DESIGN AND LOADING OF
DRAGLINE BUCKETS

Draglines are an expensive and essential part of open cut coal mining. Small improvements in performance can produce substantial savings. The design of the bucket and the way in which it fills with overburden are very important to the overall dragline performance. Here we use a numerical model to simulate this filling process and to differentiate between the flow patterns of two different buckets. Extensions to the model are explored.

1. Introduction

Draglines are used to remove the overburden, that may be up to 50 m in depth, exposing the coal deposits beneath in open cut coal mining. They use huge buckets, up to 100 m³, which are dragged up the sides of the pit and fill with pre-blasted overburden. There are over 60 such large walking draglines operating in Australia. They are an expensive and essential part of open cut coal mining. It is estimated that a 1% improvement in dragline performance would save tens of millions of dollars annually. Their performance is affected by many factors, including how the dragline is operated, where the material is taken from and where it is deposited, and the positioning of temporary bridges. Here we study the actual filling process of the bucket.

The aim of this Study Group problem was to determine whether the filling of a dragline bucket can be simulated using a modern particle based computer code and to evaluate the potential of such numerical models in suggesting bucket modifications that could lead to improved designs.

The computer code used follows the motion of every particle in the system, predicting their trajectories and collisions, both with the bucket and with other particles. It has been applied successfully to modelling long run-out landslides and grinding pin-mills (Cleary, 1991). For industrial applications, the method must be able to model complex boundary geometries. This capability has been added to the code specifically for this application. However, it is sufficiently general to allow modelling of a wide range of other applications.

We examined various aspects of draglines to determine which ones are important in the modelling. We also explored the relationship between rheological properties of the spoil material and particle properties in the model. The numerical method and collision model are described as are the extensions required to implement the complex boundary geometries. We also examined a range of further extensions that would enhance the applicability and range of applications of the model.

Finally we present simulations for two different bucket designs, using both uniform particles and particles with a moderate size distribution, and interpret these results. Validation and applicability of these modelling techniques to improvement of bucket design and operation are discussed.

2. Dragline buckets and dragline operation

Draglines are huge earth moving machines. They operate by lowering a bucket to the surface of the overburden to be excavated and dragging it along the slope towards the machine. A typical dragline operation is shown in figure 1. As the bucket, shown in figure 2, moves across the ground, its teeth bite into the rock and earth, breaking and loosening it. This material is then lifted by the teeth and the front lip of the bucket and flows inside, eventually filling it. The bucket is then lifted as the dragline rotates. Finally, the spoil (the material in the bucket) is dumped onto a spoil pile. This is located in a previously mined area. The dragline excavates down to the coal seam, which can then be mined, once it is fully exposed. The dragline *walks* slowly along the edge of the pit. More detailed information on the actual operation of the draglines can be found in the reports of the other two dragline problems in these Proceedings.
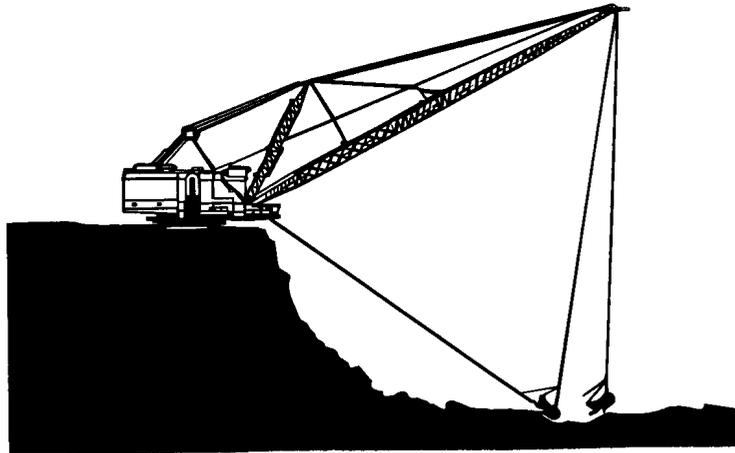

Figure 1: A cross-section schematic diagram of a walking dragline removing overburden.

This problem does not deal with the overall operation of the dragline, but concentrates on the actual filling process of the bucket. Improving the performance of the bucket is only one of many aspects that should be considered for improving the efficiency of the entire operation.

In this work we examine the filling of two different bucket designs. The first, an Esco bucket, is about 4.2 m long (from the lip to the back) and 2.5 m high at the back.

Figure 2: An actual bucket is pictured as it is lowered to the ground in preparation for dragging.

The second, a BE HPS bucket, is about 5.6 m long and only 1.5 m high at the back. It is the height of the back that is important in these two dimensional calculations, rather than the height of the sidewalls. Figure 3 shows side views of the two buckets and the expected spoil profiles from work done by ACIRL.
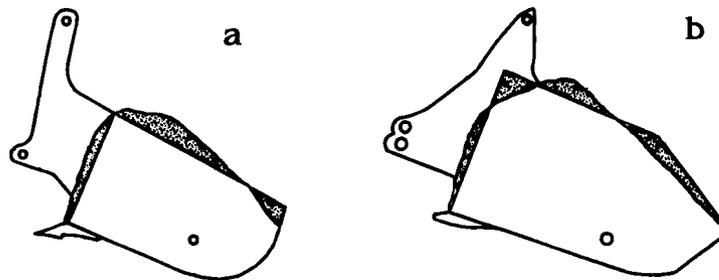
Figure 3: The two bucket designs, ESCO and BE HPS, used in the simulations. They also show the expected spoil profiles.

Aspects of bucket design and filling that were identified as being very important to the performance of the dragline were:

- The teeth on the front of the bucket. The purpose of the teeth is to break the rock apart and to begin lifting the overburden into the bucket. The purpose of the lip is to lift the remaining material, between the teeth, into the bucket.

- The lip/teeth geometry. The ratio of lip width to tooth width, the length and attack angle of the lip and the teeth are all expected to be important to the performance of the bucket. To evaluate these effects a three dimensional model would be required. Given the regular nature of the teeth, this would best be handled by using periodic boundary conditions in the transverse direction and only modelling

the front section of the bucket. This would reduce the computation to a feasible size.

- The overall width/length ratio of the bucket. This is constrained by the desire to use the smallest amount of steel in the bucket, thus maximising the amount of material moved for a given amount of lifting power. In real bucket filling one observes that, in the absence of very large rocks, the bulk flow into the bucket is relatively two dimensional. That is, the motion in the transverse direction across the bucket is small.

- The material properties of the spoil varies substantially, depending on the type of material, the blasting, water content, and whether it is being rehandled. Rehandled material has a much higher fines content and generally has a wider size distribution. It was concluded that the case of filling with relatively large particles was more important than the cases involving large fines content.

- The bucket sinks into the ground as it fills, requiring increasing tension in the upper cables in order to prevent the bucket from burying itself. The bucket angle changes in response to variations in the balance of the array of forces that act on the bucket. A bucket and all these forces are shown in figure 4. The forces include the tension in the drag lines T_1, the weight of the bucket W_b and the weight of the spoil in the bucket W_s and its distribution, the collisional force of the particles on the lip and teeth of the bucket F_c and the force S_r applied to the back of the bucket by the spoil. This is the opposite force to the one applied by the bucket to the spoil that slows its movement and causes it to stop in the bucket. Precise details of the rigging and support lines vary from one bucket and operation to another. Here we combine all the forces from the rigging into the force T_2. For our simulations, with fixed buckets, the precise details of T_2 are unimportant. They would, however, be crucial in any simulation where the bucket moved dynamically. The moments, around the center of mass of the bucket, generated by these forces determines the rotational motion of the bucket.

- Cohesive forces between particles are important for between one third and one half of all materials handled. These materials have significant clay contents and stick together when compressed. This has a strong effect on the spoil profile in the bucket and more importantly on its density. In particular, cohesive materials are found to be more dense towards the back of the bucket than they are at the front. This relates to the voids being squeezed out and being unable to reform because of the plastic nature of the material.

- The remaining half to two thirds of all material handled is friction dominated. Here cohesive forces are negligible or unimportant. This type of material is ideal for simulation with the present code.

The most important parameters that are relevant to dragline operation are given below. These are used either as input to the model or are to be compared with the output of the model to help in its validation.

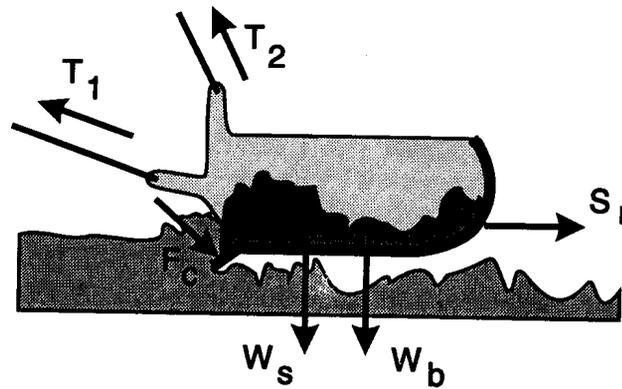|	
Dragline/bucket speed	$1.75 - 2.0$ ms^{-1}
Average slope angle	22°
Average bucket length	5 m
Drag length	4-6 bucket lengths : 20 - 30 m
Filling time	10 - 20 s
Average spoil density	1.6 g/cm^3
Typical angle of repose	$37 - 40^{\circ}$
Internal angle of friction	40°

Figure 4: The forces that act on the bucket and determine its angle and motion. T_2 represents the combined effect of all the forces applied by the rigging to the bucket and is not intended to model the entire interaction.

3. The modelling method

The modelling technique is a soft particle granular flow method. It has been successfully used to simulate long run-out landslides and grinding pinmills (Cleary, 1991), ice pack behaviour (Hopkins, 1991), rock fracturing (Potapov *et al.*, 1992) and many other applications. Other collision models, such as the hard particle model, are available but less suited to this application.

The algorithm has three key parts:

- A search grid is used to build a near-neighbour interaction list.

- All the forces on the particles are evaluated efficiently using the near-neighbour list.

- The particles are simultaneously moved by integrating the equations of motion.

A search grid with cell dimension $l_{cell} = \alpha\, d_{min}/\sqrt{2}$ is constructed to cover all the particles. Here d_{min} is the minimum particle diameter in the system and $0.8 < \alpha < 0.95$. This ensures that only one particle can have its center in each cell at a given time. This search grid is then used to build a near-neighbour interaction list. The particles are all labelled with an integer index. The labelling order is unimportant. Consider a group of circular particles (including particles with the labels i, j and k, where $j < i < k$). Figure 5 shows such a situation with the search grid overlayed.
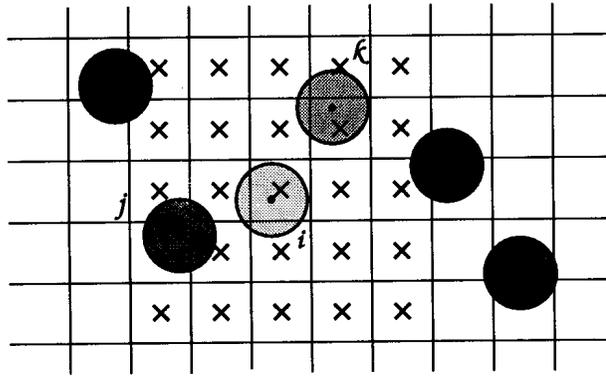

Figure 5: Search grid overlaying a group of uniform size circular particles.

The search is a two stage operation. Firstly, all the particles are mapped onto an integer array. Each array location, corresponding to a cell, stores either 0 (no particle) or a positive integer (the particle index). Appropriate inverse maps are also constructed.

For the second stage, consider the lightest shaded particle i. Using the array from the first step we can determine which cell this particle resides in. An appropriate number of cells n_c around this cell (indicated by crosses in figure 5) are examined by looking up the contents of the integer search grid array constructed earlier. Here $n_c = 1 + int\left((1 + \delta)(r_i + r_{max})/l_{cell}\right)$, where $0 < \delta < 0.1$, r_i is the radius of particle i and r_{max} is the radius of the largest particle. For uniform particles this is two cells. This represents the maximum number of cells that must be searched in order to guarantee that every collision, even for non-uniform particles, is detected.

If the array contains a zero for a particular cell then the cell is empty. If it is not empty, the particle index k (belonging to the slightly darker particle in figure 5) is returned. If the distance between the centers of the two particles is small, $|x_i - x_k| < (1+\delta)(r_i+r_k)$, then the pair of particles should be considered as candidates for collision. If $k > i$ then the pair of indices are added to the near-neighbour list. Since the order of particles in the pair is unimportant, the inequality is used to ensure that the pair is added to the list only once. In the example above, the particle pair (i,j) has previously

been added and this time is omitted. The dark shaded particles lie outside the search region, indicated by the crosses, and are not considered in the interaction calculations.

The near-neighbour list is rebuilt every n_s timesteps. n_s can either be specified or calculated dynamically using a worst case overlap criterion.

All previous applications of this method used very simple boundaries. These were either periodic in one or both dimensions or consisted of one or two long flat plates. These normally represent the ground and their only function is to support the flowing granular material.

For applications such as modelling dragline bucket filling, we need to be able to model objects with complex geometries. This requires a powerful, flexible and convenient way of prescribing and handling these boundaries. These facilities have now been added to the codes. Nearly arbitrary shaped boundaries can now be specified as a sequence of piecewise linear and circular segments. These can be linked together to form almost any shape. In this implementation the boundaries are all fixed. The generalised boundary segments affect the code in only the first and second parts of the algorithm.

In the search phase, a virtual marker particle is placed in each cell through which the boundary center line passes. This involves calculating the intersection of the boundaries with the search grid. This is shown in figure 6 for both boundary types. A cross marks each of the cells into which a virtual marker particle is added.
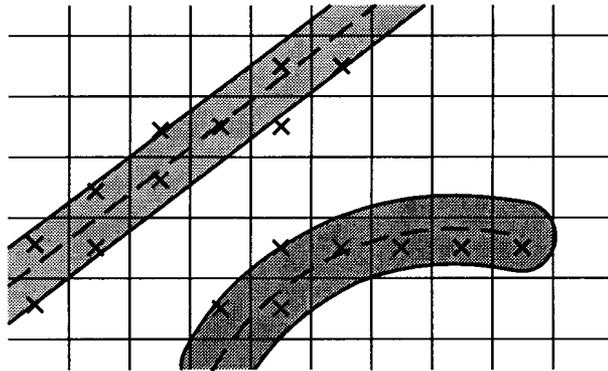

Figure 6: Linear and circular segment boundaries and virtual particles.

The boundaries have a thickness that can be specified. This has no effect on the particle-boundary interactions, except to determine where they occur. The boundaries, even thin ones, are rigid but slightly squashy. Here we make them the same diameter as the particles. The boundaries also have curved ends. This eliminates difficulties with particles striking sharp corners on the boundary. The search algorithm detects boundary particles in the same way it detects normal particles, without any modification.

The complexities of the search algorithm are necessary to reduce the computational overheads involved in calculating the collisional forces. If a straight-forward N-body calculation were performed then the number of operations required per timestep would be proportional to N^2. For even moderate particle numbers N this becomes prohibitively expensive. Using the above search procedure the number of operations is proportional to $(d_{max}/d_{min})^2 N$. For uniform sized particles the variation reduces to $O(N)$. That is, the computational time increases linearly with the size of the simulation. For very large size distributions, the cost of the search can also become very expensive as the coefficient varies quadratically with the ratio of largest to smallest particle sizes. This makes modelling very small particles, such as fines, expensive, but not impossible.

The second part of the algorithm involves determining which particles are actually colliding and evaluating all the resulting collisional forces on the particles. The particles are allowed to overlap as they move around. Thus the particles are described as *soft*. Only particle pairs that are in the near-neighbour list are examined for potential collisions. If the centers of the pair are closer together than the sum of their radii, then the particles are overlapping and they are deemed to be colliding.

For each pair of particles that are actually colliding the resulting forces are determined by the spring-dashpot model shown in figure 7. The normal force F_n has a spring component to provide the repulsive force that pushes the particles apart and a dashpot that provides dissipation resulting in an effective coefficient of restitution. The tangential component has a spring that is subject to the frictional limit of μF_n, where μ is the dynamic friction coefficient.

$$F_n = -k\,\Delta x + c\,v_n$$

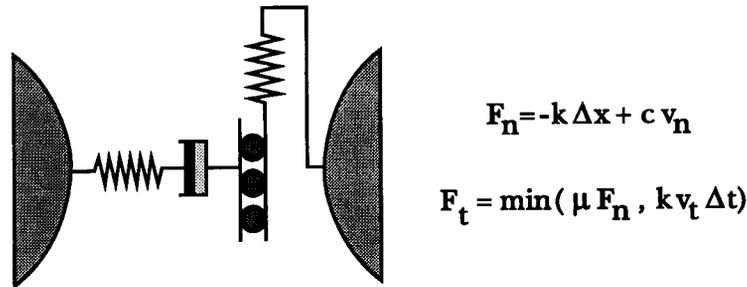$$F_t = \min(\mu\,F_n,\ k v_t\,\Delta t)$$

Figure 7: The soft particle collision model. The normal force uses a spring and dashpot. The tangential force uses a spring with a frictional limit.

For particle-particle collisions of circular particles the normal force is along the line of centers and the tangential force is orthogonal to the normal force. Whenever there is surface friction between colliding bodies, these bodies spin after impact. Since our particles are circular it is necessary only to calculate the rotation rate for each particle and not their orientation. The spin of the particles is very important. It is a substantial mechanism for the generation of granular temperature near boundaries (Campbell,

1990), and affects the dynamics throughout the entire system. Granular temperature is a measure of the fluctuating component of the kinetic energy and is analogous to the turbulent kinetic energy in fluid dynamics and to the normal temperature in thermodynamics.
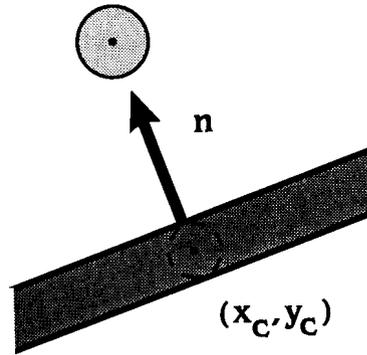
Figure 8: A particle approaching a linear boundary. A virtual particle, placed opposite, represents the boundary.

For particle-boundary collisions a virtual boundary particle (as shown in figure 8) is placed directly opposite the incoming particle, so that their line of centers is orthogonal to the surface tangent. The normal vector and the virtual particle location for a linear boundary is shown in figure 8. The curved boundary situation is comparable. The forces are then evaluated in precisely the same way as for a particle-particle collision (figure 7).

Complex boundary geometries require substantial book-keeping. This is the main difficulty in their implementation. Calculating the location of the boundary particle, the normal, checking that each particle collides with only one virtual particle from each boundary it contacts, are all straightforward, if tedious calculations. The important aspect of this part of the algorithm is that it must also be efficient. We do not want the boundary particle interactions to seriously degrade the overall computational speed.

The size of overlap between particles is determined by the stiffness k of the spring in the normal direction. Typically average overlaps of $0.1 - 1.0$ % are desired, using spring constants of the order of $10^3 - 10^4$. The damping coefficient is calculated from the user prescribed coefficient of restitution ε and the spring stiffness k by

$$C_d = 2\gamma \sqrt{m_{av} k}$$

where

$$\gamma = -\frac{\ln(\varepsilon)}{\sqrt{\pi^2 + \ln^2(\varepsilon)}}$$

This arises from the analytic solution of the collision equation for two such particles with average mass m_{av}. The particles all have the same damping coefficient. This means that when using large size distributions, the effective coefficient of restitution will be somewhat dependent on the diameters of the particles involved.

The third part of the algorithm involves simultaneously moving all the particles in response to the forces that have just been calculated. This is done by integrating the equations of motion:

$$\dot{\mathbf{x}}_i = \mathbf{u}_i \tag{1}$$

$$\dot{\mathbf{u}}_i = \sum_j \mathbf{F}_{ij} + \mathbf{g} \tag{2}$$

$$\dot{\omega}_i = \sum_j M_{ij} \tag{3}$$

where $\mathbf{x}_i, \mathbf{u}_i$ and $\mathbf{F}_{ij}$ are the cartesian position, velocity and collisional forces on particle i, and ω_i is the rotation rate produced by the moments M_{ij}. $\mathbf{g}$ is the gravity vector. The integration scheme is a second order predictor-corrector. Between 20 and 50 timesteps are required to accurately integrate each collision. This necessitates very small timesteps. If the integration is accurate then the coefficient of restitution from which the damping coefficient was calculated is recovered. A higher order integration scheme using a larger timestep may initially seem attractive, but the timestep is also constrained by the requirement that for two particles moving together, the initial overlap is at the very worst 10%. In many cases, this is a more stringent constraint on the timestep than the earlier one. This removes the advantage of the higher order scheme. Overall the timestep is given by

$$\Delta t = \min\left(\frac{\pi}{25} \sqrt{k/m_{av}(1 - \gamma^2)} \ , \ 0.1\, d_{min}/n_s\, U_{max} \right)$$

The second constraint uses the maximum speed in the entire granular flow U_{max} and the diameter of the smallest particle d_{min}.

The inclusion of other forces on the particles is a relatively straightforward matter of calculating these additional forces and including them in the force summation (2) and (3) at each timestep.

4. Making the model fit reality

The behaviour of the numerical model is governed by the parameters ε, μ and k. They must be chosen so that the resulting particles behave in the same way as real rocks. It is also desirable for large masses of these particles to have similar rheological properties to those of the real material. Rheological properties are controlled by several effects including those above, particle size and shape distribution, and other effects such as cohesive forces, interstitial fluids, particle density distributions and many more. With the exception of the cohesive force and the interstitial fluid, all the other effects can presently be included in the calculation.

The difficulty is not in their inclusion, but in quantifying them. What particle size and density distributions are needed? What is an appropriate coefficient of restitution? How much does it vary? Little, if any, work has been done on this problem. Variations in moisture content, density, chemical composition, hardness, brittleness, friction, elasticity and microstructure can cause appreciable changes in the flow properties. There is no established method of *completely* characterising particulate materials or relating them to our model parameters. At present there is no substitute for comparison with experimental measurements for the purpose of calibrating the model to the *specific* material involved in the application.

It is not just a matter of being unable to relate the model parameters to the real material. The real particulate material and the relationship between its characteristics and the flow behaviour are not well understood. This is no longer the realm of the mathematics, but of materials science.

A systematic study is required of the bulk properties obtained from different particle configurations and properties and how they change as the model parameters vary. This exercise is substantially beyond the scope of this workshop. Experimental calibration to real materials is unavoidable.

Some parameters can be chosen sensibly. The friction coefficient can be reasonably given by $\mu = \tan \theta$, where θ is the average slump angle of the spoil or the internal angle of friction. Both are a measure of the relative surface friction between particles. Rocks do not bounce very much. They are very inelastic. We chose a coefficient of restitution of $\varepsilon = 0.1$. Studies of long run-out landslides (Cleary, 1991) indicate that, for small ε, the flow behaviour is not sensitive to the precise value. The important thing is that almost all the relative kinetic energy is dissipated in every collision. This highly dissipative environment tends to make the active regions quite small. The areas of high granular temperature are small. The flow tends to have liquid rather than gaseous behaviour. Substantial solid or crystalline regions are anticipated.

The size distribution is expected to play an important role in the flow behaviour. Uniform circular particles exhibit little resistance to shear forces. The layers of the

crystalline microstructure slide easily over each other. Materials with non-uniform size distributions have much higher resistances to shear. The internal friction angle of the spoil material is similar to its slump angle. This suggests that the geometrical shape effects on the material's flow pattern are not significant.

Fines are a computational difficulty. The computational time required is proportional to the square of the ratio of the largest to the smallest particle diameters. Resolving dust is therefore very expensive. Alternatives to actually modelling the fine particles are ignoring them, treating them as an interstitial fluid or treating them as a lubricant. The last of these possibilities would manifest as a reduction in the magnitude of the dynamic friction from the value calculated from the slump angle. Calibration would need comparison with experimental results.

Voids

Reasonable density variations are found in many of the spoils. These result from variation in the void fraction of the material. The void fraction is determined by the size, geometry and surface properties of the particles.

Voids are rare in granular systems containing uniform circular particles. When they settle, they have a very strong predisposition towards a hexagonal microstructure. The only voids that are seen here are when one of the sites in the hexagonal structures is vacant. The dynamic nature of deposition processes means that the collisional impulses are almost always strong enough to perturb the system and fill the voids.

Voids smaller than one particle diameter never occur if the particles are free to choose their final state. Such voids only occur when particles are placed in a confining space whose dimensions are a non-integer multiple of the particle diameter. Even then there will usually only be one small void per layer of particles in the final microstructure.

Voids larger than one particle diameter are rarely seen. Such large void structures are only possible if there is a symmetric bridging structure over the void. These bridges are supported by a network of normal stresses and are unstable. If one particle is out of place it will experience a net force away from the support line of the bridge. It will then either move towards or away from the void. The surrounding particles will move in the opposite direction. In all cases the particles will rearrange themselves and the bridge will collapse filling the void, at least partially. Large voids are therefore extremely unlikely to exist. This is mostly because the collisional forces in a flowing system perturb the symmetry of the bridge causing it to collapse. This is a genuine physical instability. Even in a static system, a symmetric bridge is likely to collapse. This is a numerical phenomena. It may be due to either roundoff error eventually perturbing the symmetry, but is more likely to occur because of the soft nature of the particles. In large bridges, some particles would squeeze through between the other particles, even though the hole is theoretically too small. This would only be a problem when modelling bridges that are several particles across in a completely static system.

Voids in real systems are almost entirely geometry dependent. The most important aspect is the variation in the sizes of the particles. Circular particles stack best when uniform in size. The larger the size distribution, the more variable is the packing. The final void fraction will be strongly dependent on the dynamics of the deposition process. If the time scale for deposition is long compared to the percolation or segregation time scale, the particles will pack quite well. Most of the voids between large particles will be filled by smaller particles. If the deposition is very rapid then large voids may be frozen into the microstructure, with the smaller particles having insufficient time to migrate into the void areas.

The shape of the particles will have an effect on the void fraction. Convex particles will pack in very much the same way as circular particles with the same size distribution. Concave and highly irregular particles will have a higher void fraction than a similar distribution of circular particles. It is more difficult for such irregular particles to move and rotate into an optimally packed microstructure, where compatible shapes must fit together. In most cases it would be expected that the relevant time scale would be too long for these extra voids to fill. Conversely, such irregular shaped particles would have a much higher tendency to fracture into more regular and better packing shapes. Modelling non-circular particles will be discussed later.

At present no attempt has been made to model density variations in the spoil. The uniform circular particles give no information on this matter. Using circular size distributions is the most important step in predicting the variation. It is expected that cohesion between the particles will be the next most important phenomena. The particle shape is considered to be the least important contributor.

5. Further extensions to the model

This section describes several extensions to the model that are desirable if we are to effectively model real granular flows, such as the filling of dragline buckets with real materials. These include dynamic boundaries, static friction, cohesion and non-circular particle geometry.

5.1 Dynamic boundaries

A full simulation of the filling or dumping of a dragline bucket requires the capability to handle fully dynamic boundaries. A real bucket moves and changes angle in response to the balance of forces shown in figure 4. It is necessary for the method to be able to evaluate all these forces and moments and then to correctly move all the boundary elements that compose our model bucket in response to them. This is principally a book-keeping operation and is not theoretically difficult. The component forces from each individual collision on the bucket are already calculated during the

simulation. They need to be summed correctly and additional rigid body equations of motion, using only these forces and moments, must be solved.

5.2 Static friction

The granular flow model described above has only a dynamic friction. The frictional force only occurs for particles that are in relative motion. The absence of a static friction allows piles of particles on flat surfaces to collapse. In order to obtain realistic angles of repose, a static friction model is required. It needs to be based on sound mechanical principles and needs to be computationally efficient.

Consider a stationary pile of uniform disks, as shown in figure 9a. The weight of the disks is supported by a network of normal stresses between the centers of all the touching particles. The forces on each of the particles above the bottom layer cancel resulting in no net forces on them. Each of the particles in the bottom layer, however, experiences a net force parallel to the surface. The dynamic friction depends on the relative tangential velocity. If the particles are stationary then there is no frictional force and the net force on these particles in outwards. The force balance on the right hand end particle is shown in figure 9b. This particle therefore moves sideways in response to this net force. The end particles no longer support those above and the pile collapses.
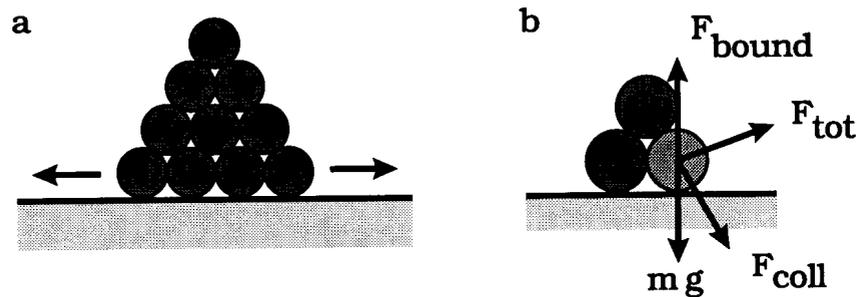

Figure 9: (a) A pile of particles collapses, (b) the forces on one particle that is in contact with the boundary in the absence of static friction.

In reality the particles in the bottom layer should not move unless the net force parallel to the surface exceeds the static frictional limit. This is given by the product of the coefficient of static friction and the normal force between the surfaces.

Our proposed algorithm would function as follows:

- When evaluating the forces for each collision, check whether one of the particles belongs to a boundary and if the relative velocity is small, $|v_i - v_b| < \varepsilon$. If this is the case, store the indices of the particle pair, the value of the normal force between them N_{ib}, and the unit vector between the line of centers. This pair is a candidate for a frictional bond.

- Later, after the net force on each particle has been calculated by summing all the appropriate forces, determine if the frictional bond is to be broken. Resolve the net force $\mathbf{F}_i$ on particle i into normal N_{ib} and tangential T_{ib} components, using the stored unit vector. The actual net tangential force T_{ib} is then compared to the limiting static friction νN_{ib}. If $T_{ib} < \nu N_{ib}$, then the tangential force is too weak to break the frictional bond and the tangential force is subtracted from the net force $\mathbf{F}_i$. Otherwise the tangential force is strong enough and the particle moves independently of the surface, according to the calculated net force. If the particle moves along the surface then the normal dynamic friction acts automatically at the next timestep.

- This is a two step calculation. The first occurs during the normal force evaluation. No new variables are needed, merely storage of already calculated values. The second step is effectively a filter on the net forces to remove any motion that should be prevented by the frictional bonds. The overheads are proportional to the number of static bonds to be tested. It does not involve substantial changes to the present algorithm and is expected to be very efficient.

Static friction between particles is not expected to be important. All stable void structures are related to geometrical properties of the particles. There are no circumstances under which a purely static frictional force will prevent particle microstructure from changing. The dynamic friction and the particle shapes determine the stability of these structures.

5.3 Particle geometry

In the algorithm described in section 3, the particles were all circular. In general this is expected to be quite adequate for modelling most flowing granular materials. The size distribution seems to have a much greater effect on the flow behaviour. In active, rapidly shearing flows, such as landslides, the particles will be ground down and rapidly approximate a circular shape. On average the variation in collision angles due to non-circularity will cancel out and the bulk flow should be similar. The precise dynamics of individual collisions will of course be different. The particle geometry is expected to have a larger effect when the particles are very highly elongated or when fracturing of the particles needs to be considered. In more quiescent granular flows, such as dragline bucket filling, or solids the geometry can play a more important role in determining the flow pattern and the shear strength of the microstructure.

Nevertheless there are circumstances, such as with highly elongated particles, where the shape of the particles needs to be considered. Hopkins (1991, 1992) has modelled ice ridging processes using rectangular particles. This soft particle polygonal approach has only been adapted to fracturing of bodies by Potapov *et al.* (1992).

There are two types of non-circular particles that can be used:

- Spheroids; these could be prolate, oblate or triaxial. Their aspect ratios can be varied to mimic the essential features of real particles. These have the important features of the normal force contributing to the spin and not lying along the line of centers, while still being smooth.

- Polygons; these can be arbitrary polygons, arbitrary convex polygons, convex polygons with a fixed number of faces, or regular polygons. These particles can mimic a very wide variety of particle geometries, but have the disadvantage of being very hard to use and of having sharp corners. Simulations using these particles are described as between very slow and grotesquely slow (Campbell, 1993, private communication).

The effects of particle shape using spheroidal particles has been examined by Rothenberg and Bathurst (1992). Polygonal particles have been more popular (Hopkins, 1991, 1992; Potapov *et al.*, 1992). In the remainder of this section we explore some of the aspects of the polygonal particle model that need to be improved and need to be made substantially more computationally efficient. We restrict our considerations to arbitrary *convex* polygonal particles, because of the complexity of the calculations involved in the collisions for more general classes of particles.

The geometry is expected to affect the algorithm in several ways:

1. Vertex data; the vertex data must be stored in the most efficient way possible, to minimise the geometry calculations.

2. The area or mass, the center of mass x_c and moment of inertia I of all the particles must be calculated and stored. This is best done by dividing the polygons up into triangles and performing the appropriate integrations over the triangles. $x_c = \int x\, dm / \int dm$ and then $I = \int r^2\, dm$, where $r = |x - x_c|$.

3. Collision detection; it is necessary to detect when collisions of arbitrary polygonal particles occur.

4. Collision location; the parts of the polygonal particles actually involved in the collision must be ascertained.

5. Forces; the direction and magnitudes are required.

6. Extra equations of motion.

The detection of collisions should be a two part process. An effective radius should be found for each particle. The entire particle should be enclosed by a circle of this radius centered on the center of mass. This effective radius would then be used in the normal grid and search operations, described in section 3 for circular particles. This
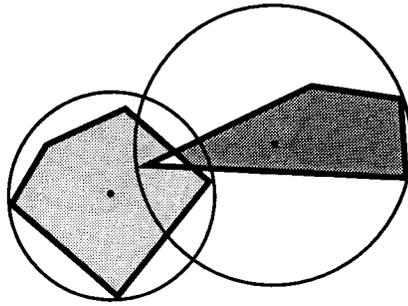
Figure 10: Two colliding convex polygonal particles and their effective radii.

then means that the potentially most time consuming step in the algorithm is equally fast for all particle types.

The detailed geometry of the particles would only be consulted when the two circles defined by the effective radii intersect, as shown in figure 10. This cannot be avoided. The simplest approach, and the one traditionally used, is to compare all p vertices on the first particle against all the q vertices on the second particle to detect whether the particles actually overlap. The total computational time for this detection is then proportional to $p \times q$. For particles with small numbers of sides, such as triangles and rectangles, this is not too bad. For complex particles it becomes prohibitive.

Strategies for reducing the computational cost of detecting and evaluating overlaps or intersections of arbitrary convex polygons can be found in a range of computer graphics and geometry texts. Preparata and Shamos (1985) in *Computational Geometry* describe an algorithm in which the polygons are divided by vertical lines passing through each of the vertices into a sequence of trapeziums. Such an arrangement is shown in figure 11a. Using these trapezium to detect and evaluate the overlap reduces the computation time to order $p + q$.

This algorithm is designed to find any overlap of the polygons and does not use the extra information that is available in this situation. The particles do not appear on top of each other with arbitrary initial overlap. They gradually move towards each other, until a tiny overlap appears. This then increases and ultimately decreases before vanishing as they separate. There is no point in checking for intersections between the vertices on the far sides of the polygons. Our suggestion, shown in figure 11b, is to use the line through the intersection points of the circles constructed from the effective radii, instead of vertical lines. This test line is then moved sideways, in both directions, to form a sequence of trapeziums, that are progressively further away from the initial line, until the entire intersecting region is isolated. This then involves the smallest number of such trapezium constructions and will be the most efficient such method. This does not eliminate the possibility of even faster methods based on other approaches.
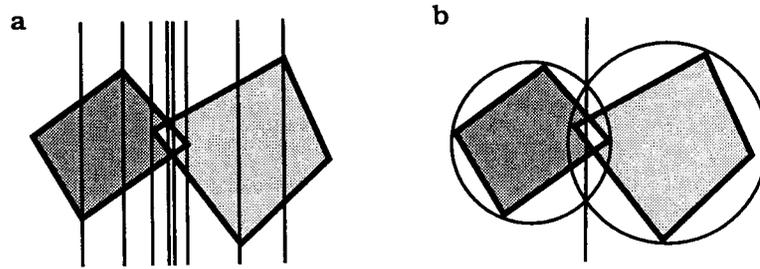
Figure 11: (a) Dividing the particles into trapeziums, (b) the line through the points of intersection of the equivalent circular particles is a good starting line to minimise computations in collision determination.

The collision model is still the soft particle one shown in figure 7 for circular particles. The magnitude of the repulsive force is proportional to the area of the overlap rather than the linear overlap (Hopkins, 1992). The direction of the force is definitely non-trivial. For a corner colliding with a face, the normal force is orthogonal to the face. For the case of two faces colliding, the normal force is the orthogonal to both faces. For the case of two corners colliding (remember soft particles overlap so this is a real possibility), the direction of the normal force is undefined. It is not presently clear what the best choice of direction is. All the methods considered are very sensitive to the precise locations of the vertices and the amount of overlap along each face. Very small changes in the orientations of the particles can produce substantial variations in the direction of the resulting collisional forces.

Once the forces are evaluated they are used in the integration of the equations of motion. Importantly the moments acting on the particles now consist of two parts; one from the tangential friction and one from the oblique normal force that no longer necessarily lies along the line of centers. Additionally the orientation of the particles must be calculated so that 6 rather than 5 equations of motion must now be integrated for each particle.

5.4 Cohesion

The inclusion of attractive particle forces such as cohesion is considered desirable. Between one third and one half of all the materials handled by draglines are cohesive materials with substantial clay components. Not only does the cohesive force affect the flow properties, it may also be an important contributing factor in producing the density variation that is found in the spoil.

An efficient and physically reasonable cohesion model and algorithm is required to simulate such sticky materials. The most straightforward possibility is to join colliding particles with a center-center spring. The particles would only separate if there were sufficient repulsive or shearing force. A fundamental problem is the existence

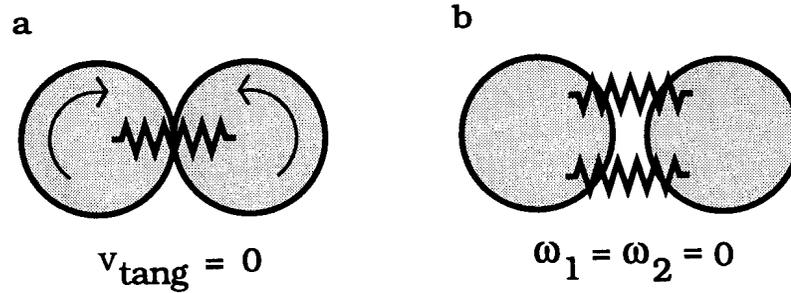$$v_{tang} = 0 \qquad \omega_1 = \omega_2 = 0$$

Figure 12: (a) A single spring cohesion model, (b) a double spring model.

of stationary, but oppositely spinning pairs of connected particles. Such a situation is shown in figure 12a. A true cohesive bond requires the particles to have no relative velocity and no rotational motion. One possibility is to use two parallel springs, as shown in figure 12b. This variation, using forces that are no longer center to center would involve substantial computational overheads. No ideal model for the cohesive forces has yet been developed. However, for groups of cohesive particles, the rotation problem of the single spring model is eliminated by the dynamic frictional contact between the many particles. It is for isolated binary pairs that this problem arises.

6. Numerical results

Numerical simulations using the algorithm described in section 3 were performed for the two bucket designs shown in figure 13. Both initial sequences were set up identically. Only the bucket design was changed. Figure 13 shows the two initial configurations.

Since the boundary elements of the bucket are not presently able to move, we change to a reference frame that moves with the bucket. This is also very convenient for visualising the results, since the bucket always remains in the center of the picture. The overburden then moves into view from the left, passes or enters the bucket, finally passing out the right side.

The overburden is pre-blasted and broken up by the teeth (not modelled here), so it is well modelled as independent lumps of material loosely stacked on the ground. For overburden that is not loose, a cohesion model as described above would be required. The overburden, in figure 13, consists of uniform 20 cm circular particles stacked in a hexagonal microstructure. They are two layers deep under the bucket and 7 layers deep on the left. The lower region is a crude representation of the ground after the bucket has previously been dragged along it. The gravity vector is inclined at 22° to the horizontal. Even though the material looks horizontal, the bucket is actually being dragged up a slope inclined at 22°. Initially the particles used were uniform in size with
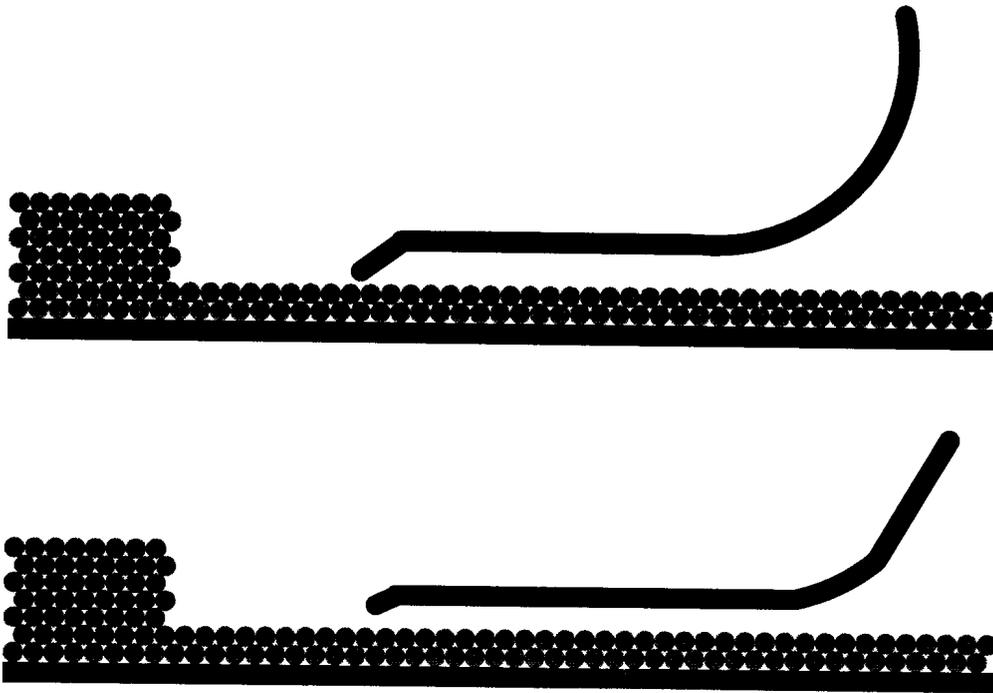
Figure 13: (a) Initial configuration for bucket 1, (b) initial configuration for bucket 2.

diameter 20 cm. This is a somewhat arbitrary size chosen for computational and video presentation purposes. Much smaller sizes can be used, if desired.

The overburden is a frictional material, with a coefficient of friction of $\mu = 0.75$. The coefficient of restitution used was $\varepsilon = 0.1$ and the spring stiffness was $k = 5000$.

Bucket 1, shown in figure 13a, is 5 m long and 2.3 m high, with a 0.5 m long lip that is inclined at 40° to the bottom of the bucket. Bucket 2, shown in figure 13b, is 5.5 m long and 1.6 m high, with a 0.25 m long lip that is inclined at 25° to the bottom of the bucket. When operating, the buckets do not bite deeply into the ground. They are dragged along at shallow depth, of around 0.5 m, and fill gradually by scooping up the small thickness of overburden above the level of the bucket. Due to the different sizes of the buckets it is not possible to have precisely the same depth. The two buckets are set up 0.4 m and 0.5 m below the surface respectively.

In the reference frame of the bucket, the material moves towards the bucket. The lip bites into the overburden, which is split into two streams. One passes under the bucket and one into the bucket. The filling of each of the buckets is shown at 4 time intervals in figures 14 and 15. Some details of the flow are common to both buckets, while others differ markedly. These are summarised below. As the bucket is dragged along, the spoil level increases until it reaches to just above the level of the top of the back of the bucket. The material then flows almost horizontally along the top of the

spoil. At this point the bucket is full and the bucket is lifted. The simulation is halted here and the final profile is shown in figures 14d and 15d.

The common and differing features of flow of a uniform size granular material into the two buckets are now summarised.

Common features for both buckets

- Both buckets have an initial rolling layer where material enters and, meeting no resistance, spreads uniformly over the entire bottom and up the back of the bucket.

- Material entering the bucket then surges onto the material previously deposited and then stops. The surging occurs because of the frictional contact between adjacent lines of particles. The force is initially too weak to produce relative motion. As the amount of material behind increases so does the shear force. Eventually, it overcomes the frictional force and shearing occurs along a fault line between two masses of material. The lower mass remains stationary, while the upper masses surges forward. It stops when its store of kinetic energy is dissipated by collisional interactions.

- Both buckets have realistic fill time of around 10-12 s for a bucket speed of about 2-2.5 ms^{-1}. This represents 4-6 bucket lengths for the fill.

- Both buckets continue to fill until the level of the spoil is slightly higher than the top of the back of the bucket.

- The final profiles, shown in figure 16, can be compared with the spoil profiles found experimentally using physical models and shown in figure 3. At the end of the simulations the spoil is almost level throughout the bucket, while the physical models show a slope down to the back from a peak about 1/3 of the distance from the front of the bucket. This will be discussed further below when we use non-uniform particles. However both video footage of the filling of real buckets and photos from manufacturer's brochures show that in many cases the spoil is actually quite flat on top. These differences are largely a function of the particle properties. In particular, uniform circular particles have little resistance to shear forces, so once the level rises above that of the back there is nothing to stop them flowing all the way over.

- These findings appear to be consistent with the observations of physical bucket filling models.
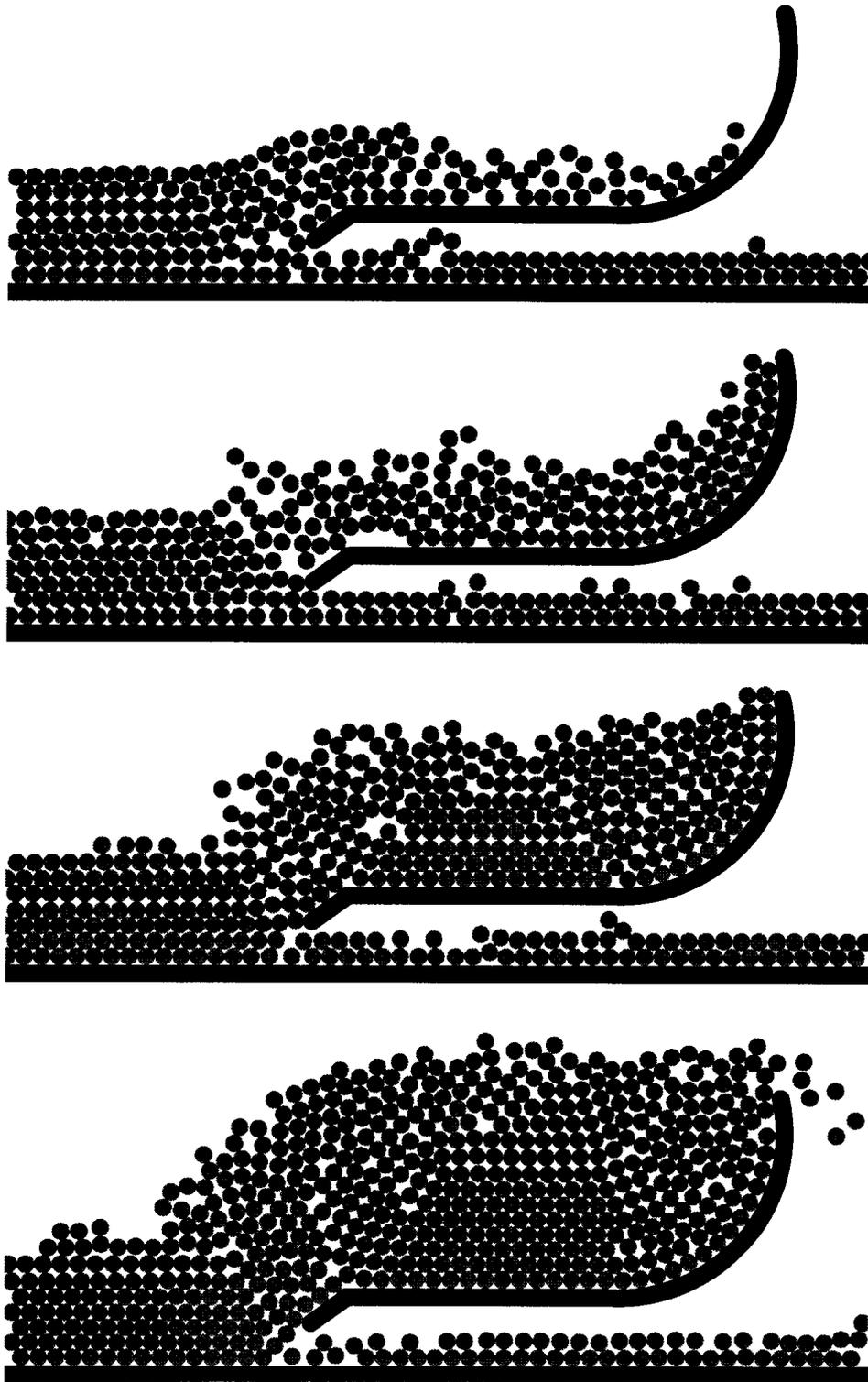
Figure 14: Sequence of snapshots showing the filling of bucket 1, at times $T = 2s, 3s, 5s$ and $8s$. The initial state is shown in figure 13a.
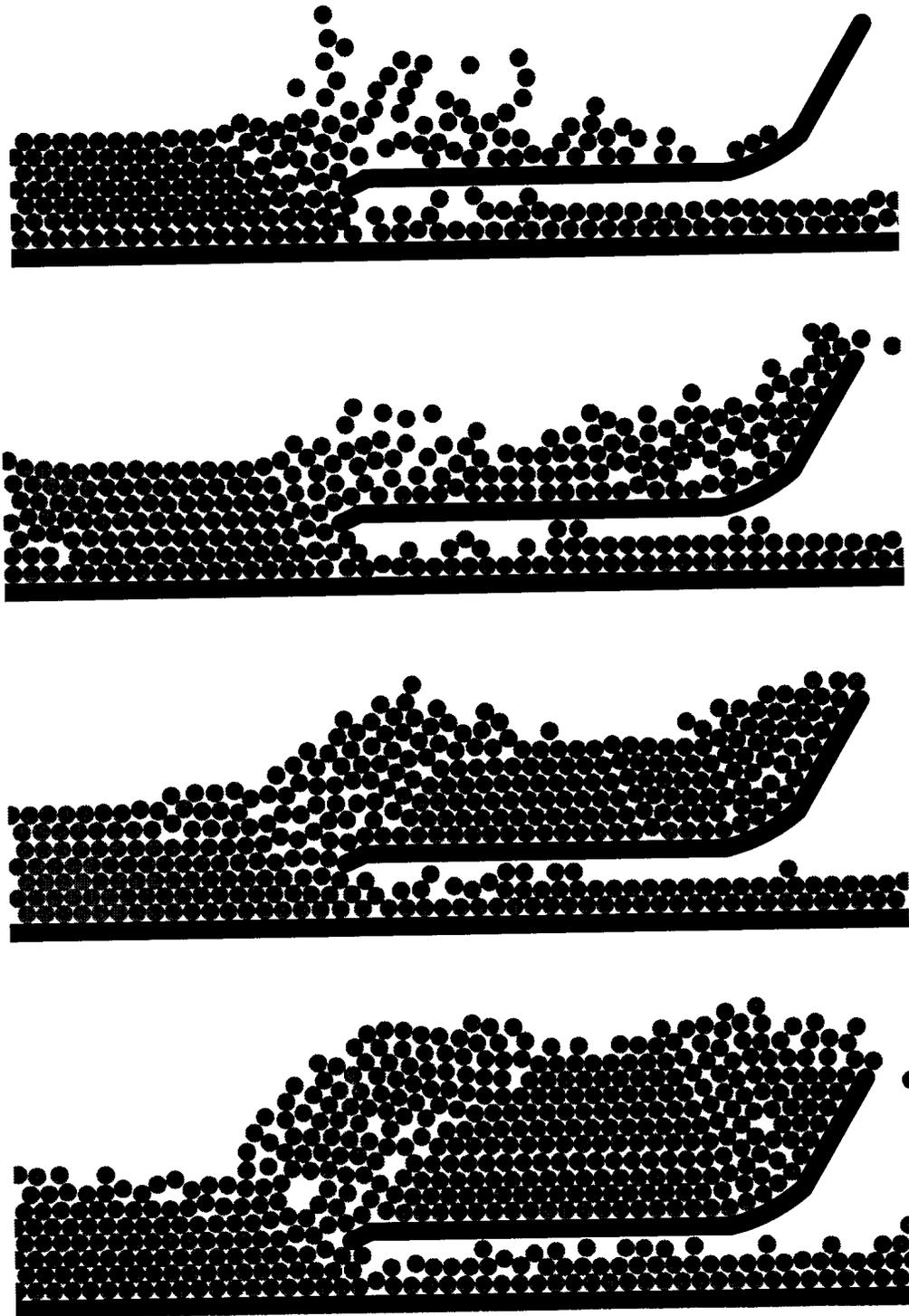
Figure 15: Sequence of snapshots showing the filling of bucket 2, at times $T = 2s, 3s, 5s$ and $8s$. The initial state is shown in figure 13b.
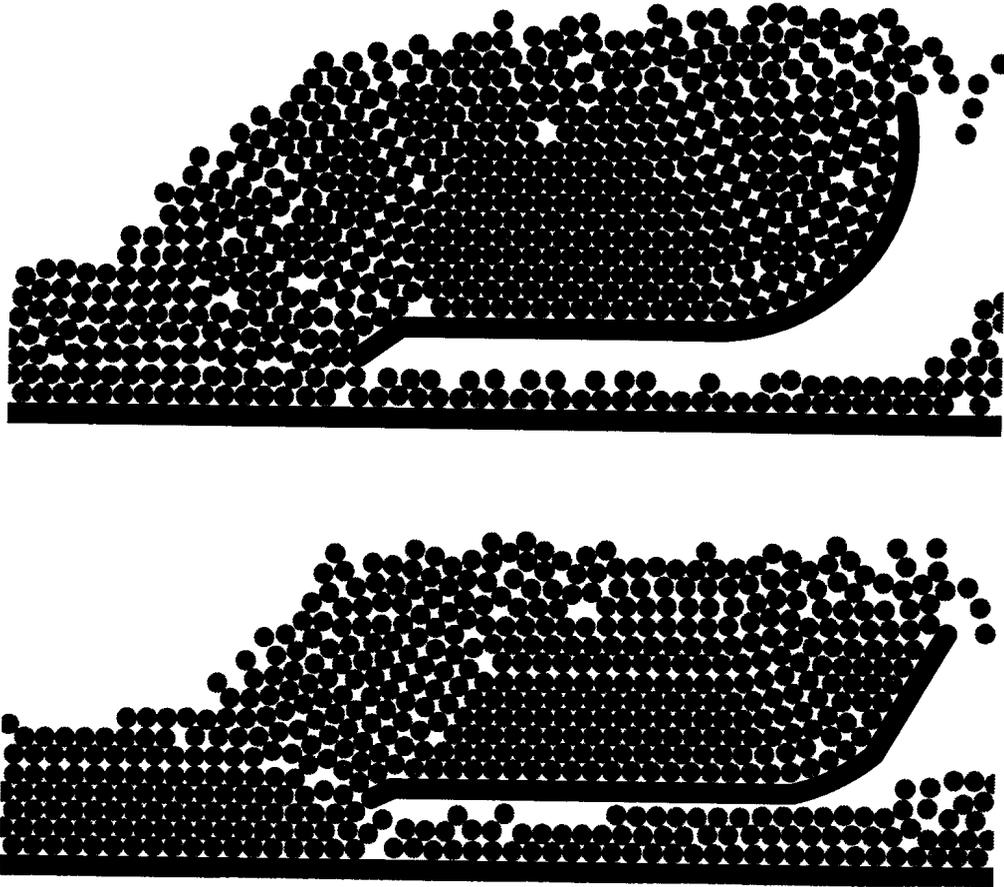
Figure 16: Final configuration of uniform size particles in (a) bucket 1, (b) bucket 2.

Features particular to bucket 1

- The filling is a two stage surging process. As the incoming material encounters the resistance near the lip of the bucket, an upward force is generated. After increasing for some time, it causes a triangular shaped region of granular material in front of the lip to surge upwards. This large raised mass pushes on all the material in the region above the lip. Eventually it causes a second triangular shaped region, adjacent to the first to surge up and into the bucket. This is a reasonably regular pattern with a whole range of fluctuations superimposed. Generally, the motion of a particle entering the bucket involves it first surging up and then later up and across into the bucket.

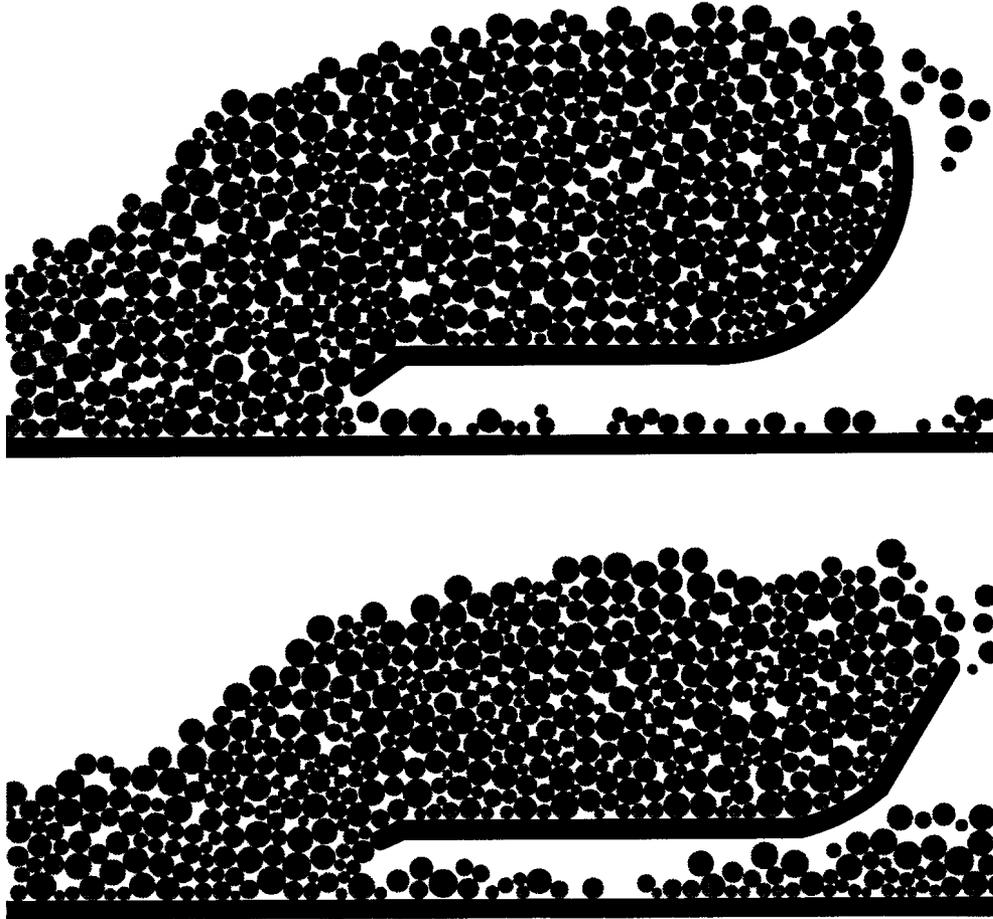- The lip lifts up the material. This causes the first vertical surge.

Figure 17: Final configuration of particles with a 3/1 size ratio in (a) bucket 1, (10 s), (b) bucket 2, (11 s).

- Material sometimes slowly slides along the bottom and up the back and is eventually swept off the back by material flowing over the top of the filled bucket. The bucket cannot supply an adequate resistive force S_r (shown in figure 4) to immobilise the spoil in the bucket. This will lead to wear on the bucket lining.

Features particular to bucket 2

- The initial material enters as a thin rolling layer and continues up the back and falls out of the bucket. This at first appears undesirable. However the material falling out the back acts to brake the material behind it in the bucket. This resistive force rapidly slows the rolling layer to a halt. The total amount of material that escapes is small.

- The filling is only a one stage surging process. Here the much shallower lip acts only to separate the incoming material into two streams. One passes into the bucket and one passes underneath. It does not provide a strong upward collisional force F_c (shown in figure 4).

- The back and bottom parts of this bucket are essentially straight and join smoothly. The resulting shape no longer has a constant radius of curvature. The uniform size material is unable to slide along the bottom and up the back to finally be lost over the back. For this material, this bucket provides enough resistance S_r in order to hold the material firmly. Once particles have come to rest in the bucket they do not move in response to the pressure exerted by the material pushed ahead of the bucket. For material with a size distribution some movement is observable.

Here we make no effort to decide which bucket has the best performance. This involves far too many factors for such a restricted study. What has been demonstrated is that different bucket designs do produce different material flow patterns and have different filling properties and that these *can* be identified and analysed numerically.

Simulations were also performed, for exactly the same bucket geometries, using non-uniform size particles. The particle diameters were uniformly distributed between 10 cm and 30 cm. This represents a size ratio of 3/1. The average particle size of 20 cm is the same as size of the uniform particles used earlier. The final configurations for both the buckets are shown in figure 17.

The main effects of the size variation are to increase the resistance of the granular material to shear and to reduce the speed of flow over the bucket (since the disordered microstructure causes more collisions and the flow is more dissipative). This affects both the flow pattern and the final spoil. In the uniform particle cases the material enters the bucket until the spoil level is above the level of the back of the bucket. All further material entering the bucket flows horizontally over the top of the previously deposited material and over the back. This prevents the material from piling above the level of the back. In the non-uniform cases, slow creeping motions are evident throughout the granular material in the bucket. The layers of particles above the level of the back feel more resistance and fewer particles flow over the back. This results in a higher final spoil level and, more importantly, the spoil loses its characteristic flat shape. It is now heaped towards the center of both buckets. It is expected that the inclusion of even larger particles will continue this trend. This emphasises the need to have and use an adequate characterisation of the particles in the spoil, in order to reproduce observed results.

As an indication of computational resources required for these simulations, the first bucket simulation ran for 40,000 time steps with 978 particles in only 75 min of CPU time on a Decstation 5000/200. Larger numbers of smaller particle, large

size distributions and many different configurations of the bucket can be performed in reasonable times on very modest computers.

7. Validation

In the above simulations and discussion we have shown flow patterns that are qualitatively similar to those observed. The filling times and surface profiles are in good agreement with observations.

Detailed quantitative validation is well beyond the scope of the Study Group The present purpose is to show that such modelling is possible rather than to present the *perfect* model. However for the work to proceed, rigorous qualitative validation is necessary.

- No attempt was made to integrate the drag forces on the bucket. These calculations are possible and would provide one point of validation. Power consumption could also be estimated.

- The University of Queensland has a two dimensional test rig. This could provide experimental flow patterns with which to compare the detailed structure of the computed flows in the bucket.

- The density of the material packed into the bucket varies throughout the bucket once it is filled. These variations can be up to 30% and affect the location of the center of mass. This has a major effect on the way the bucket is rigged. Predicting the density variations and matching experimental values is a crucial validation test. Correctly reproducing these variations is arguably the most important feature to reproduce, given the enormous importance it has on bucket performance.

8. Conclusions

Most importantly, we have demonstrated that particle methods can be used to at least qualitatively model dragline bucket filling. The particles do enter the bucket and their motion is broadly consistent with the available information. Further quantitative validation of the model is desirable, but is well beyond the scope of the Study Group.

The model predicts the motion of circular two dimensional particles with various size distributions. The boundary specifications are sufficiently flexible that any bucket shape can be tested.

The model *can* identify variations in the pattern of bucket filling resulting from changes in the bucket geometry. The bulk flows of material into each of the two buckets modelled is demonstrably different. We have not attempted to decide which is a better bucket design. This requires a far more thorough study. What has been demonstrated is that such a study is feasible.

A realistic model of a dragline bucket filling requires the addition of static friction, cohesion and dynamic moving boundaries. Progress has been made on all these areas. They are all either presently implemented or could be implemented in the future, if the requirement arose. Fully dynamic boundaries would enable the bucket to move and change orientation, in response to the complex balance of forces acting on it. This would enable detailed prediction of the bucket behaviour. Both the filling and emptying phases could be equally well modelled. Drag forces and power requirements could also be predicted.

Size variations in the spoil affect both its volume and its shape. Non-uniform particles have more resistance to shear forces. This causes a larger spoil volume and a more heaped shape in comparison to the almost flat ones obtained with uniform size particles.

In summary, such a model as described here could eventually, after further extension and validation, be used to:

- Evaluate different bucket designs and test their filling and emptying characteristics. This would enable predictions of filling times and spoil volumes to be made. Density variations and the center of mass could be determined. This would provide information that may be used to to help optimise the rigging used on the bucket.

- Explore wear patterns on the buckets with a view to extending the life span of the buckets and minimising the downtime of the draglines.

- Assist in making operational decisions – in helping to determine the best slope of the cut, positioning of the dragline and digging strategies. An understanding of the behaviour of the flow patterns of the rocks throughout the entire digging process can be used enhance operational decision making.

- Study the breaking and lifting of the rock by the teeth and lip. This would require rock fracturing to be included. A study of the lip and teeth interaction would necessarily be three dimensional. Both are possible, but are much more difficult.

Video animations of the simulations are available from the moderator.

Acknowledgements

References

M.A. Hopkins, "Particle Simulations: Volume 1", *Report No 87–7*, Clarkson University, Potsdam, NY, USA (1987).

C.S. Campbell, "Rapid Granular Flows", *Annual Rev. Fluid Mech.* **22** (1990), 57–92.

P.W. Cleary, "Granular Flows", *CSIRO 2nd Annual Post-doctoral report* (1991).

M.A. Hopkins, W.D. Hibler, and G.M. Flato, "On the numerical simulation of the sea ice ridging process", *J. Geophys. Research* (1991).

M.A. Hopkins, "The numerical simulation of systems of multitudinous polygonal blocks", US Army Cold Regions Research and Engineering Laboratory, *USACREEL Report CR 513* (1992).

A.V. Potapov, C.S. Campbell and M.A. Hopkins, "Computer Simulation of particle fracture", *USC Report IFPRI.4*, University of Southern California, USA (1992).

L. Rothenberg, and R.J. Bathurst, "Effects of particle shape on micromechanical behavior of granular materials" in *Advances in Micromechanics of Granular Materials* (Eds. H. Shen, M. Satake, M. Mehrabadi, C. Chang, and C. S. Campbell) Elsevier (1992), 343–352.