

MIT Open Access Articles

Fast spot-based multiscale simulations of granular drainage

The MIT Faculty has made this article openly available. **Please share** how this access benefits you. Your story matters.

Citation: Rycroft, Chris H., Yee Lok Wong, and Martin Z. Bazant. "Fast spot-based multiscale simulations of granular drainage." Powder Technology 200.1-2 (2010): 1-11.

As Published: 10.1016/j.powtec.2010.01.009

Publisher: Elsevier

Persistent URL: <http://hdl.handle.net/1721.1/59277>

Version: Author's final manuscript: final author's manuscript post peer review, without publisher's formatting or copy editing

Terms of use: Attribution-Noncommercial-Share Alike 3.0 Unported



Fast spot-based multiscale simulations of granular drainage

Chris H. Rycroft^{a,b}, Yee Lok Wong^c, Martin Z. Bazant^{c,d}

^a*Department of Mathematics, University of California, Berkeley, CA 94720*

^b*Department of Mathematics, Lawrence Berkeley Laboratory, Berkeley, CA 94720*

^c*Department of Mathematics, Massachusetts Institute of Technology, Cambridge, MA 02139*

^d*Department of Chemical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139*

Abstract

We develop a multiscale simulation method for dense granular drainage, based on the recently proposed spot model, where the particle packing flows by local collective displacements in response to diffusing “spots” of interstitial free volume. By comparing with discrete-element method (DEM) simulations of 55,000 spheres in a rectangular silo, we show that the spot simulation is able to approximately capture many features of drainage, such as packing statistics, particle mixing, and flow profiles. The spot simulation runs two to three orders of magnitude faster than DEM, making it an appropriate method for real-time control or optimization. We demonstrate extensions for modeling particle heaping and avalanching at the free surface, and for simulating the boundary layers of slower flow near walls. We show that the spot simulations are robust and flexible, by demonstrating that they can be used in both event-driven and fixed timestep approaches, and showing that the elastic relaxation step used in the model can be applied much less frequently and still create good results.

Keywords: granular materials, numerical methods

Email addresses: chr@math.berkeley.edu (Chris H. Rycroft), ylwong@mit.edu (Yee Lok Wong), bazant@mit.edu (Martin Z. Bazant)

1. Introduction

Particle-based simulation of slow, dense granular flow is needed in many engineering applications, but presents a difficult computational challenge. One simulation approach is the discrete-element method (DEM) [1], whereby individual particles are integrated according to Newton’s laws with a contact force model, but simulating realistic three-dimensional flows still requires days to weeks on a parallel computer. While this method is useful in-depth analysis, it is impractical in certain situations, such as for process control, where it may be advantageous to estimate features of a flow in real-time, or for optimization, where a large number of varying configurations may need to be considered.

In this paper, we develop a multiscale simulation technique that can be used to rapidly simulate many features of dense granular drainage. The model is simple and easy to implement, and can approximate both microscopic and macroscopic flow features, using two to three orders of magnitude less computational power than DEM. A key strength of the simulation is its ability to model granular mixing, for which relatively few descriptions are available. In some industrial hopper flows, where several granular materials of different compositions are draining through a single hopper, it may be important to estimate in real-time how much mixing is taking place. One example of where this occurs is the pebble-bed nuclear reactor concept [2, 3], that features a reactor core comprised of spherical fuel pebbles of diameter ~ 6 cm that are slowly cycled. Some designs, such as the MIT Modular Pebble-Bed Reactor (MPBR) [4], feature an additional type of reflecting moderator pebbles, and the amount of moderator/fuel pebble mixing has direct implications on reactor power output and fuel burnup [5, 6]. Mixing in this geometry has been investigated using DEM [7], but even simulating a single cycle of this large-scale, three-dimensional geometry took several weeks of time on a parallel computer.

The physics of mixing in slow, dense granular flow is significantly different from traditional models of diffusion of gases and liquids. In these slow, dense flows, particles move quasi-statically and experience long-lasting contacts with their neighbors. Kinetic stresses (based upon the variance of particle velocities) are small in comparison to the static stresses (based upon the particle contact forces). Mixing in this regime been investigated experimentally by Choi et al. [8] using 3 cm glass beads in a thin rectangular silo. These experiments pointed to a rate-independence for granular diffusion: if

the total drainage rate was changed, then the amount of mixing over the course of the run would remain the same, thus being controlled by the total deformation, as opposed to time. It was also observed that the total amount of particle mixing in these experiments was very small, on a scale of two to three orders of magnitude less than the system size. Particles have persistent cages of neighbors, with a single particle keeping more than 90% of its neighbors even after a large amount of flow.

A number of lattice-based models have been proposed for approximately simulating granular drainage. Perhaps the simplest is the void model [9, 10, 11] where particles lie on a hexagonal lattice, and move in response to voids of empty space that are introduced at the orifice that propagate upwards through the material according to a random walk. Taking the continuum limit of this model shows that the mean vertical velocity v_z follows a diffusion equation of the form $\partial_z v_z = b \partial_{xx} v_z$. The kinematic model [12, 13] derives this same equation purely from continuum considerations, and it leads to Gaussian velocity profiles and a parabolic region of flow, which is in reasonable agreement with experimental measurements [14, 15, 16]. More recently, similar ideas have been employed in cellular automata models [17, 18, 19]. However, all of these models have a fundamental problem when estimating mixing, that whenever a particle moves from one lattice site to another, it necessarily loses contact with many of its neighbors, violating the slow cage-breaking seen in experiment. Indeed, a continuum analysis of diffusion in the void model shows that the length scale associated with particle diffusion would exactly match the length scale of the flow width b [20].

Motivated by these observations, Bazant proposed the spot model for random packing dynamics [21, 22]. In this approach, particles are held off-lattice, and motion is mediated by “spots”, which represent a region of free interstitial space spread across several particle diameters, as shown by the blue circle in Fig. 1(a). When the spot moves according to the blue arrow, it induces a small, correlated motion of all particles within range in the opposite direction. This model is simple enough for mathematical analysis [21], and predicts the correct magnitude for particle diffusion: since spots cause particles to move co-operatively with their neighbors, cage-breaking occurs much less frequently. However, simulations based on Fig. 1(a) do not enforce packing constraints, which, over time, result in unphysical packings. In order to preserve valid packings, a second step was proposed, whereby a small elastic relaxation is applied, during which the particles and their nearest neighbors experience a soft-core repulsion with each other, as shown in

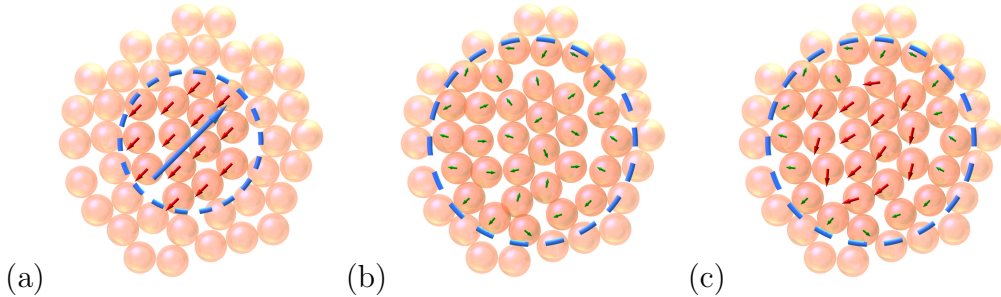


Figure 1: The mechanism for structural rearrangement in the spot model. The random displacement of a diffusing spot of free volume (dashed circle) causes affected particles to move as a block by an amount (a), followed by a relaxation under soft-core repulsion forces(b); the net co-operative motion combining these two steps (c). (Particle displacements are exaggerated for clarity.) Figure from Ref. [22].

Fig. 1(b). This is done on a purely geometrical basis, and no mechanical quantities such as contact forces, energy, or momentum are considered. The net effect, as shown in Fig. 1(c), is then a co-operative local deformation, whose mean is roughly the original block motion.

This model formed the basis of a multiscale simulation technique that was demonstrated by Rycroft et al. [22] to reproduce granular drainage. A DEM drainage simulation in a rectangular silo was carried out, and a systematic procedure was then derived to fit several free parameters in the spot simulation, based upon physical measurements from DEM (such as velocity correlation measurements, particle diffusion, and total flow). A spot simulation was then run by introducing spots at the orifice, and having them propagate upwards according to a random walk. The spot simulation reproduced many features of the granular packing, including mean velocity profiles, particle diffusion, and velocity correlations. In addition, the simulation recreated statistical signatures of the particle packings, such as the radial distribution function $g(r)$ and the bond angle distribution function $g_3(\theta)$.

In order to simulate the flow of a given initial packing using the spot model, all that remains is to specify the statistical dynamics of spots. Although the microscopic deformation of the particle packing is determined entirely by geometrical constraints, the mesoscale dynamics of spots reflects overall mechanical response, specific to the material. One such theoretical framework has recently been developed by Kamrin and Bazant [23], based on the hypothesis of a “stochastic flow rule” (SFR) for limit-state plasticity,

where spots perform random walks along slip lines biased by local stress imbalances upon fluidization (localized yielding). In the case of Mohr–Coulomb (MC) plasticity for two-dimensional granular material at incipient yield, they derive a simple theory of spot drift, by assuming that fluidization leads to a localized change in the friction coefficient from the static value to a lower dynamic value, which disrupts mechanical equilibrium; the resulting force on particles affected by the spot causes them to move in a collective fashion, thereby propelling the spot in the opposite direction.

The MC/SFR theory provides a possible mechanical basis for the spot model, which can be implemented in (at least) two ways: (i) The continuum limit can be taken to obtain a drift-diffusion (Fokker–Planck) equation for the spot density, which closes the model by connecting stresses to the mean velocity field and particle diffusion [23], or (ii) the spot drift vector field obtained from the stresses can be used in discrete spot-based multiscale simulations [24]. These approaches provide an appealing, unified description of some previously distinct granular flows, notably silo drainage under gravity and shear flow in a Couette cell with a moving, rough inner cylinder [23, 24]. In the case of the continuum model, these flows are accurately predicted by introducing only one new parameter, the spot size (or velocity correlation length), which is obtained from explicit measurements and not adjusted to fit velocity profiles. A recent analysis of continuum variables in a wide variety of DEM simulations, however, casts doubt on the validity of certain hypotheses in the SFR theory [25] and points toward some more complicated phenomena that may eventually need to be incorporated into the theory.

Regardless of the general mechanical basis, it is clear that the random-walk based spot algorithm provides a very efficient means of approximately simulating a wide variety of hopper drainage problems. The aim of Rycroft’s study [22] was to validate the spot microscopic mechanism as a model for flowing random packings. In this paper, we focus instead on developing aspects of the model that would be useful in a practical context. We introduce the simulation method in Sec. 2, and then demonstrate extensions that can be used to model free surface behavior (Sec. 3) and boundary layer behavior (Sec. 4). We also demonstrate that the spot-based simulations are robust and flexible, showing that they can be used in both event-driven and fixed timestep approaches, and demonstrating that the elastic relaxation step can be applied much less frequently and while still creating very good results (Sec. 5).

2. Simulation overview

The simulations in this paper are carried out using monodisperse spheres of mass m and diameter d in a thin three-dimensional rectangular container, that makes use of the same geometry as in Ref. [22]. The container is constructed with walls at $x = \pm 25d$, $y = \pm 4d$, and $z = 0$. Gravity g acts in the negative z direction, which defines a natural time unit $\tau = \sqrt{d/g}$.

As a baseline for comparison, a DEM simulation was first carried out in this geometry, making use of the LAMMPS software package developed at Sandia National Laboratories [26]. In this simulation, particles interact according to the Cundall–Strack contact model [27] for cohesionless particulates that makes use of Hertzian, history-dependent contact forces. If a particle and its neighbor are separated by \vec{r} , and they are in compression, so that $\delta = d - |\vec{r}| > 0$, then they experience a force $\vec{F} = \vec{F}_n + \vec{F}_t$, where the normal and tangential components are given by

$$\vec{F}_n = \sqrt{\delta/d} \left(k_n \delta \vec{n} - \frac{\gamma_n \vec{v}_n}{2} \right) \quad (1)$$

$$\vec{F}_t = \sqrt{\delta/d} \left(-k_t \Delta \vec{s}_t - \frac{\gamma_t \vec{v}_t}{2} \right) \quad (2)$$

Here, $\vec{n} = \vec{r}/|\vec{r}|$, and \vec{v}_n and \vec{v}_t are the normal and tangential components of the relative surface velocity. $\Delta \vec{s}_t$ is the elastic tangential displacement between spheres, obtained by integrating tangential relative velocities during elastic deformation for the lifetime of the contact. If $|\vec{F}_t| > \mu |\vec{F}_n|$, so that a local Coulomb yield criterion is exceeded, then \vec{F}_t is rescaled so that it has magnitude $\mu |\vec{F}_n|$ and $\Delta \vec{s}_t$ is modified so that equation 2 is upheld. Particle–wall interactions are handled with the same model, although the friction coefficient μ_w is set independently. The contact model is integrated using the Velocity Verlet scheme with timestep Δt , and the parameters used in the simulation are given in Table 1(a). The normal spring constant we use is four to five orders of magnitude smaller than what would be realistic for typical hard materials such as sand or glass, but is chosen this way for computational efficiency, since a significantly higher spring constant would need a much smaller timestep to be modeled effectively. It has been shown that the value of k_n we employ is a reasonable compromise, capturing the dynamics of the hard particles without too many detrimental elastic effects. For more detailed information, see Refs. [28, 29, 22].

An initial packing was created by pouring in 55,000 particles from $z = 160d$ and allowing them to come to rest, filling the container up to $z = 110d$. A drainage simulation was then carried out by opening a circular orifice of diameter $8d$ in the center of the container base, with snapshots of all particle positions being recorded at fixed intervals of 2τ . To effectively model hard particles, the normal spring interaction in these simulations is very stiff, requiring small timesteps to run, and thus the computations were carried out in parallel on 24 processors, taking three days to simulate the complete drainage of the packing. The DEM simulations have been shown to be in very good agreement with experimental data [30] for a wide variety of microscopic and macroscopic measurements.

The spot simulation was implemented in C++, with the main routine being written as part of a class that represents the entire simulation domain. For efficiency, the class divides the simulation up into a rectangular subgrid of regions, and keeps a separate list of position vectors of particles within each region. When particles are added to the container, they are sorted into the correct region. Two key routines are used to implement the spot microscopic mechanism. The first, $\text{spot}(\vec{p}, \vec{v}, r_s)$ applies the spot motion, by displacing all particles within a distance r_s of \vec{p} by an amount \vec{v} .

The second, $\text{relax}(\vec{p}, r_e)$ applies the elastic relaxation procedure to all particles within a distance r_e of \vec{p} . Suppose that the positions of particles within r_e of \vec{p} are labeled $\vec{x}_1, \vec{x}_2, \dots, \vec{x}_m$ and that all remaining particles are labeled $\vec{x}_{m+1}, \vec{x}_{m+2}, \dots, \vec{x}_n$. For two particles i and j , the amount of overlap is defined to be as $\delta_{ij} = \min\{d - |\vec{x}_i - \vec{x}_j|, 0\}$. If \hat{n}_{ij} is the normal vector pointing from \vec{x}_j to \vec{x}_i , then the relaxation displacement experienced by a particle i is given by

$$\Delta\vec{x}_i = \frac{1}{2} \sum_{j=1, j \neq i}^m \alpha \delta_{ij} \hat{n}_{ij} + \sum_{j=m+1}^n \alpha \delta_{ij} \hat{n}_{ij}. \quad (3)$$

Once all particles have been considered, these displacements are applied simultaneously. For this paper, we make use of $\alpha = 0.8$, but previous studies have shown that the physical results are largely insensitive to the parameter [22]. Typically, the particle displacements caused by a single **spot** operation are on the order of $10^{-3}d$ or less. Hence only a very small relaxation displacement is needed and the details of its operation are relatively unimportant. Also, throughout this paper, we carry out the relaxations within a radius $r_e = r_s + d$. After a **spot** event, the majority of the particle overlaps caused by a spot will be at the interface between the displaced particles and

Parameter	Value
Normal elastic constant k_n	$2 \times 10^5 mg/d$
Tangential elastic constant k_t	$2k_n/7$
Normal viscoelastic constant γ_n	$50\tau^{-1}$
Tangential viscoelastic constant γ_t	$50\tau^{-1}$
Integration timestep Δt	$10^{-4}\tau$
Friction coefficient μ	0.5
Wall friction coefficient μ_w	0.5

Parameter	Value	Fitting method
Spot diffusion length, b	$2.28d$	The width of the DEM flow profile
Spot/particle displacement ratio, w	399	A single particle diffusion measurement in the top center of the container
Spot radius, r_s	$2.60d$	Fitting a length scale to the DEM spatial velocity correlation function
Spot insertion rate, λ	$375\tau^{-1}$	Overall DEM flow rate
Spot move rate, μ	$28.0\tau^{-1}$	Density drop during flow

Table 1: (a) The parameters used in the DEM simulation. (b) The five parameters used in the spot model simulations that were fitted from DEM simulation. All values are taken from Ref. [22].

those not displaced. A displaced particle could potentially touch a particle up to one diameter away, so making use of the above value of r_e will account for these interactions. Since the `spot` and `relax` operations are both local, they can be carried out very efficiently, by only testing the regions of the container which they overlap. The operations take care of cases when particles move from one region into another.

To carry out the spot simulation of the granular drainage process, an initial packing of particles is copied from the DEM simulation. The drainage process is then implemented as an event-driven simulation, whereby individual spots are introduced and moved according to exponential waiting time distributions with parameters λ and μ respectively. Spots initially start at the orifice at $\vec{s} = \vec{0}$. When a spot at position \vec{s} moves, its displacement \vec{v} is randomly chosen from one of the four vectors in $V = \{(\pm\Delta x, \pm\Delta y, \Delta z)\}$. If a spot's displacement would cause it to come within a buffer distance d_w of a wall, then \vec{v} is truncated so that its position after the displacement would lie exactly d_w from the wall. The spot's position is updated to $\vec{s} + \vec{v}$, and the microscopic mechanism is applied at the midpoint of this step, by calling `spot`($\vec{s} + \vec{v}/2, -\vec{v}/w, r_s$) and then `relax`($\vec{s} + \vec{v}/2, r_e$). When a spot reaches a height z_{\max} above the top of the packing, it is removed from consideration.

This simulation approach is detailed in the following algorithm, where the current position vectors of spots is held in a list S that is initially set to empty:

```

 $t = 0, S = \{\}$ 
while  $t < t_{\text{final}}$  do
   $t \rightarrow t - \log(\text{rand}()) / (\lambda + |S|\mu)$ 
  if  $\text{rand}() < \lambda / (\lambda + |S|\mu)$  then
    choose  $\vec{s}$  from  $S$ 
    if  $s_z < z_{\text{max}}$  then
      choose  $\vec{v}$  from  $V$ 
      truncate  $\vec{v}$  if within  $d_p$  of a wall
       $\text{spot}(\vec{s} + \vec{v}/2, -\vec{v}/w, r_s)$ 
       $\text{relax}(\vec{s} + \vec{v}/2, r_e)$ 
       $\vec{s} \rightarrow \vec{s} + \vec{v}$ 
    else
      delete  $\vec{s}$  from  $S$ 
    end if
  else
    add  $\vec{0}$  to  $S$ 
  end if
end while

```

In the same manner as the DEM simulation, snapshots of the particles are saved at intervals of 2τ . The parameters $\{\Delta x, \Delta y, \Delta z, w, r_s, \lambda, \mu\}$ control the speed and characteristics of the flow, and in general, appropriate values can be determined from DEM simulation or experimental data, that can be carried out once and then used in many simulations. Ref. [22] describes a systematic process for fitting these parameters from the DEM simulation based upon physical considerations. In the current paper, we do not concentrate on the fitting process, and make use of the same parameters from the previous study that are summarized in Table 1(b). The five parameters listed here control five independent degrees of freedom governing the main physical characteristics of the bulk flow. In addition, we make use of $d_w = 1d$ from Ref. [22], although this only affects particle motion near the walls. The vertical step size Δz determines how accurately the diffusing spots are modeled, but does not strongly affect the results. Here, we choose $\Delta z = 0.1d$; smaller steps have been investigated, but increase the total simulation time with little change in the physical results. Once the vertical step size is cho-

sen, the horizontal displacements must be set to $\Delta x = \Delta y = 0.676d$ so that the spot diffusion rate b satisfies the relationship $2b \Delta z = \Delta x^2 = \Delta y^2$.

Since the spot simulation is based on geometry alone and does not have to accurately model individual particle contacts, it runs much faster than DEM: draining the entire packing takes approximately eight hours to run as a serial code on a Mac Pro system with a 2.67 GHz dual-core Intel Xeon processor. The elastic relaxation process is the most time-intensive part of the simulation, with a typical `relax` call taking $115 \mu s$, which is about sixteen times slower than a `spot` call, that takes $7.15 \mu s$ to run.

Figure 2 shows snapshots of the DEM and spot simulations at $t = 60\tau$. Despite the two simulations being very different in running times, the overall agreement is very high, with the colored bands of particles deforming similarly. It should be borne in mind that the total flow rate and overall width of the spot simulation were fitted from DEM, but the shape of the velocity profile, and the individual particle mixing and rearrangement are reproduced well. Although not discussed here, a quantitative study of these two simulations showed good matches for velocity profiles, particle diffusion, spatial velocity correlation functions, radial distribution functions, and bond angle distribution functions [22].

Since spots can be thought of as carrying negative volume, it is interesting to examine the local changes in packing fraction that occur during the simulation. Figures 3(a) and 3(b) show plots of the local packing fraction in the DEM and spot simulations respectively, computed using the Voronoi cell software library *Voro++* [31]. Initially the packing fraction is approximately 64% but during flow this decreases to below 60%. The snapshots at $t = 20\tau$ show that the front of lower densities propagates at a similar speed in the two simulations. However, in DEM the areas of largest density drop are located at the sides, in regions of highest shear, whereas in the spot model, the areas of highest density drop are located the center, in regions of highest spot density and velocity. However, it is interesting to observe that at later times when the flow is developed, the regions of lower packing fraction are similar between spot and DEM simulations.

2.1. Fixed timestepping

Since the spot model microscopic mechanism is local, it offers the possibility of being coded efficiently in parallel, by carrying out many spot motions in different parts of the container simultaneously. Several possible algorithms have been implemented [20] that display modest parallel efficiency. However,

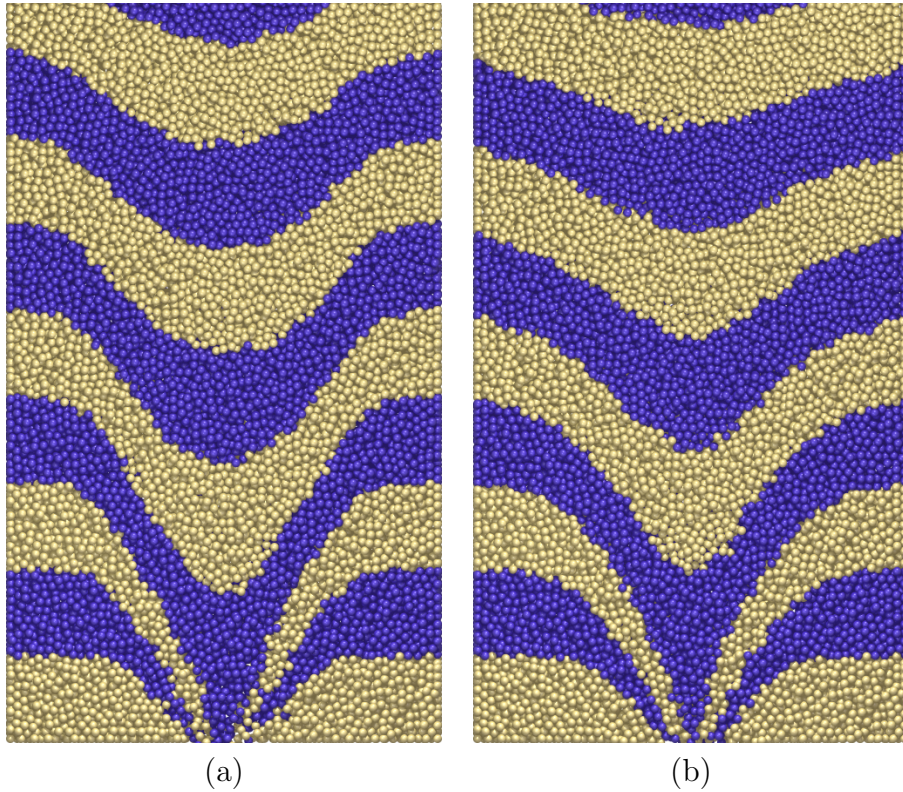


Figure 2: Snapshots of the (a) DEM simulation and (b) event-driven spot simulations at $t = 60\tau$. The central slice of the packing is shown, by only plotting those particles with $y > 0$. The two colors of particles are physically identical and initially form layers of thickness $10d$, and are used to highlight the flow and particle mixing that takes place.

the main hurdle to these algorithms is to correctly implement the event-driven nature of the spot motion: there are cases when a single spot may move several times in quick succession, or where several spots may overlap with each other, in which case the motions must be applied serially and the parallel efficiency is diminished. Similarly, if the simulation is divided into separate regions, each of which is controlled by a different processor that handles spots and particles in that area, there are difficulties with correctly synchronizing the time.

Because of these difficulties, we therefore investigated whether the event driven spot motion could be replaced with a fixed timestep, where all spots move after an interval Δt . To match the event-driven simulation, this timestep was chosen to be the mean time between spot move events, $1/\lambda$, and the number of spots introduced at the orifice at each timestep followed a Poisson distribution with parameter μ/λ . In this approach there is some freedom in what order the spot motions are applied, and we considered two different methods: (a) applied in a random ordering at each timestep, and (b) ordered according to the time they were inserted, applying the newest spots first. The ordering of the spots could potentially have a small effect on the particle motion. Method (a) is similar to the event-driven procedure where the spot motions occur randomly. Method (b) could potentially result in slightly different behavior, since by applying the newer, lower spots first, there is slightly more free space available for the particles above to move into. Simulations with both of these methods yield almost identical results to the event-driven approach, and snapshots of the flow are indistinguishable. Figure 3(c) shows Voronoi plots of the packing fraction in a fixed timestep simulation using method (b). The front of lower density at $t = 20\tau$ is almost identical to the event-driven case, and the behavior at later times is very similar.

We therefore think it is reasonable to conclude that both event-driven and fixed timestepping approaches can be used in spot simulations. With a fixed timestepping approach, it becomes feasible to write a fully distributed parallel spot algorithm, where all spot and motions are taken care of locally, and a master node is not required to keep track of the clock. We leave this as a subject for future work, but expect that very high parallel efficiency could be achieved. Our results also suggest that the precise ordering of spot motions does not play a significant role. For a multicore spot algorithm, this suggests that there may be some leeway in reordering spot motions while still achieving similar results, allowing for further boosts in speed.

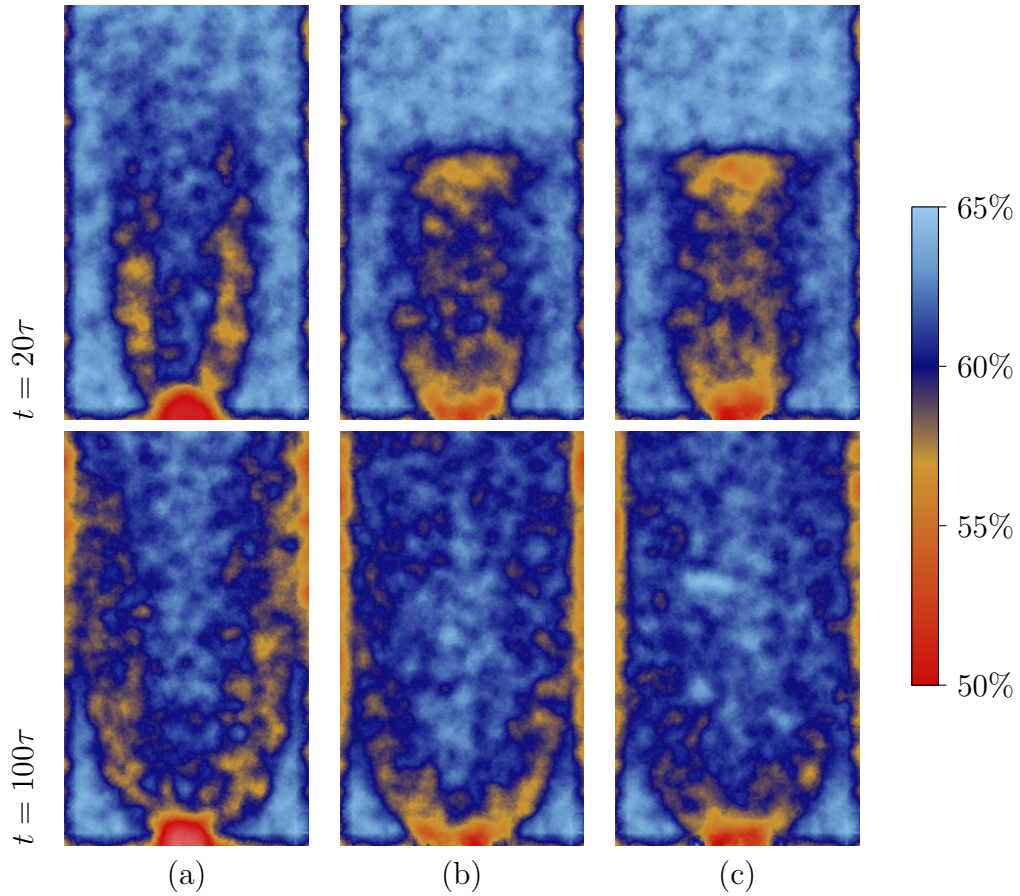


Figure 3: Plots of local packing fraction at two different times, for (a) the DEM simulation, (b) the event-driven spot simulation, and (c) the fixed timestep spot simulation. At each point, the local packing fraction is computed by finding all particles within a distance of $2.2d$ of that point, and then dividing the total particle volume by the total volume of their Voronoi cells. (For packing fractions in the range 0%–50% near the orifice, the color is smoothly graded from white to red.)

3. Modeling the free surface

The simulations presented previously concentrated only on the particle flow in the bulk, and as such, the free surface of the packing was omitted from investigation and not shown. However, many situations arise where modeling the free surface would be essential, and in this section, we show that it is possible to modify the spot random walk process using methods that preserve all of the flow properties in the bulk, but also give a realistic description of the top of the packing. The free surfaces of granular materials have been extensively studied [32, 33, 34], and it is well-known that the inclination of the surface of a granular pile will not exceed a critical angle, referred to as the angle of repose. In granular drainage, an initially flat surface will become progressively more inclined towards the angle of repose as drainage occurs. As shown in the DEM snapshot in figure 4(a), the yellow and cyan particles near the top surface avalanche towards the center during flow. The unbiased random walk process described in the previous section will not capture this behavior, as the free surface will follow the mean velocity streamlines in the bulk.

In the void model, the evolution of the free surface has been addressed by making use of a very simple modification of the random walk process [11]. In the bulk of the packing, when a void generally has two particles in the lattice points above it, the void moves to each of these sites with equal probability. However, in the case when only one of these two sites is filled with a particle, the void always moves in the direction of the particle. A void is only removed from the simulation when both of the sites above it are vacant. This simple modification suffices to create heaps and avalanching at the free surface, as when a void reaches a heap, it travels diagonally upwards along the heap surface. The angle of repose in this model is tied to the spacing of the underlying lattice.

This, in effect, creates a simple biasing of the random walk: there are two locations it can move to, and it chooses randomly among the available options. This concept can be adapted to the spot model. Suppose a spot can move to N different locations, and it would influence p_i particles if it moved to location i . Let $q = \sum_{i=1}^N p_i$. If $q = 0$, remove the spot from the simulation. Otherwise, let

$$\mathbb{P}(\text{Spot moves to } i) = \frac{p_i}{q}. \quad (4)$$

In the bulk, where the density of the packing is almost constant, this does

not alter the random walk process by a large amount, but at the surface, the motion of the spots is biased to create heaps. A snapshot of a spot drainage simulation using this procedure is shown in figure 4(b), that qualitatively captures the particle avalanching seen in DEM. However, the free surface angle is too large, and furthermore it can be seen that particles near the top surface have become separated. This can be seen more clearly in figures 5(a) & 5(b), which show close-up images of the the central slice of the packing. Here, each particle is colored according to the packing fraction computed from its Voronoi cell, revealing a large drop in density in the spot model. This occurs because there is no explicit gravity in the spot model, so particles which become separated during elastic relaxation events will remain separated.

A further modification to the spot model can be employed to correct for this. In the previous implementation, when a spot moves by \vec{r}_s , then the particles experience a displacement $-w\vec{r}_p$, where w is a fixed quantity. Suppose that a spot is going to influence p particles, each of volume V_p . If spots are thought of as carrying a completely fixed amount of free volume V_s , then another possible approach would be to let $w = V_s/pV_p$, so the spot's influence is divided equally among the particles in range. In the bulk, where the particles are roughly at constant density, this modification has little effect. However, at the free surface, where p is lower, the spots give a larger downwards push, stopping the particles from separating.

The preceding argument about spots carrying a completely fixed amount of volume is only directly applicable in the bulk of the packing, and must be modified slightly at the orifice and at the free surface. For a spot in the neighborhood of the orifice, at a height $z_s < r_s$, we compute the modified influence $V'_s(z_s)$ by multiplying V_s by the proportion of the spot that is above $z > 0$. A volume integration gives

$$V'_s(z_s) = \left(\frac{1}{2} + \frac{3z_s}{4r_s} - \frac{z_s^3}{4r_s^3} \right) V_s. \quad (5)$$

At the free surface, as spots exit the packing, the number of particles within range of a given spot may be very small, and hence those particles may experience a very large displacement due to the $1/p$ dependence in the influence. An additional constraint was therefore implemented: if $p < 20$, then the spot displacement was calculated based on $p = 20$. Physically, this modification could be thought of as spots partially evaporating when they get close to the top of the packing, so that their influence is weakened. A snapshot of a

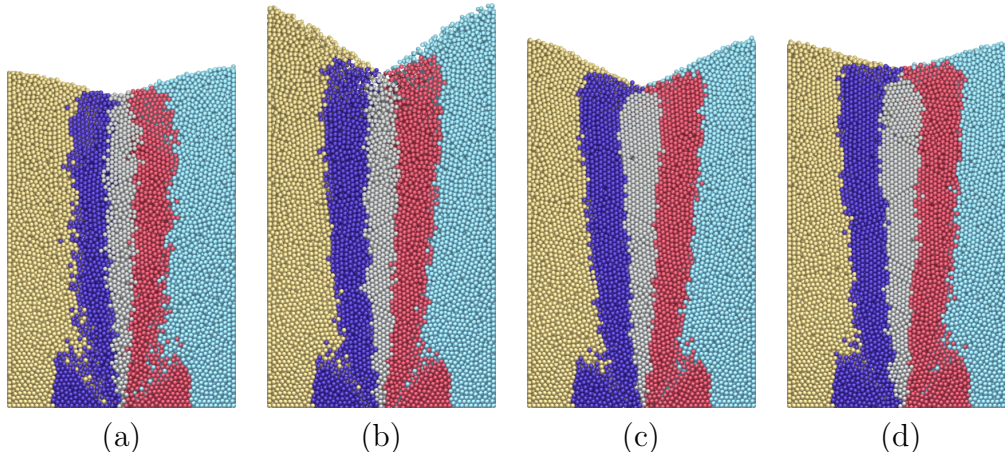


Figure 4: Snapshots of (a) DEM simulation, (b) a spot simulation using a random walk with simple biasing, (c) a spot simulation using simple biasing and influence weighting, and (d) a spot simulation using adapted biasing (with $\beta = 3$) and influence weighting. The snapshots are taken at $t = 300\tau$.

simulation using this prescription is shown in figure 4(c), and appears very promising. The free surface correctly forms heaps, and the Voronoi computation of packing fraction in figure 5(c) shows that particles no longer become separated at the free surface.

The specific implementation of the random walk biasing procedure given in Eq. 4 is arbitrary, and could be replaced by a different scheme. One possibility is the nonlinear power-law weighting,

$$\mathbb{P}(\text{Spot moves to } i) = \frac{p_i^\beta}{\sum_{i=1}^N p_i^\beta}, \quad (6)$$

where β is a parameter that can be used to control the amount of biasing. For $\beta > 1$ the bias is more strongly weighted toward high particle densities (above the mean), while for $\beta < 1$ the bias is reduced compared to linear weighting. Figure 4(d) shows a snapshot of a simulation using $\beta = 3$, in addition to the spot influence weighting, and figure 5(d) shows a close-up of the free surface, with very good agreement to DEM. Values of up to $\beta = 10$ were tested, and result in progressively shallower free surface slopes.

To track the time evolution of the free surface in these simulations, a simple regression procedure was used to extract the surface angle. We define

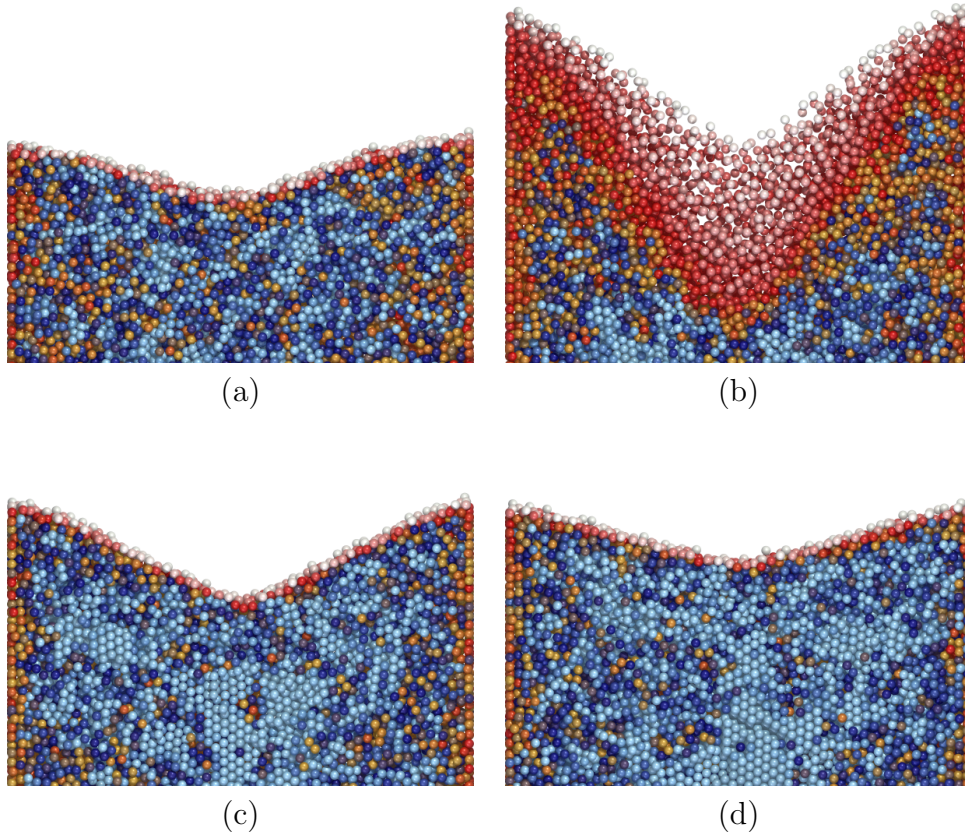


Figure 5: Close-up snapshots of the free surface of the particle packings for the same simulations in figure 4, for the region $50d < z < 90d$. Each particle is colored according to its local packing fraction, computed as the ratio of its volume to its Voronoi cell volume. The central slice of the packing is shown by only plotting particles with $y > 0$. The same color scheme is used as for figure 3.

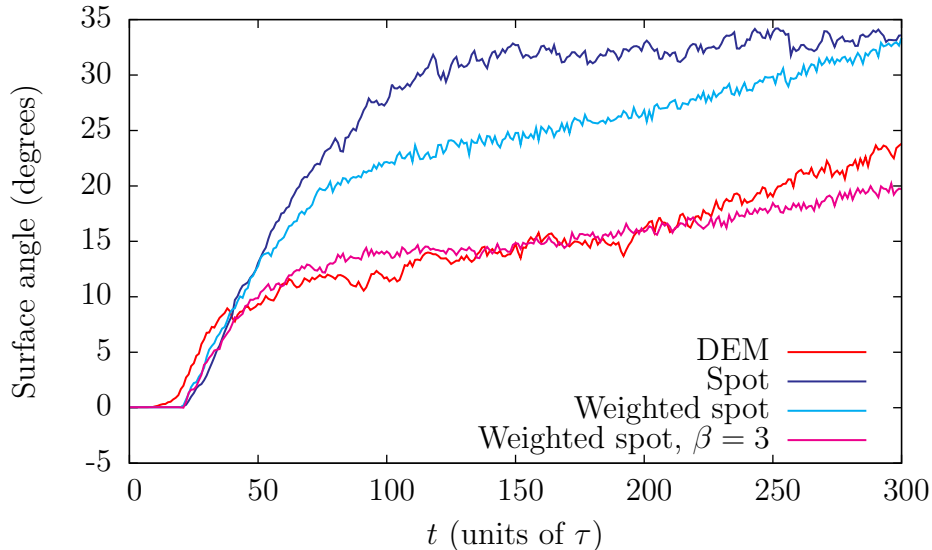


Figure 6: Time evolution of the angle of the free surface, calculated using linear regression, of the four simulations shown in figure 4.

$x_j = (2.5j - 1.25)d$ for $j = 1, 2, \dots, 10$. For a given particle snapshot, the values z_j are computed as the maximum particle z coordinate in the range $||x| - x_j| < 1.25d$. The angle of the slope can then be computed by applying linear regression on set of (x_j, z_j) points. Figure 6 shows the a plot of this angle for the four simulations considered. The weighted spot simulation with $\beta = 3$ is in good agreement with the DEM simulation, and can track a gradual increase in the slope angle during flow.

4. Boundary layers

In transport phenomena, a boundary layer is a region near a surface where the continuum variables vary rapidly in the normal direction on a length scale much smaller than the global flow domain. The proper mathematical treatment of boundary layers is based on the method of matched asymptotic expansions, when the governing equations contain a small singular perturbation parameter which can often be expressed as a typical boundary layer thickness divided by a geometrical length. In fluid mechanics, boundary layers generally arise due to the no-slip boundary condition, causing viscous stresses to be important close to the surface, while often less important in the bulk (at high Reynolds number). In granular materials, the continuum

description is less clear, and combines liquid-like and solid-like behavior, but the appearance of boundary layers is possible whenever wall interactions retard the flow (*e.g.* by roughness or friction), compared to the bulk flow, if the bulk material has relatively short-range correlations.

In the test geometry considered here, the best place to examine this behavior is in the y direction, across the $8d$ width of the packing. Figure 7(a) shows the velocity profile in this direction in the central region of flow, for DEM simulations with five different values of wall friction, μ_w . For $\mu_w = 0.1$ the flow is almost constant across the width of the packing, but as μ_w is increased, regions of slower flow next to the wall become apparent.

In the spot simulations, the downwards velocity profile is given approximately by the convolution of the spherical spot influence with the width over which the spots move. (Additional motion from particle relaxation could also play a role, although it can be expected that this is a less important effect.) Near the container walls, the velocities will therefore begin to smoothly decrease, and the size of this slower-moving region will be determined by the wall buffer distance d_w , that controls how close the spot centers can come to the wall. Figure 7(b) confirms this by showing the same cross sections for spot simulations with the default value of $d_w = d$, and three other values of d_w . For $d_w = 0.5d$ the width of the spot simulation boundary layer is in reasonable agreement with those in DEM for high wall friction values, although the very rapid drop in velocities close to the wall is not well captured. In order to reproduce the low wall friction case where no boundary layer is present, the best match is achieved with $d_w = -1.5d$, so that the centers of spots are allowed to move outside of the container walls.

Allowing the spot centers to move outside the walls is valid, since these spots still create valid collective motions of the particles near the wall. However, it introduces an additional complication: in determining the original spot model parameters, the insertion rate λ is set by balancing the DEM particle outflow rate with amount of displacement an individual spot causes. If a fraction of the total number of spots are allowed to significantly drift out of the container then they influence fewer particles and the total particle outflow rate is significantly lowered. For the simulation with $d_w = -1.5d$, the total particle outflow rate is $106.8\tau^{-1}$ over the time interval used in Fig. 7, whereas the outflow rate for the DEM simulation used in the original parameter fitting was $131.1\tau^{-1}$.

The spot influence weighting procedure discussed in the previous section, where the particle displacements caused by a spot are scaled by the number

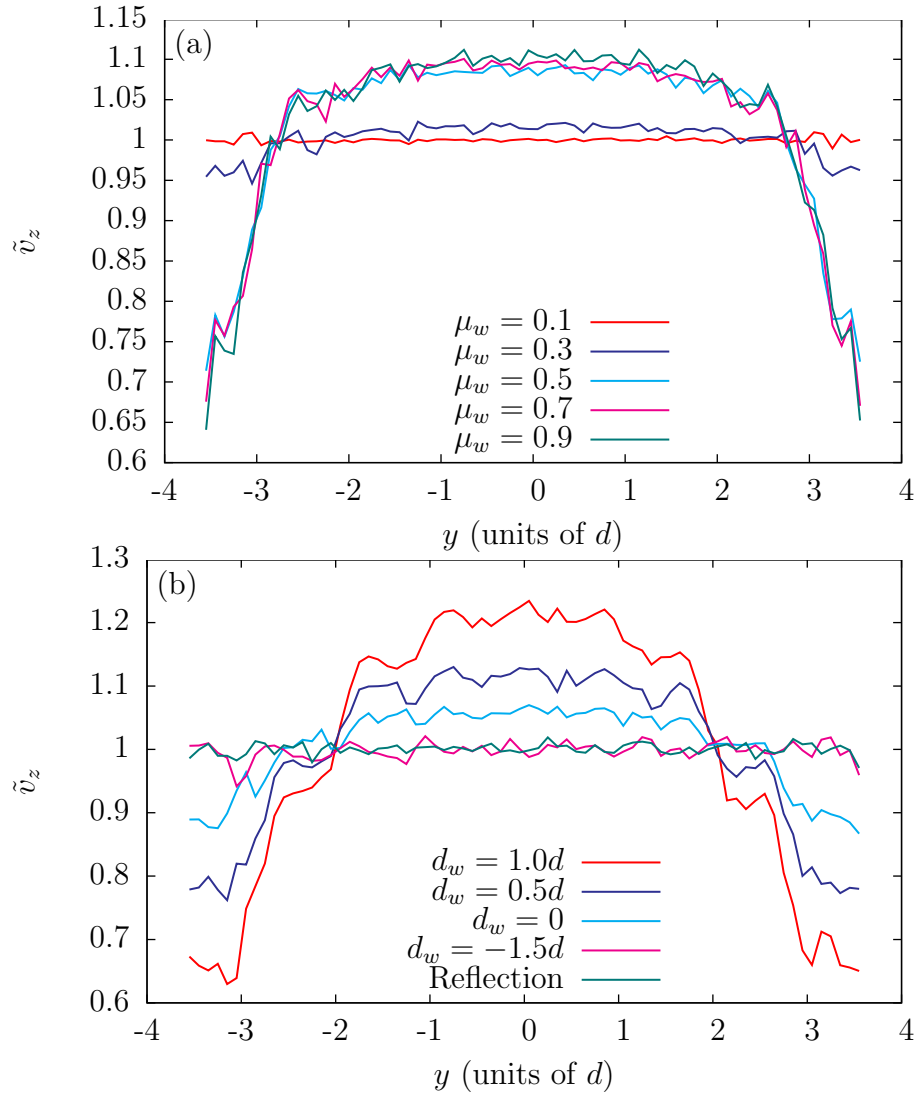


Figure 7: Normalized vertical velocity profiles across the width of the container, in the test region $x < 10d, 50d < z < 70d$, over the time window $80\tau < t < 180\tau$, for (a) five different DEM simulations with different values of wall friction μ_w , and (b) five different spot simulations with different values of d_p , plus the reflection procedure. For each simulation, the velocities are normalized by the average velocity \bar{v}_z in the test region, so that $\tilde{v}_z(y) = v_z(y)/\bar{v}_z$.

of particles influenced, provides one method of circumventing this issue, as each spot will always have equal influence. Another possible approach is to set $d_w = 0$, and then reflect the part of the spot's influence that lies outside the container back into the packing. More specifically, if a spot is at $\vec{s} = (s_x, s_y, s_z)$ with $s_y > 0$, and there is a wall at $y = 0$, then the displacement on a particle at \vec{x} is scaled according to

$$S(\vec{x}) = \begin{cases} 0 & \text{if } |\vec{x} - \vec{s}| \geq r_s \\ 1 & \text{if } |\vec{x} - \vec{s}| < r_s \text{ and } |\vec{x} - (s_x, -s_y, s_z)| \geq r_s \\ 2 & \text{if } |\vec{x} - \vec{s}| < r_s \text{ and } |\vec{x} - (s_x, -s_y, s_z)| < r_s. \end{cases} \quad (7)$$

This procedure means that the total spot influence is uniform across the entire width of the packing. For spots near edges between two walls, the reflection procedure is applied for both walls. As shown in Fig. 7(b), the vertical velocity profile is roughly uniform using this procedure, without the large drop in particle flux seen in the $d_w = -1.5d$ simulation.

With the ability to approximate boundary layers of slower flow, it is possible to gain good estimates of particle residence-time distributions during a drainage process. In previous work, it has been shown that a continuum analysis of the kinematic model can be useful in predicting the tails of a the residence time distribution [7], even without any careful treatment of the flow near the boundaries. In the spot simulation, where the bulk flow is similar to the kinematic model, and the boundary layers of slower flow can be approximated, it will be possible to predict residence-time distributions to a higher degree of accuracy.

5. Infrequent relaxation

One of the most surprising aspects of the spot simulation is that addition of the relaxation step, shown in figure 1(b), is enough to completely enforce packing constraints. While the displacements introduced by the relaxation step are based upon geometry alone with no details of the contact physics, and typically are the order of 20% of the spot displacement, the radial distribution function $g(r)$ was exactly zero over the range $0 < r < 1d$ for the entirety of the simulation, corresponding to no overlapped particles. Furthermore, over a medium time interval, the spot model simulation was accurate enough to track minuscule changes in $g(r)$ and the bond angle distribution function $g_3(\theta)$ that were seen in DEM. This success can be largely attributed to the fact that the particle displacement induced by a single spot motion

is extremely small, on the order of a hundredth of a particle diameter, so after each motion, only an extremely small motion is required to fix packing constraints.

As discussed in section 2, even when efficiently implemented, the elastic relaxation step is the most time-consuming part of the simulation, requiring a consideration of all neighboring pairs of particles, and can take approximately sixteen times as long as a spot motion. Here, we ask whether it is possible to apply relaxation less often and still recreate valid particle packings, as a potential method of speeding up the code even further. While this will result in a loss of accuracy, it may be appropriate for some situations, where we do not require perfect random packings, but would still like the particles not to suffer from large local buildups in density.

A series of simulations was carried out whereby the amount of relaxation applied was reduced by a factor of k . For a given reduction factor, three different methods were implemented:

1. A local relaxation is applied after each step with probability $1/k$, for $k = 10, 100, 1000$.
2. Each spot keeps an individual counter of the number of times it has moved, and a local relaxation is applied after every k th motion, for $k = 10$.
3. After a time $\Delta t = k/\mu$, when spots have each moved k times on average, a global relaxation is applied, for $k = 1, 10, 100$.

It is not clear *a priori* whether methods 1 & 2 with local relaxations, or method 3 with global relaxations, would be more efficient in maintaining the quality of the packing. By applying local relaxations infrequently, the possibility may arise that some particles will not be part of any relaxation event for a long period of time, but with periodic global relaxations, one is assured that all particles will be considered equally often. However, the local relaxation procedures have the advantage that the amount of relaxation in a particular region is proportional to the number of spot motions in that region. A summary of the simulations considered and their running times is given in Tab. 2.

Figure 8 shows a selection of simulation snapshots during the flow, at $t = 120\tau$. For the original simulation with full relaxation (a), the snapshot looks very similar to DEM, with no visible evidence of overlapped particles. This is in contrast to the simulation, with no relaxation shown in (d), where the absence of any relaxation causes significant particle overlaps as the flow

Relaxation method	Relaxation ratio k	t_{relax} (s)	B
Full relaxation	1	22446.4	0.00110
No relaxation	–	0	0.501
Method 1	10	2359.23	0.00798
Method 1	100	244.4	0.0286
Method 1	1000	23.8	0.0871
Method 2	10	1629.81	0.00792
Method 3	1	261.1	0.0183
Method 3	10	28.0	0.0631
Method 3	100	2.8	0.171
DEM	–	–	0.00365

Table 2: Summary of the simulation runs considered making use of infrequent inelastic relaxations. t_{relax} refers to the total time spent in the relaxation routines. Also given is an overall packing badness B , computed over $40\tau < t < 160\tau$.

takes place. In figures (b) and (c), the same snapshots are shown using method 1 relaxation for $k = 100$ and $k = 1000$ respectively. Despite the relaxation steps being applied much less frequently, the particle packings appear to be in very good agreement with the original case, and very little evidence of significantly overlapped particles can be seen.

To investigate this quantitatively, the radial distribution function $g(r)$ was computed in the central region of flow, $C = \{-15d < x < 15d, 15d < z < 45d\}$ over all snapshots in the time interval $40\tau < t < 200\tau$. This is done by first computing the frequency distribution $N(r)$ of neighbor separations $r = |\vec{x} - \vec{y}|$ for all particle pairs (\vec{x}, \vec{y}) subject to the restriction that $\vec{x} \in C$. After this, $\bar{N}(r)$ is calculated as the theoretical frequency distribution for a homogeneous arrangement of particles at the same average density ρ as the test packing. In three dimensions this would simply be $\bar{N}(r) = 4\pi r^2 \rho$ by integrating over a spherical shell. However, here the packings are confined over an $8d$ range in the y direction, so that some of the spherical shells are not complete. By carrying out a surface integration, it can be shown that

$$\bar{N}(r) = \begin{cases} 4\pi\rho r^2(1 - \frac{r}{16d}) & r \leq 8d \\ 16\pi\rho dr & 8d < r \leq 10d. \end{cases} \quad (8)$$

We do not consider this for $r > 10d$, since the spherical shells centered within the test region C would then be affected by the side walls also, and the integration would become significantly more complicated. The radial distribution

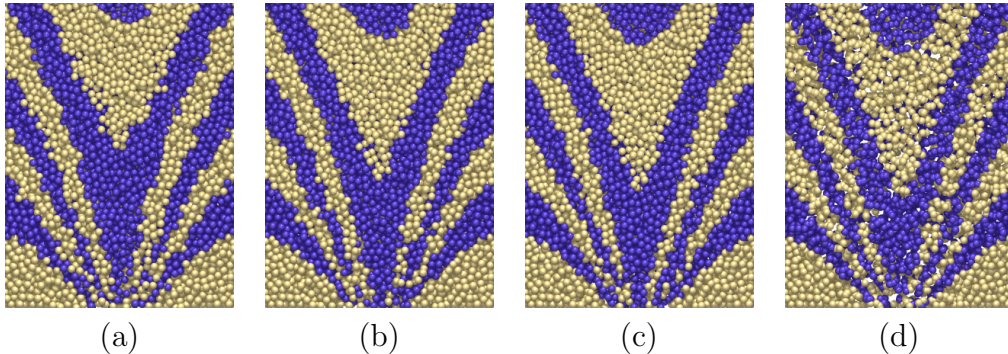


Figure 8: Snapshots at $t = 120\tau$ for (a) the original simulation, (b) the method 1 simulation with $k = 100$, (c) the method 1 simulation with $k = 1000$, and (d) the simulation with no relaxation. The region $-15d < x < 15d, 0 < z < 40d$ is shown, and only particles with $y > 0$ are plotted, to view the central slice of the particle packings.

function is then calculated as $g(r) = N(r)/\bar{N}(r)$, so that it represents the deviations in the neighbor separations from a completely homogeneous packing. Figure 9 shows the computed curves for all of the spot simulations listed in Tab. 2. For the simulation with full relaxation, there are no significantly overlapped particles: $g(r)$ is identically zero over the range $0 < r < 0.9925d$. The curve also has significant peaks at $r = \sqrt{3}d, 2d$, due to local particle ordering, a behavior which closely matches the corresponding DEM simulation. The radial distribution functions for the simulations with infrequent relaxation are difficult to distinguish from the full relaxation case, and the differences only become clear when looking at a zoomed-in region, as shown in Fig. 10(a). In this plot, we can see that the the runs with less relaxation give progressively smoother curves. Figure 10(b) shows a semi-logarithmic plot of $g(r)$ for small separations to highlight the amount of overlapped particles. For the infrequently relaxed simulations, there are some overlaps, although even with very little relaxation the tails in $g(r)$ do not significantly extend beyond $0.8d$ to $0.9d$. All the curves with some relaxation are significantly more realistic than the case with no relaxation, which has a large number of separations less than $1d$, and has an almost uniform distribution, as would be expected for randomly placed points in a domain.

To directly compare the results of each simulation, an overall packing badness B is computed for each, based on the amount that particles are overlapped. If a given particle i has n_i overlapping contacts, and the overlap amounts are specified by δ_{ij} for $j = 1, \dots, n_i$, the packing badness is

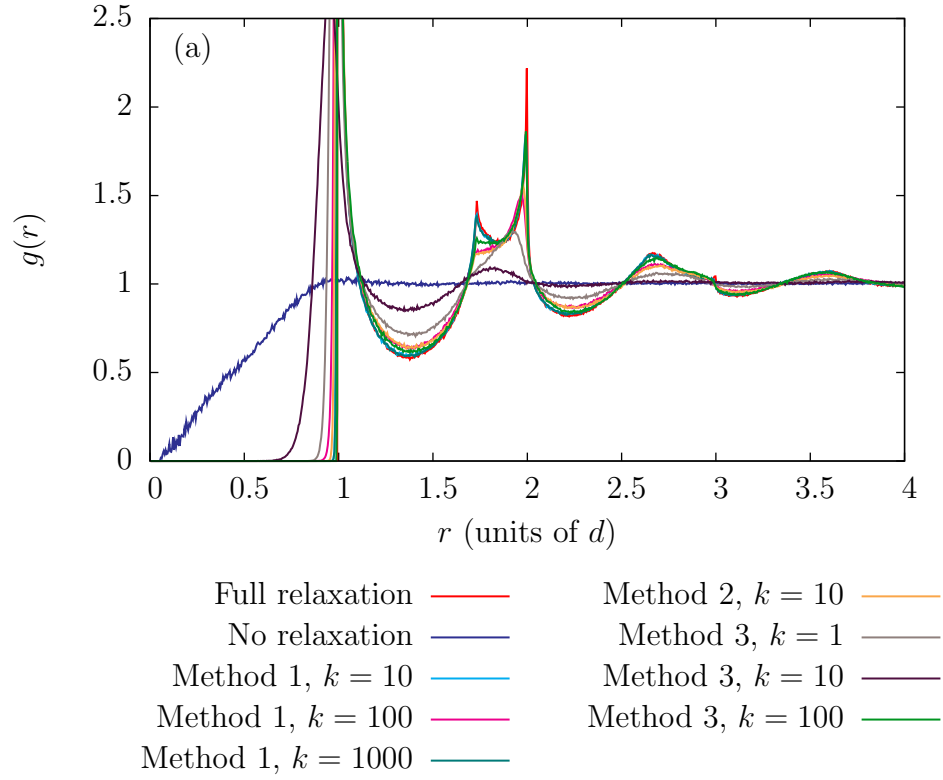


Figure 9: The radial distribution function $g(r)$ for the spot simulations listed in Tab. 2, computed in a central region of flow, $-15d < x < 15d, 15d < z < 45d$ for all snapshots over the range $40\tau \leq t \leq 200\tau$.

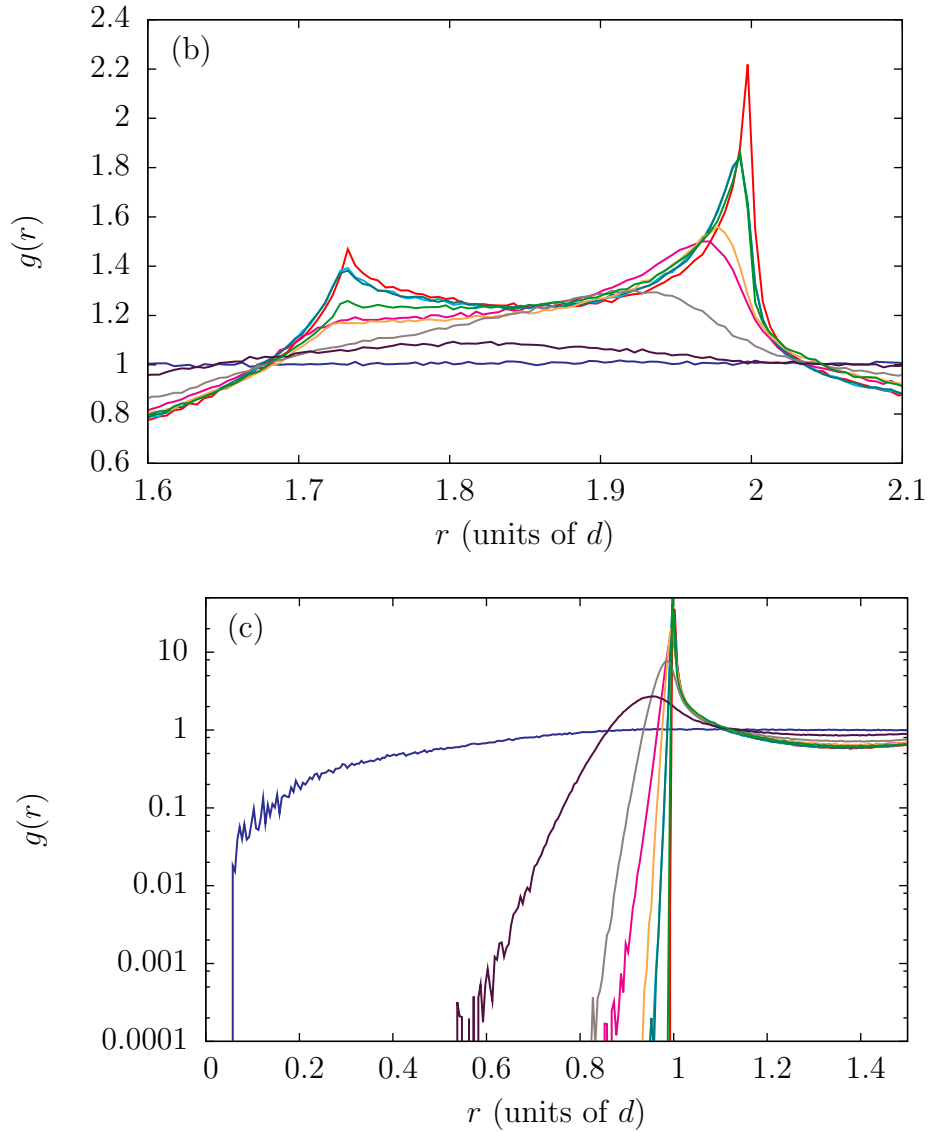


Figure 10: (a) A zoomed-in region of the radial distribution functions of Fig. 9 showing the second and third peaks. (b) A semi-logarithmic plot of the radial distribution functions in the region $0 < r < 1.5d$, highlighting the parts of the curves that correspond to overlapped particles. The colors of the curves are the same as those used in Fig. 9

computed as

$$B = \frac{1}{p} \sum_{i=1}^p \sum_{j=1}^{n_i} \frac{\delta_{ij}^2}{d^2}. \quad (9)$$

Here we consider all particles in the same region as the $g(r)$ distributions, using the same time interval. The results are shown in Tab. 2 and figure 11. For the three different infrequent relaxations considered, there is a roughly one-to-one relationship between the total simulation time spent on relaxation and the packing badness, suggesting that the precise details of how the relaxation is applied have little overall effect on the packing structure.

These results suggest that infrequent relaxation is a promising method of speeding up a spot simulation. As discussed in section 2, a typical local relaxation step takes about sixteen times as long as a spot motion. Thus a simulation will run an order of magnitude faster if relaxation is applied only a tenth of the time, and at this level, the computed packing badnesses are very small, and roughly comparable to DEM, where particles necessarily overlap to create forces. Reducing the relaxation by a factor of a hundred or a thousand still results in reasonable packings, although there is less practical justification for using these, since at that level the total simulation time becomes dominated by the spot motions.

6. Conclusion

In this paper, we have demonstrated a simple multiscale simulation technique that is capable of modeling many features of granular drainage at a fraction of the computational cost of DEM. The spot model microscopic mechanism provides a reasonable description of particle flow and rearrangement in the bulk, and in addition, we have shown that simple modifications to the simulations can be used to model free surfaces and boundary layers. We believe that the spot model may have applications in a large number of practical problems where features of granular drainage must be estimated in real time. We envisage that the free parameters in the model could be initially estimated, either from DEM simulation or from experimental data, and then used as a basis for many spot simulations.

We have also demonstrated that the basic concept of breaking down a flow into mesoscopic group displacements is robust, and that the physical results do not depend strongly on the precise implementation. Both event-driven and fixed timestep simulations yield largely similar results. One of the most

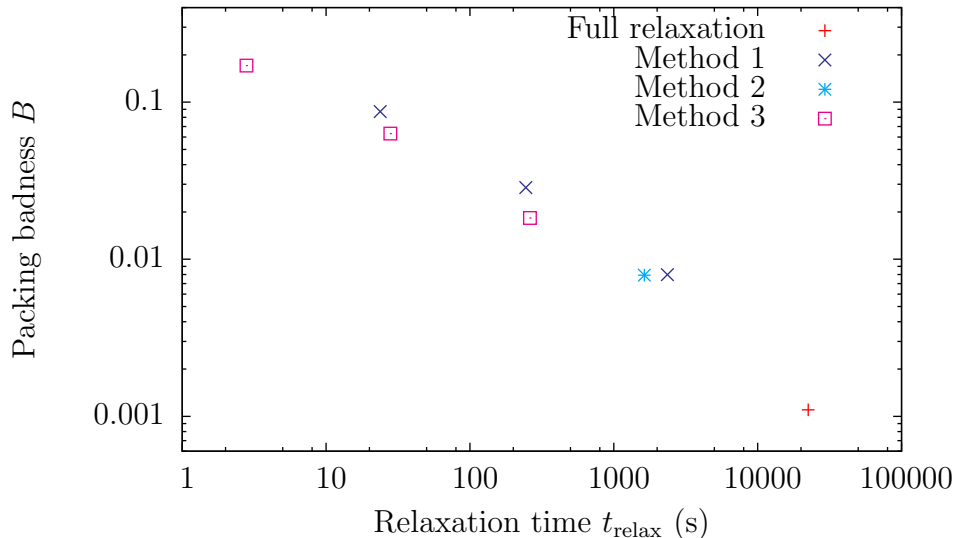


Figure 11: Logarithmic plot of the packing badness B versus the total time spent on relaxation in the simulation t_{relax} , for the simulation with full relaxation, and the simulations with the three different types of infrequent relaxation.

surprising conclusions of the simulations by Rycroft et al. [22] was that the simple elastic relaxation step was good enough to preserve random packings, with no significantly overlapped particles, and here we have shown that even if relaxation is applied very infrequently, the random packings can still be reasonably accurate. These results bode well for designing future spot-based algorithms, as they point to a great deal of flexibility in the implementation. Since the spot simulations are much quicker than DEM, and handle particle interactions purely based on geometry as opposed to a detailed consideration of contact dynamics, they offer the possibility of simulating problems that would be otherwise infeasible with DEM. It would be possible to simulate several orders of magnitude more particles than can currently be considered with DEM. Since particle dynamics in the spot model is based on geometric considerations, it is straightforward to generalize to polydisperse systems or irregular particles. The particle relaxation displacements given in Eq. 3 can be generalized to polydisperse particles by weighting according to the relative particle masses. Using this, it may be possible to understand the role of geometric packing constraints on effects such as segregation. For elongated or irregular particles, a more complex relaxation step would be needed, that also induces rotations on the particles, allowing for the study of effects such

as texturing under shear.

The largest limitation of the simulations presented here is that the technique only applies to granular drainage problems where the kinematic model, with diffusing vertical velocity profiles, is a good approximation. However, the basic concept of breaking down a flow in a mesoscopic group displacements appears to be generally applicable, and related work [23, 25] suggests that it may be possible to link this mechanism of particle motion with a mechanical theory of granular flow, to create a complete model. The microscopic particle mechanism may be a useful technique in studying other systems featuring dense amorphous arrangements of particles, where co-operative particle motion [35] is a frequently observed feature.

7. Nomenclature

b spot diffusion length (L)

d particle diameter (L)

μ particle–particle Coulomb friction coefficient

μ_w particle–wall Coulomb friction coefficient

τ simulation time unit (T)

g gravitational acceleration (L/T²)

r_s spot radius (L)

r_e relaxation radius (L)

δ_{ij} particle overlap (L)

λ spot insertion rate (T)

μ spot move rate (T)

d_w spot wall buffer distance (L)

$\Delta x, \Delta y, \Delta z$ spot random walk step sizes (L)

w spot displacement ratio

k relaxation factor

$g(r)$ radial distribution function

B packing badness

t_{relax} simulation time for relaxation (T)

8. Acknowledgments

This work was partially supported by the National Science Foundation under grants DMS-0410110 and DMS-070590; by the Director, Office of Science, Computational and Technology Research, U.S. Department of Energy under Contract Numbers DE-AC02-05CH11231 and DE-FG02-02ER25530; and the Norbert Weiner Research Fund and the NEC Fund at MIT. We are grateful to the Scientific Cluster Support (SCS) program at the Lawrence Berkeley National Laboratory.

9. Bibliography

- [1] P. A. Cundall, A computer model for simulating progressive large scale movements in blocky rock systems, in: Proc. Symp. Int. Soc. for Rock Mechanics, Nancy, France, 1971, pp. 129–136.
- [2] S. Reiss, Let a thousand reactors bloom, Wired Magazine 12.09.
- [3] D. Nicholls, The pebble-bed modular reactor, Nuclear News 44.
- [4] <http://web.mit.edu/pebble-bed/>.
- [5] H. D. Gougar, W. K. Terry, A. M. Ougouag, Matrix formulation of pebble circulation in the PEBBED code, in: Proceedings of the Conference on High-Temperature Reactors, Petten, NL, International Atomic Energy Agency, Vienna, Austria, 2002, pp. 1–6.
- [6] W. K. Terry, H. D. Gougar, A. M. Ougouag, Direct deterministic method for neutronics analysis and computation of asymptotic burnup distribution in a recirculating pebble-bed reactor, Annals of Nuclear Energy 29 (2002) 1345–1364.
- [7] C. H. Rycroft, G. S. Grest, J. W. Landry, M. Z. Bazant, Analysis of granular flow in a pebble-bed nuclear reactor, Phys. Rev. E 74 (2006) 021306.

- [8] J. Choi, A. Kudrolli, R. R. Rosales, M. Z. Bazant, Diffusion and mixing in gravity driven dense granular flows, *Phys. Rev. Lett.* 92 (2004) 174301.
- [9] J. Litwiniszyn, The model of a random walk of particles adapted to researches on problems of mechanics of loose media, *Bull. Acad. Pol. Sci.* 11 (1963) 593.
- [10] J. Mullins, Stochastic theory of particle flow under gravity, *J. Appl. Phys.* 43 (1972) 665.
- [11] H. Caram, D. C. Hong, Random-walk approach to granular flows, *Phys. Rev. Lett.* 67 (1991) 828–831.
- [12] R. M. Nedderman, U. Tüzün, A kinematic model for the flow of granular materials, *Powder Technology* 22 (1979) 243.
- [13] U. Tüzün, R. M. Nedderman, Experimental evidence supporting the kinematic modelling of the flow of granular media in the absence of air drag, *Powder Technology* 23 (1979) 257.
- [14] A. Medina, J. A. Cordova, E. Luna, C. Trevino, Velocity field measurements in granular gravity flow in a near 2D silo, *Physics Letters A* 220 (1998) 111–116.
- [15] A. Medina, J. Andrade, C. Trevino, Experimental study of the tracer in the granular flow of a 2D silo, *Physics Letters A* 249 (1998) 63–68.
- [16] J. Choi, A. Kudrolli, M. Z. Bazant, Velocity profile of gravity-driven dense granular flow, *J. Phys.: Condensed Matter* 17 (2005) S2533–S2548.
- [17] G. M. Gutt, P. K. Haff, An automata model of granular materials, in: *Proceedings of the Fifth Distributed Memory Computing Conference*, IEEE, 1990, pp. 522–529.
- [18] G. Peng, T. Ohta, Velocity and density profiles of granular flow in channels using a lattice gas automaton, *Phys. Rev. E* 55 (6) (1997) 6811–6820.
- [19] J. Kozicki, J. Tejchman, Application of a cellular automaton to simulations of granular flow in silos, *Granular Matter* 7 (2005) 45–54.

- [20] C. H. Rycroft, Multiscale modeling in granular flow, Ph.D. thesis, Massachusetts Institute of Technology (2007).
- [21] M. Z. Bazant, The spot model for random-packing dynamics, *Mechanics of Materials* 38 (2006) 717–731.
- [22] C. H. Rycroft, M. Z. Bazant, G. S. Grest, J. W. Landry, Dynamics of random packings in granular flow, *Phys. Rev. E* 73 (2006) 051306.
- [23] K. Kamrin, M. Z. Bazant, Stochastic flow rule for granular materials, *Phys. Rev. E* 75 (2007) 041301. [arXiv:cond-mat/0609448](https://arxiv.org/abs/cond-mat/0609448).
- [24] K. Kamrin, C. H. Rycroft, M. Z. Bazant, The stochastic flow rule: A multi-scale model for granular plasticity, *Modelling Simul. Mater. Sci. Eng.* 15 (2007) S449–S464.
- [25] C. H. Rycroft, K. Kamrin, M. Z. Bazant, Assessing continuum postulates in simulations of dense granular flow, *J. Mech. Phys. Solids* 57 (2009) 828–839.
- [26] <http://lammps.sandia.gov/>.
- [27] P. A. Cundall, O. D. L. Strack, A discrete numerical model for granular assemblies, *Geotechnique* 29 (1979) 47.
- [28] L. E. Silbert, D. Ertas, G. S. Grest, T. C. Halsey, D. Levine, S. J. Plimpton, Granular flow down an inclined plane: Bagnold scaling and rheology, *Phys. Rev. E* 64 (5) (2001) 051302. doi:10.1103/PhysRevE.64.051302.
- [29] J. W. Landry, G. S. Grest, L. E. Silbert, S. J. Plimpton, Confined granular packings: structure, stress, and forces, *Phys. Rev. E* 67 (2003) 041303.
- [30] C. H. Rycroft, A. V. Orpe, A. Kudrolli, Physical test of a particle simulation model in a sheared granular system, submitted to *Phys. Rev. E*.
- [31] C. H. Rycroft, Voro++: a three-dimensional Voronoi cell library in C++, Tech. Rep. LBNL-1430E, Lawrence Berkeley National Laboratory (2009).

- [32] T. Elperin, A. Vikhanski, Granular flow in a rotating cylindrical drum, *Europhys. Lett.* 42 (1998) 619.
- [33] A. V. Orpe, D. V. Khakhar, Scaling relations for granular flow in quasi-two-dimensional rotating cylinders, *Phys. Rev. E* 64 (2001) 031302.
- [34] D. Bonamy, F. Daviaud, L. Laurent, Multiscale clustering in granular surface flows, *Phys. Rev. Lett.* 89 (2002) 034301.
- [35] C. Donati, J. F. Douglas, W. Kob, S. J. Plimpton, P. H. Poole, S. C. Glotzer, Stringlike cooperative motion in a supercooled liquid, *Phys. Rev. Lett.* 80 (1998) 2338–2341.