf{s}\boldsymbol{\lambda},0} = \begin{bmatrix} \mathbf{A}_{\mathbf{s}_{xx}\boldsymbol{\lambda},0} \\ \mathbf{A}_{\mathbf{s}_{xy}\boldsymbol{\lambda},0} \\ \mathbf{A}_{\mathbf{s}_{xz}\boldsymbol{\lambda},0} \\ \mathbf{A}_{\mathbf{s}_{yz}\boldsymbol{\lambda},0} \\ \mathbf{A}_{\mathbf{s}_{zz}\boldsymbol{\lambda},0} \end{bmatrix}, \mathbf{A}_{\mathbf{sz},0} = \begin{bmatrix} \mathbf{A}_{\mathbf{s}_{xx}\mathbf{z},0} \\ \mathbf{A}_{\mathbf{s}_{xy}\mathbf{z},0} \\ \mathbf{A}_{\mathbf{s}_{xz}\mathbf{z},0} \\ \mathbf{A}_{\mathbf{s}_{yz}\mathbf{z},0} \\ \mathbf{A}_{\mathbf{s}_{zz}\mathbf{z},0} \end{bmatrix}, \tag{42}$$

$$\mathbf{b}_{\mathbf{s},0} = \begin{bmatrix} \mathbf{b}_{\mathbf{s}_{xx},0} \\ \mathbf{b}_{\mathbf{s}_{xy},0} \\ \mathbf{b}_{\mathbf{s}_{xz},0} \\ \mathbf{b}_{\mathbf{s}_{yz},0} \\ \mathbf{b}_{\mathbf{s}_{zz},0} \end{bmatrix}, \tag{43}$$

where

$$\mathbf{A}_{\mathbf{ss},0} = \Delta t \hat{\mathbf{D}}^T \mathbf{W}_F^u (\mathbf{P}^u \mathbf{W}_L^u)^{-1} \hat{\mathbf{D}} + \frac{\Delta t}{\tau} \hat{\mathbf{F}}_s^T \mathbf{M}_r^{-1} \hat{\mathbf{F}}_s, \tag{44}$$

$$\mathbf{A}_{\mathbf{sp},0} = \Delta t \hat{\mathbf{D}}^T \mathbf{W}_F^u (\mathbf{P}^u \mathbf{W}_L^u)^{-1} \hat{\mathbf{G}} + \frac{\Delta t}{\tau} \hat{\mathbf{F}}_s^T \mathbf{M}_r^{-1} \hat{\mathbf{F}}_p, \tag{45}$$

$$\mathbf{A}_{\mathbf{s}\boldsymbol{\lambda},0} = \Delta t \hat{\mathbf{F}}_s^T \mathbf{M}_r^{-1} \mathbf{B}_{\boldsymbol{\lambda}}, \tag{46}$$

$$\mathbf{A}_{\mathbf{sz},0} = \Delta t \hat{\mathbf{F}}_s^T \mathbf{M}_r^{-1} \mathbf{B}_{\mathbf{z}}, \tag{47}$$

$$\mathbf{b}_{\mathbf{s},0} = \hat{\mathbf{D}}^T \mathbf{W}_F^u \mathbf{u}^* - \hat{\mathbf{F}}_s^T \mathbf{v}^*. \tag{48}$$