

SPH Granular Flow with Friction and Cohesion

Iván Alduán Miguel A. Otaduy

URJC Madrid, Spain

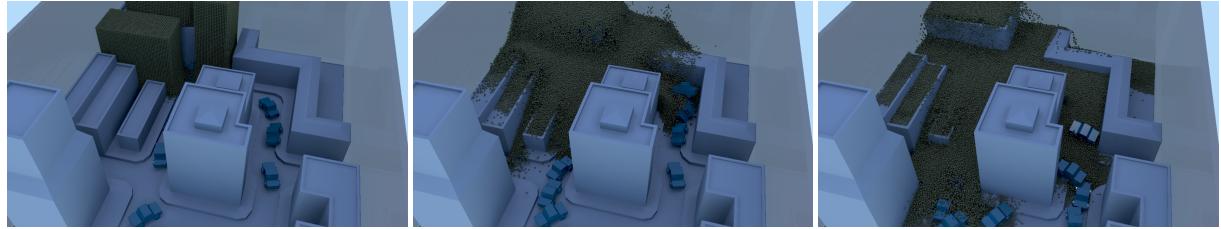


Figure 1: An avalanche of sand flooding a city populated with dynamic objects.

Abstract

Combining mechanical properties of solids and fluids, granular materials pose important challenges for the design of algorithms for realistic animation. In this paper, we present a simulation algorithm based on smoothed particle hydrodynamics (SPH) that succeeds in modeling important features of the behavior of granular materials. These features are unilateral incompressibility, friction and cohesion. We extend an existing unilateral incompressibility formulation to be added at almost no effort to an existing SPH-based algorithm for fluids. The main advantages of this extension are the ease of implementation, the lack of grid artifacts, and the simple two-way coupling with other objects. Our friction and cohesion models can also be incorporated in a seamless manner in the overall SPH simulation algorithm.

1. Introduction

Granular materials exhibit a large range of interesting macroscopic phenomena, including piling, flow or fracture. Effects such as piling may be well described using large amounts of small rigid bodies, with inter-body frictional contact governing the geometry of piles. Flow, on the other hand, may be well described by incompressible fluid models.

However, both rigid-body and fluid models suffer important limitations when applied to granular media. Rigid body models suffer an explosion of complexity, because classic non-penetration and friction constraints have to be resolved at the scale of fine particles. Common fluid models, such as incompressible Navier-Stokes equations, do not capture well the behavior of dispersing granular media or constant pressure distribution on grain silos. Recently, Narain et al. [NGL10] have formulated a continuum model for granu-

lar flow that succeeds at modeling dispersing and frictional behavior, as well as constant pressure distribution in a sand watch. They modify the traditional incompressible Navier-Stokes equations to pose incompressibility as a unilateral constraint, and they describe how to discretize and solve the modified equations on an Eulerian grid.

The first contribution of our work is the observation that unilateral incompressibility can be easily adapted to the predictive-corrective incompressible SPH (PCISPH) framework for fluid simulation [SP09], as we describe in Section 3. The major benefits of this adaptation to PCISPH are ease of implementation and lack of grid artifacts.

The second contribution of our work is the definition of continuum models for friction and cohesion, presented respectively in Section 4 and Section 5. Our friction model dissipates only relative velocity, and our cohesion model coun-

teracts material scattering until a debonding limit is reached. Both the friction and cohesion models can be simulated in a predictive-corrective fashion, therefore we integrate them in the PCISPH framework.

Our results show dynamic avalanches, interesting motion due to frictional contact, and blocking and fracture-like effects due to cohesion. We also show the possible range of behaviors that may be obtained with the friction and cohesion models.

2. Related Work

Simulation of Granular Media The simulation of granular materials has a large impact in the engineering field, for the analysis of terrains and avalanches, but also in the animation industry [ABC^{*}07, ABCT07]. For an extensive treatment of the major simulation methods, please refer to the book of Pöschel and Schwager [PS05].

Most of the algorithms for simulating granular media belong to two large categories: discrete and continuum methods. Discrete methods model the internal forces of a granular medium through the contact interactions between small dynamic particles. Early approaches modeled spherical particles interacting through penalty forces [CS79], with the later addition of friction [LH93]. Others have also investigated the use of non-spherical particles [PB93, PB95]. Following this approach, Bell et al. [BYM05] introduced a technique for computer animation that models particles as oriented rigid bodies composed of several spheres. The major challenge with discrete methods is the high cost required to compute contact dynamics on densely sampled particle sets. Multi-scale techniques try to address this complexity issue [LHM95], for example through coarse-scale computation of internal forces combined with fine-scale computation of external forces [ATO09].

Continuum methods define the behavior of a granular material through mass and momentum balance laws, combined with a constitutive model of internal stress. We are mostly interested in fluid-like models used in the animation field. Zhu and Bridson [ZB05] simulated granular media with the FLIP algorithm. They augmented the Navier-Stokes fluid equations with friction forces and a method to handle rigid blocks of material. Lenaerts and Dutré [LD09] used a similar model with SPH discretization, and they integrated granular materials and fluids with two-way coupling. Recently, Narain et al. [NGL10] have presented an important modification to the fluid-like model of granular materials, turning incompressibility into a unilateral constraint. Their approach produces rich free-flowing granular motion comparable to the one in discrete methods, but at a much lower computational cost. We extend the unilateral incompressibility constraint to a SPH algorithm. This is done with minor modifications to the standard algorithm, and our results do not suffer from the grid artifacts in the work of Narain et al. Additionally,

we propose a friction model that dissipates relative velocity, not absolute velocity like Narain's, and we combine friction with cohesion.

SPH Fluids The SPH method was designed originally to model fluid dynamics in astrophysics [Mon92]. The method was adopted in computer graphics to simulate fire and smoke [SF95], highly deformable solids [DG96] and viscous fluids like lava [SAC^{*}99]. The method was first used to simulate largely inviscid fluids in computer graphics by Müller et al. [MCG03], achieving interactive simulation rates. One of the interesting features of particle-based fluid simulation methods is that they enjoy attractive features for coupling animations of different media, all under a Lagrangian discretization. Examples of methods to couple SPH fluids to other media include the interaction with deformable solids [MST^{*}04], general fluid-solid coupling [SSP07, BTT09], a uniform fluid-and-solid simulation framework based on co-rotated SPH [BIT09], sand-water interaction [RSKN08], or the coupling of fluids and granular media [LD09].

One of the major drawbacks of the original SPH method was the challenge to model strongly incompressible fluids. Early approaches computed pressure using a penalty function on density deviations, which requires very small time steps for high incompressibility. A variation to SPH, called the Moving-Particle Semi-Implicit method [PTB^{*}03] discretizes the fluid equations using particles and kernel functions, but solves incompressibility with a Poisson solve on the particle-based discretization. Solenthaler and Pajarola [SP09] introduced the PCISPH method, which iterates pressure adjustments to project the density to acceptable values. Later, PCISPH has been improved to handle boundaries more robustly and to allow adaptive time-stepping [IAGT10].

Other efforts on SPH simulation have been directed to accelerating computations, either through adaptive sampling [APKG07], or through highly parallel algorithms executed on graphics processors [KSW04, HKK07, YHK08].

3. Granular Flow Using PCISPH

In this section, we first describe the basic laws that model the behavior of a granular material in the continuum. Then, we introduce the SPH discretization of those equations. Our algorithm diverts from previous work in the way it handles unilateral incompressibility in the context of the PCISPH algorithm.

3.1. Model of Granular Flow

In the continuum, we describe the behavior of a granular material using a modified version of the Navier-Stokes equations. Conservation of momentum is expressed as

$$\rho \frac{D\mathbf{u}}{Dt} = \mathbf{F}_{\text{ext}} - \nabla p + \nabla \cdot \mathbf{s}, \quad (1)$$

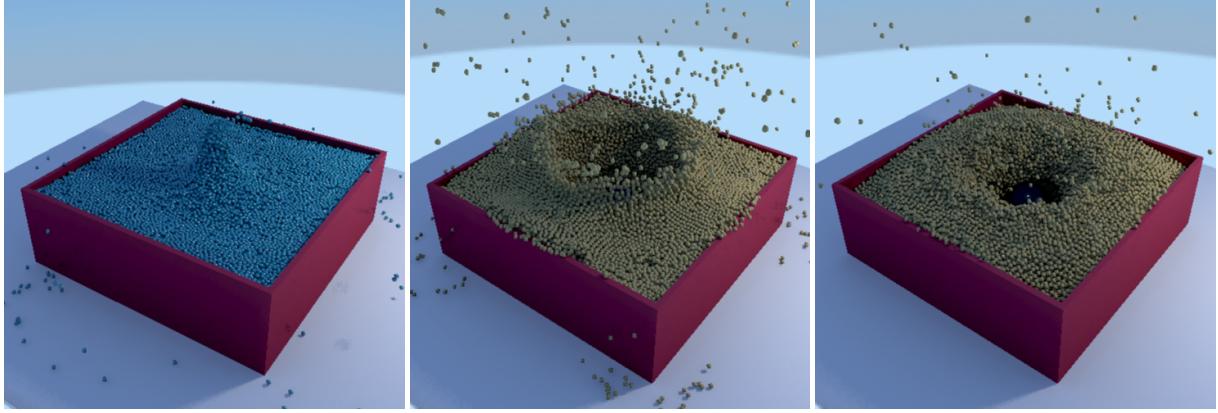


Figure 2: A ball falling into a box full of particles. On the left, with bilateral incompressibility as in regular PCISPH; in the center, with our unilateral incompressibility algorithm; and on the right, with the addition of friction.

where $\frac{D\mathbf{u}}{Dt}$ denotes the Lagrangian derivative of the velocity field, ρ the density field, \mathbf{F}_{ext} external forces, p the pressure, and \mathbf{s} a stress tensor capturing friction and cohesion effects. In this section, we will describe the discretization of the equations ignoring the friction and cohesion term. These two effects will be described, respectively, in Sections 4 and 5.

In addition to momentum conservation, the granular flow is governed by unilateral incompressibility [NGL10]. This is formulated through two complementary inequality constraints, one enforcing positive pressure, the other enforcing density lower than ρ_{\max} , a critical density of the material that allows free flow:

$$\rho \leq \rho_{\max} \perp p \geq 0. \quad (2)$$

Complementarity implies that pressure forces are present only when the critical density is reached.

3.2. SPH Discretization

In SPH, the density and the forces acting on a particle are computed by a weighted sum of the contributions of neighboring particles. Next, we define the specific SPH discretizations relevant to our model. The density of a particle i at position \mathbf{x}_i is computed as

$$\rho_i = \sum_j m_j W_{ij}, \quad (3)$$

where $W_{ij} = W(\mathbf{x}_i - \mathbf{x}_j, h)$ is the smoothing kernel with support h , and evaluated for particles i and j . m_j is the mass of the j^{th} particle.

The discretization of the pressure force involves the discretization of the pressure gradient. We follow the symmetric

formulation of Monaghan [Mon92]:

$$\mathbf{F}_{i,\text{pressure}} = -m_i \sum_j m_j \left(\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2} \right) \nabla W_{ij}. \quad (4)$$

The discretization of the friction and cohesion force involves the discretization of the divergence of a tensor. This turns into a sum of products between stress tensors and kernel gradients. Again, we apply a symmetric formulation:

$$\mathbf{F}_{i,\text{friction \& cohesion}} = m_i \sum_j m_j \left(\frac{\mathbf{s}_i}{\rho_i^2} + \frac{\mathbf{s}_j}{\rho_j^2} \right) \nabla W_{ij}. \quad (5)$$

Finally, for the computation of strain rate, we need to discretize the velocity gradient. This turns into a sum of tensor products (i.e., outer products) between kernel gradients and velocities:

$$\nabla \mathbf{u}_i = \sum_j \frac{m_j}{\rho_j} \nabla W_{ij} \mathbf{u}_j^T. \quad (6)$$

In our implementation, we have used the same kernels (for density computation and for gradients) as in the work of Müller et al. [MCG03].

3.3. Unilateral Incompressibility in PCISPH

The PCISPH algorithm for incompressible fluids begins each simulation step by updating particle neighborhoods, evaluating external forces, gravity and viscosity, and resetting pressure to zero. Then, the algorithm performs Jacobi-style iterations over the pressure until the nominal density ρ_{\max} of the fluid is reached. In each iteration, the algorithm first applies the current pressure force and estimates particle velocity and position. Given the estimated positions, it predicts particle density and computes a correction Δp to the current pressure. In particular, the pressure correction for a



Figure 3: Granular piles under three different coefficients of friction.

particle with estimated density ρ_i is given by

$$\Delta p_i = \delta \cdot (\rho_i - \rho_{\max}), \quad (7)$$

$$\delta = \frac{\rho_0^2}{2m^2 \Delta t^2 \left(\sum_j \nabla W_{ij}^T \sum_j \nabla W_{ij} + \sum_j \nabla W_{ij}^T \nabla W_{ij} \right)}.$$

The corrective factor δ is precomputed only once for a prototype particle with a filled neighborhood. Please refer to [SP09] for the details about the PCISPH algorithm. In Section 6 we outline our full predictive-corrective algorithm, including unilateral incompressibility, friction and cohesion.

From an algorithmic point of view, the unilateral incompressibility constraint from Eq. (2) can be trivially incorporated into a predictive-corrective SPH algorithm. During each iteration, we first predict the density $\rho_{i,0}$ for each particle i . If the density is higher than ρ_{\max} , then we compute a corrective pressure increment Δp_i as in the original PCISPH algorithm. However, if the density is lower than ρ_{\max} the material is free to flow, and the pressure is not corrected. This modification to PCISPH is highlighted in lines 12 to 15 in the Algorithm in Section 6.

With unilateral incompressibility, pressure remains zero in low-density regions, and particles are free to move and get arbitrarily close to each other. This poses a problem if density increases again, as the spiky kernel used for pressure force computation will introduce sudden large pressure forces. We limit inter-particle distance at low density regions, applying low-stiffness discrete-particle forces [BYM05] if particles get closer than half the kernel radius, $h/2$. Although our discrete-particle solution works in practice and is inexpensive, it would be desirable to find a more elegant solution that naturally detects a density increase when two particles get very close to each other.

4. Predictive-Corrective Formulation of Friction

We derive our friction model from the principle of maximum dissipation. In particular, we propose to express maximum dissipation as the minimization of the strain rate, i.e., $\min \|\dot{\epsilon}\|$, as this is a measure of relative velocity. This formulation is different from the one followed by Narain et al. [NGL10], who minimize kinetic energy, which is akin to minimizing absolute velocities.



Figure 4: A block of particles retains its shape under higher cohesion (from left to right).

Following the formulation of Narain et al., we express frictional forces $\nabla \cdot \mathbf{s}$ from a traceless deviatoric stress tensor \mathbf{s} . Friction is limited by pressure p by the Drucker-Prager yield criterion, $\|\mathbf{s}\|_F \leq \sqrt{3}\alpha \cdot p$, with $\|\mathbf{s}\|_F = \sqrt{\sum s_{ij}^2}$ the Frobenius norm of \mathbf{s} and the frictional coefficient $\alpha = \sqrt{2/3} \sin \theta$ for an angle of repose θ . Same as Narain et al., we also approximate the yield constraint in a piecewise linear manner on each component s_{ij} of \mathbf{s} , and then our friction model in the continuum can be summarized as

$$\min \|\dot{\epsilon}\|, \text{ s.t. } \|s_{ij}\| \leq \alpha \cdot p. \quad (8)$$

In a predictive-corrective setting, the goal is to estimate a friction stress \mathbf{s} that will *correct* a *predicted* frictionless strain rate $\dot{\epsilon}_0$. For this, we follow largely a derivation analogous to the one in the PCISPH algorithm for density correction. Friction forces aim to produce a change of strain $\Delta \dot{\epsilon}$, such that $\dot{\epsilon} = \dot{\epsilon}_0 + \Delta \dot{\epsilon} = 0 \Rightarrow \Delta \dot{\epsilon} = -\dot{\epsilon}_0$. Given the symmetric definition of strain rate, $\dot{\epsilon} = 1/2 (\nabla \mathbf{u} + \nabla \mathbf{u}^T)$, the dissipation of strain rate is achieved by a change in the velocity gradient,

$$\Delta \nabla \mathbf{u} = -\dot{\epsilon}_0. \quad (9)$$

In the SPH setting, the change of velocity gradient of a particle i can be expressed in terms of the velocity change of its neighboring particles, from Eq. (6), as:

$$\Delta \nabla \mathbf{u}_i = \nabla \mathbf{u}_i - \nabla \mathbf{u}_{i,0} = \sum_j \frac{m_j}{\rho_j} \nabla W_{ij} \mathbf{u}_j^T - \nabla \mathbf{u}_{i,0}$$

$$\Rightarrow \Delta \nabla \mathbf{u}_i = -\dot{\epsilon}_{i,0} = \sum_j \frac{m_j}{\rho_j} \nabla W_{ij} \Delta \mathbf{u}_j^T. \quad (10)$$

Similarly, the corrective frictional force $\nabla \cdot \Delta \mathbf{s}$ on a particle j can be expressed in terms of the frictional stress of neighboring particles by discretization of the tensor divergence operator, as written in Eq. (5). Similar to the Jacobi-style iterations of PCISPH, in the corrective update of the frictional stress for particle i , we discard the change of frictional stress on all other particles (except for $\Delta \mathbf{s}_j = \Delta \mathbf{s}_i$). Then, the frictional force on particle j due to the corrective frictional stress at particle i can be approximated as

$$\mathbf{F}_j = -\frac{2m_i^2}{\rho_i^2} \Delta \mathbf{s}_i \nabla W_{ij}. \quad (11)$$

Through time integration, this frictional force produces a velocity change $\Delta\mathbf{u}_j = \frac{\Delta t}{m_j} \mathbf{F}_j$. Then, we can relate the frictional force in Eq. (11) to the dissipation of strain rate in Eq. (10) and obtain the expression for the corrective frictional stress:

$$\Delta\mathbf{s}_i = \mathbf{D}^{-1} \dot{\varepsilon}_{i,0}, \quad \text{with } \mathbf{D} = \frac{2m_i^2 \Delta t}{\rho_i^2} \left(\sum_j \frac{1}{\rho_j} \nabla W_{ij} \nabla W_{ij}^T \right). \quad (12)$$

Analogous to the strategy followed for pressure correction by Solenthaler and Pajarola [SP09], we propose to precompute \mathbf{D} for a prototype particle with a filled neighborhood.

In the context of the full PCISPH algorithm, a corrective term $\Delta\mathbf{s}_i$ is added in each iteration to the frictional stress of the i^{th} particle for the present time step. Then, we enforce a traceless stress by subtracting $\frac{1}{3}\text{trace}(\mathbf{s}_i)\mathbf{I}_{3 \times 3}$ from the accumulated stress, and we apply to the result the yield condition from Eq. (8). Lines 18 to 21 of the modified PCISPH Algorithm in Section 6 show the application of our friction model.

5. Cohesion Model

Our approach to model cohesion is to consider it a dissipative effect, analogous to friction. In other words, maximum cohesion is also defined as maximum dissipation of the strain rate, and it is applied through the same stress tensor \mathbf{s} , but its magnitude is limited by different criteria. One interesting feature of this approach is that cohesion can easily be added to the PCISPH algorithm together with friction.

Specifically, we separate the stress tensor \mathbf{s} into the (traceless) deviatoric part $\mathbf{s} - \frac{1}{3}\text{trace}(\mathbf{s})\mathbf{I}_{3 \times 3}$, and a mean hydrostatic part $\frac{1}{3}\text{trace}(\mathbf{s})\mathbf{I}_{3 \times 3}$. While friction is computed from a deviatoric stress (i.e., the mean hydrostatic part is canceled), cohesion admits a mean hydrostatic stress too. Intuitively, this means that cohesion acts also on the normal direction of relative motion, not just on tangential directions. We model cohesion by minimization of the strain rate subject to a *maximum cohesion* constraint expressed on each component of the full stress tensor \mathbf{s} , i.e.,

$$\min \|\dot{\varepsilon}\|, \quad \text{s.t. } \|s_{ij}\| \leq \beta^2 C, \quad (13)$$

where β is the cohesion intensity, and C is a user-defined *maximum cohesion* property.

To combine friction and cohesion, we apply Eq. (8) on the deviatoric part of the stress, Eq. (13) on the full stress tensor, and then we pick the least restrictive value, i.e., the one with largest absolute value. In other words, we consider both friction and cohesion and we pick the one that maximizes dissipation. In the PCISPH algorithm, in each iteration of strain prediction and stress computation described in Section 4, we simply modify the application of the constraints on the components of the stress tensor \mathbf{s} .

To define the cohesion intensity, we borrow from adhesion

models in the contact mechanics literature [Wri02], which define a debonding function based on a thermodynamics model. The original model evolves the cohesion intensity based on the magnitude of a gap function at each contact. Instead, and since our framework lacks knowledge of the rest structure, we evolve the cohesion intensity as a function of the norm of the strain rate. Specifically, both for bonding and debonding, we have adapted the models by Gascón et al. [GZO10].

6. Algorithm Overview

Our algorithm corrects pressure and friction in a staggered manner, similar to the handling of normal contact forces and friction for solid bodies by Kaufman et al. [KSJP08]. The correction of pressure for all particles (and the same for friction) is done in a Jacobi-like manner. The predictive-corrective loop terminates when the density residual falls under a small tolerance value and the stress tensor of all particles varies between iterations less than a small threshold. We define the density residual as the largest excess density over all particles, i.e., $f_{\text{err}}(\rho) = \max_i (\max(\rho_i - \rho_{\max}, 0))$. We apply at least five iterations to ensure better propagation of forces.

To model two-way coupling between the granular medium and rigid bodies, we have implemented the direct forcing algorithm by Becker et al. [BTT09]. To handle friction with boundaries, both static and moving objects, we have used the Coulomb friction model as described by Zhu and Bridson [ZB05]. Finally, for SPH discretization at boundaries, we have applied *wall weight functions* as described by Harada et al. [HKK07].

Our complete simulation algorithm performs the following actions on each time step:

1. **for all** particles i **do**
2. Find neighborhoods N_i .
3. **for all** particles i **do**
4. Compute gravity, external forces and viscosity.
5. Initialize pressure $p_i = 0$ and dissipative stress $\mathbf{s}_i = 0$.
6. **while** $f_{\text{err}}(\rho) > \text{tol}_p$ OR $\max_i \|\Delta\mathbf{s}_i\|_F > \text{tol}_s$ **do**
7. **for all** particles i **do**
8. Add pressure and dissipative forces.
9. Predict velocity and position.
10. Apply boundary conditions.
11. **for all** particles i **do**
12. Predict density $\rho_{i,0}$.
13. **if** $\rho_{i,0} > \rho_{\max}$ **then**
14. Compute corrective pressure Δp_i (Eq. (7)).
15. Increment pressure $p_i \leftarrow p_i + \Delta p_i$.
16. **else** Add discrete-particle forces to close particles.
17. **for all** particles i **do**
18. Predict strain rate $\dot{\varepsilon}_{i,0}$.
19. Compute corrective dissipative stress $\Delta\mathbf{s}_i$ (Eq. (12)).
20. Increment dissipative stress $\mathbf{s}_i \leftarrow \mathbf{s}_i + \Delta\mathbf{s}_i$.
21. Test yield and cohesion, Eq. (8) and Eq. (13), on \mathbf{s}_i .



Figure 5: A sand castle remains upright thanks purely to our cohesion model, until a couple of balls tear it down.

7. Results

We have compared the behavior of our PCISPH-based simulator under different settings of incompressibility, friction and cohesion. Fig. 2 compares different splashes for bilateral incompressibility, unilateral incompressibility and with the addition of friction. The scattering of the splash is clearly different with unilateral incompressibility, enforcing the granular-like flow. Friction behaves robustly in our SPH setting. The behavior under different friction coefficients is highlighted in Fig. 3, which depicts very different piling profiles. Finally, the effect of varying cohesion coefficients is shown in Fig. 4. Note that, thanks to the dissipation of strain rate, which measures the symmetric part of the velocity gradient, our cohesion model preserves both linear and angular momentum.

Other examples depict the high applied potential of our algorithm, such as the sand avalanche example in Fig. 1 and the castle example in Fig. 5. In this last example, we applied a high cohesion coefficient to the castle, which keeps it straight at the beginning, but the impacts of two balls partially tear it down.

Table 1 shows performance statistics for several examples. All of them were executed on a 4-core 2.67 GHz Intel i7 720 machine with 6 GB of RAM, and the algorithm was programmed as a plugin for RealFlow 5. The table shows the number of particles, the time step size (in steps/frame at 24 fps), the average cost for a time step, and the average number of predictive-corrective iterations per time step, for the splash demo with friction (Fig. 2), the friction test with highest friction coefficient (Fig. 3), the avalanche demo (Fig. 1), and the sand castle demo (Fig. 5).

8. Discussion and Future Work

Our proposed algorithm for simulating sand and other granular media succeeds at capturing interesting dynamics aspects such as free-scattering flow, internal friction, or even cohesion. The continuum friction and cohesion models, although applied here in the context of PCISPH, could be extended to Eulerian discretizations. However, as demonstrated in the examples, one of the interesting features of the integration into the SPH framework is the possibility to adapt the granular flow to arbitrary boundaries without grid artifacts.

There are many lines of future work that could be ex-

Demo	particles	steps/frame	time/step [s]	iters/step
Friction	50k	30	0.19755	7.21
Splash	60k	30	0.12879	11.72
Avalanche	180k	25	0.84370	5.88
Sandcastle	20k	40	0.08409	12.04

Table 1: Performance statistics for several examples.

plored. The nature of the PCISPH algorithm, which is highly parallelizable, calls for implementation on massively parallel architectures such as GPUs. This could be just one way to accelerate the performance, but it would also be interesting to research novel algorithmic ways to address the predictive-corrective algorithm, perhaps allowing larger time steps. Indeed, we found that one possible limitation of our cohesion model is that, under very strong cohesion coefficients (such as the one needed for the castle example), the time step restrictions are higher.

Another limitation, inherent to the lack of a re-configuration structure, is that cohesion cannot be perfectly sustained over time, and it may suffer small drift. Moreover, since we do not model brittle fracture explicitly, the fracture behavior of the material does not match exactly that of dry granular media. Tuning the gain of the debonding model reduces the pseudo-elastic appearance of the material under cohesion, but this comes with time step restrictions.

Acknowledgments

The generous comments of the anonymous reviewers largely helped improve the quality of this paper. We would also like to thank Maribel Herreros for discussions about the friction model, Sara C. Schwartzman for demo production, and the rest of the members of the GMRV team at URJC Madrid. This work was funded in part by the TIN2009-07942 project of the Spanish Dept. of Science and Innovation.

References

- [ABC*07] AMMANN C., BLOOM D., COHEN J. M., COURTE J., FLORES L., HASEGAWA S., KALAITZIDIS N., TORNBERG T., TREWEEK L., WINTER B., YANG C.: The birth of sandman. In *ACM SIGGRAPH 2007 sketches* (2007). 2
- [ABCT07] ALLEN C., BLOOM D., COHEN J. M., TREWEEK

- L.: Rendering tons of sand. In *ACM SIGGRAPH 2007 sketches* (2007). [2](#)
- [APKG07] ADAMS B., PAULY M., KEISER R., GUIBAS L. J.: Adaptively sampled particle fluids. In *ACM SIGGRAPH 2007 papers* (2007). [2](#)
- [ATO09] ALDUÁN I., TENA A., OTADUY M. A.: Simulation of high-resolution granular media. In *Proceedings of the Spanish Computer Graphics Conference (CEIG)* (2009). [2](#)
- [BIT09] BECKER M., IHMSEN M., TESCHNER M.: Corotated SPH for deformable solids. In *Proceedings of the Eurographics Workshop on Natural Phenomena* (2009). [2](#)
- [BTT09] BECKER M., TESSENDORF H., TESCHNER M.: Direct forcing for lagrangian rigid-fluid coupling. *IEEE Transactions on Visualization and Computer Graphics* 15 (2009), 493–503. [2, 5](#)
- [BYM05] BELL N., YU Y., MUCHA P. J.: Particle-based simulation of granular materials. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2005). [2, 4](#)
- [CS79] CUNDALL P. A., STRACK O. D. L.: A discrete numerical model for granular assemblies. *Geotechnique* 29, 1 (1979), 47–65. [2](#)
- [DG96] DESBRUN M., GASCUEL M.-P.: Smoothed particles: A new paradigm for animating highly deformable bodies. In *Proceedings of the 6th Eurographics Workshop on Computer Animation and Simulation* (1996). [2](#)
- [GZO10] GASCÓN J., ZURDO J. S., OTADUY M. A.: Constraint-based simulation of adhesive contact. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2010). [5](#)
- [HKK07] HARADA T., KOSHIZUKA S., KAWAGUCHI Y.: Smoothed particle hydrodynamics on GPUs. In *Proceedings of Computer Graphics International* (2007). [2, 5](#)
- [IAGT10] IHMSEN M., AKINCI N., GISSLER M., TESCHNER M.: Boundary handling and adaptive time-stepping for PCISPH. In *7th Workshop on Virtual Reality Interaction and Physical Simulation* (2010). [2](#)
- [KSJP08] KAUFMAN D. M., SUEDA S., JAMES D. L., PAI D. K.: Staggered projections for frictional contact in multibody systems. In *ACM SIGGRAPH Asia 2008 papers* (2008). [5](#)
- [KSW04] KIPFER P., SEGAL M., WESTERMANN R.: Uberflow: A GPU-based particle engine. In *ACM SIGGRAPH/Eurographics Conference on Graphics Hardware* (2004). [2](#)
- [LD09] LENARTS T., DUTRÉ P.: Mixing fluids and granular materials. *Computer Graphics Forum* 28 (2009), 213–218. [2](#)
- [LH93] LEE J., HERRMANN H. J.: Angle of repose and angle of marginal stability: Molecular dynamics of granular particles. *Journal of Physics A* 26, 2 (1993), 373–383. [2](#)
- [LHM95] LUCIANI A., HABIBI A., MANZOTTI E.: A multi-scale physical model of granular materials. *Proceedings of Graphics Interface* (1995). [2](#)
- [MCG03] MÜLLER M., CHARYPAR D., GROSS M.: Particle-based fluid simulation for interactive applications. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2003). [2, 3](#)
- [Mon92] MONAGHAN J. J.: Smoothed particle hydrodynamics. *Annu. Rev. Astronomy and Astrophysics* 30 (1992), 543–574. [2, 3](#)
- [MST*04] MÜLLER M., SCHIRM S., TESCHNER M., HEIDELBERGER B., GROSS M.: Interaction of fluids with deformable solids. *Comput. Animat. and Virtual Worlds* 15, 3-4 (2004), 159–171. [2](#)
- [NGL10] NARAIN R., GOLAS A., LIN M. C.: Free-flowing granular materials with two-way solid coupling. In *ACM SIGGRAPH Asia 2010 papers* (2010). [1, 2, 3, 4](#)
- [PB93] PÖSCHEL T., BUCHHOLTZ V.: Static friction phenomena in granular materials: Coulomb law versus particle geometry. *Physical Review Letters* 71, 24 (1993), 3963–3966. [2](#)
- [PB95] PÖSCHEL T., BUCHHOLTZ V.: Molecular dynamics of arbitrarily shaped granular particles. *Journal of Physics I France* 5, 11 (1995), 1431–1455. [2](#)
- [PS05] PÖSCHEL T., SCHWAGER T.: *Computational Granular Dynamics: Models and Algorithms*. Springer, 2005. [2](#)
- [PTB*03] PREMOZE S., TASDIZEN T., BIGLER J., LEFOHN A. E., WHITAKER R. T.: Particle-based simulation of fluids. *Computer Graphics Forum* 22, 3 (2003), 401–410. [2](#)
- [RSKN08] RUNGJIRATANANON W., SZEGO Z., KANAMORI Y., NISHITA T.: Real-time animation of sand-water interaction. *Computer Graphics Forum* 27, 7 (2008), 1887–1893. [2](#)
- [SAC*99] STORA D., AGLIATI P. O., CANI M. P., NEYRET F., GASCUEL J.-D.: Animating lava flows. In *Proceedings of the 1999 Conference on Graphics Interface* (1999). [2](#)
- [SF95] STAM J., FIUME E. L.: Depicting fire and other gaseous phenomena using diffusion processes. In *Proceedings of SIGGRAPH 95* (Aug. 1995), Computer Graphics Proceedings, Annual Conference Series, pp. 129–136. [2](#)
- [SP09] SOLENTHALER B., PAJAROLA R.: Predictive-corrective incompressible SPH. In *ACM SIGGRAPH 2009 papers* (2009). [1, 2, 4, 5](#)
- [SSP07] SOLENTHALER B., SCHLÄFLI J., PAJAROLA R.: A unified particle model for fluid-solid interactions. *Computer Animation and Virtual Worlds* 18 (2007), 69–82. [2](#)
- [Wri02] WRIGGERS P.: *Computational Contact Mechanics*. Wiley, 2002. [5](#)
- [YHK08] YASUDA R., HARADA T., KAWAGUCHI Y.: Real-time simulation of granular materials using graphics hardware. In *Proceedings of the 5th International Conference on Computer Graphics, Imaging and Visualisation* (2008). [2](#)
- [ZB05] ZHU Y., BRIDSON R.: Animating sand as a fluid. In *ACM SIGGRAPH 2005 papers* (2005). [2, 5](#)