Sand Simulation - COMP768 project report

Abhinav Golas

May 07, 2009

Contents

1 Introduction

Granular materials are ubiquitous in our world, and we can see examples in everyday life. Examples of granular materials include grains, sand, pebbles among other things. Technically, the definition of granular materials is a conglomeration of discrete solid, macroscopic particles characterized by a loss of energy at every particle interaction. Such grains can be of any size, be it from small sizes like 1 μm to large ones like icebergs.

Sand belongs to a class of granular materials called powders, which have the special property of being more cohesive and able to be suspended in a gas, due to their small particles. The simulation of granular materials is of interest to engineers due to their varied applications in various industries, be it agriculture, pharmaceuticals to even energy production. In addition, they find application in numerous movies and games as part of the surrounding environment.

1.1 Physical properties

Sand, and other granular materials have properties which differ from other states of matter, i.e. they can't be exactly classified as either solids, liquids or gases. They can support small amounts of shear stresses indefinitely, an example of which can be seen in sand dunes and their like. Such a feature differs them from liquids and other fluids, as fluids deform under any amount of shear stress. Due to their ability to flow, they are definitely different from solids as well. Both of these properties combined can sometimes manifest as clogging in a constriction, as the normal stresses can make the material withstand large shear stresses. In addition, they are often inhomogeneous and anisotropic, and can often exhibit avalanches.

As shall be detailed later, such materials have this ability to withstand shear stresses due to the effect of friction, including interaction with other particles as well as any solid contact with the boundaries. Due to this, the hydrostatic pressure in granular materials tends to cap off rather than increase indefinitely, as the weight load is distributed to the surrounding walls due to static friction. This is of special interest to civil engineers involved in silo construction, as this friction can cause grain silos to collapse if not designed properly.

As mentioned earlier, granular materials are also characterized by a loss of energy at every particle interaction. Thus all particle collisions can be assumed to inelastic. Though this is not accurate, it is qualitatively similar, as due to the large number of particles, the number of particle interactions in finite time approach infinity. This leads to the accumulation of the loss of energy resulting in the in-elasticity assumption. This phenomenon is also known as inelastic collapse. [4] provides some useful examples of this, and other behavior described in this section.

2 Sand Simulation

This project deals with the simulation of sand, and to do so we must understand the problem at hand. Sand exists in various states of density, which can affect its behavior. The most common form in which sand exists is the medium-to-high density sand which can flow, and also stand. In this form, some parts of sand actually behave as solids, so even if any flow exists, it will not propagate to the bulk of the material. Rather it will remain restricted to the boundaries. The other common form is as a suspension in a gas. This is naturally observed as sandstorms.

Any simulation methods designed have a natural extension to granular materials. For example, situations like avalanches and landslides can also be simulated to techniques very similar to the methods described later.

2.1 Stress

To understand the behavior of sand, it is essential to understand the underlying physical quantities. The first such physical quantity is stress. Stress is defined as the force acting on a surface per unit area, i.e. $\sigma = F/A$. In 3-D space, this can be naturally expressed as 2nd order tensor:

$$\sigma = \begin{pmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{xy} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{xz} & \sigma_{yz} & \sigma_{zz} \end{pmatrix} \tag{1}$$

Though a general 2nd order tensor would have 9 independent components, under equilibrium conditions, the stress tensor has only 6 independent components as shown above, which results in a symmetric matrix. This definition of stress is clearly dependent on the definition of a basis x-y-z in 3-D space, and so we can have

an infinite number of stress tensors for any point in space. To talk about stress, certain invariants need to be defined, so behavior can be defined in a manner independent of the frame of reference.

The first and most natural way to do so, would be to use the eigendecomposition of this matrix. The resulting 3 eigenvalues, σ_1, σ_2, and σ_3 are called the principal eigenvalues. Another way is to define stress invariants. In fact, hydrostatic pressure is an example of a stress invariant, being equal to 1/3 times the trace of stress matrix. Thus in stress notation it is also known as I_1, the first stress invariant. Similarly, 2 other stress invariants can also be described.

Another alternate set of invariants(J_1, J_2) can defined on $\sigma - I_1\delta$, which are known as the deviatoric stress invariants. We shall be using the invariant J_1 in later description.

2.2 Yield Condition

The yield condition describes whether a materials will yield under the application of stress. It is specific to materials, and many different yield conditions have been proposed for different materials. The one relevant to granular materials is the Mohr-Coulomb yield condition, and it is defined using I_1 (also written as mean stress or σ_m), i.e. our hydrostatic pressure, and an invariant called the Tresca equivalent stress. However, Tresca equivalent stress has smoothness issues in simulation, and thus a J_2 (or $\bar{\sigma}$), i.e. our 2nd deviatoric stress, also known as Von Mises equivalent stress can be used. It is defined as:

$$J_2 = \frac{\|\sigma - I_1\|_F}{\sqrt{2}} \tag{2}$$

The yield condition, or the no-slip condition can then be defined as:

$$\sqrt{3}\bar{\sigma} < \sigma_m sin\Phi \tag{3}$$

where, $sin\Phi$ is the coefficient of friction, defined as the yield angle. Thus, any point lying inside the yield surface by the above equation will not yield to that stress value, remaining rigid. It is important to note, that the Frobenius norm is used to define the equivalent stress, and as such is a non-linear function.

2.3 Simulation complexity

Any scene involving sand simulation contains of the order of tens of millions to even billions and trillions of sand grains. Thus any simulation method must scale well to such a large scale. The scale of the problem also throws up a representational issue, i.e. whether sand in the simulation should be represented as a continuum, or as discrete particles. As we shall see later, both approaches have been tried out in existing approaches. In addition, due to the non-linearity of the yield condition, it is difficult to define an efficient numerical method, as most methods exist for linear problems.

3 State-of-the-Art

In this section we give a brief overview of existing work on sand simulation. We encourage the reader to refer to the source material for further details.

3.1 Particle-Based simulation of Granular Materials, Bell et al. [2]

This paper approaches the granular material simulation problem from the discrete domain. Every grain is constructed from spherical primitive particles, and a simplified rigid body simulation is run on the set of

3

spheres. For defining rigid body interaction, the paper uses Molecular dynamics, in which every interacting/colliding particle pair is allowed to intersect, and the depth of intersection is used to define the interacting forces. The paper then details the normal and tangential forces that act on each particle pair.

Since no amount of tangential forces can keep to spherical particles in an unstable equilibrium, there is a need for forces that will keep these particles at rest. Without such forces, we would be unable to observe something as simple as sand piles. To do so, instead of using spring forces between interacting particles, the paper proposes the construction of grains as tetrahedral arrangements of spheres. The normal forces arising from the interaction of such grains would result in the desired stability in sand piles.

In order for the system to interact with rigid bodies, the paper decomposes rigid bodies into similar conglomerations of spheres by sampling sphere on the surface of the mesh, or at a certain distance away from the mesh by using a distance field. To improve the efficiency of collision detection, the proposed method uses uniform grids to bin each object. Since particles move under adaptive timesteps restricting them to move fractions of grid cell dimensions at each timesteps, every grid cell contains links to every particle lying in it, and those lying in the 26 neighboring cells. This results in lesser number of updates to the grid, and move memory coherency when retrieving these lists from each cell. For such simulations where stagnant regions abound, this is a massive gain in efficiency.

The paper has an advantage of being accurate and faithful to all aspects of the behavior of granular materials. However, it suffers from issues of scale, being able to handle scenes of the order or 100k spheres, and thus not suited to large scale sand simulation.

With this in mind, we look at the continuum approach to the problem.

3.2 Animating Sand as a Fluid, Zhu et al. [5]

This paper attempts to simulate sand as an extension of an existing fluid simulator. It attempts to add on the additional behavior as a "bootstrap" addition to grid based fluid solvers. Thus after doing the regular fluid simulation steps of advection, body forces, and pressure solve, it evaluates the effect of friction on sand. To do so, it first estimates the stress of the material in 2 cases, whether the sand is acting as a fluid, or solid. This is done as a function of the deformation rate or strain rate $D = [\nabla u + \nabla u^T]$ The solid stress is used to check whether the cell satisfies the yield condition or not. If it does not, the fluid stress is used to modify the grid velocity by the frictional force. If it does, then the velocities of such rigid cells are projected to the space of rigid body motions, i.e. $D = 0$. This is done by an approach outlined in [3].

This results is relatively accurate behavior and can handle most cases of sand simulation. However, it still simulates sand as a perfect fluid, and thus does not simulate the pressure capping effect of sand. As a result it cannot simulate examples such as a simple hourglass, which relies on this property. Still, the method is extremely robust and scales very well to large scale scenes, as it is independent of the number of grains in the scene.

I wish to propose a method to improve this solution method, as the method to add the effect of friction is bootstrapped and not an integral part of the simulation. At the least, an iterative solution is necessary to properly map the frictional behavior of sand.

4 Proposed Method and Progress

4.1 Proposed method

As noted in the paper [5], the yield condition must be checked for different continuum behavior to be applied. However, the approach presented in the paper applied only one iteration of an optimization. In actuality,

application of frictional stress would destroy the divergence-free property of the velocity field. As is apparent, we have multiple constraints which must simultaneously satisfied. The constraints are:

- Incompressibility: $\nabla.v = 0$ or Unilateral incompressibility: $\rho < \rho_{max} \perp p > 0$

- Boundary conditions: $u.n \geq 0$

- Yield condition: $\sqrt{3}\bar{\sigma} < \sigma_m sin\Phi$

The first 2 conditions are linear, but the 3rd is not. But linearized the yield condition is simple, the only thing making it non-linear is the norm used to define the equivalent stress$\bar{\sigma}$, which can simply be converted to a 1-norm or the infinity norm.

Then we rely on the initial observation of granular materials, that all interactions are inelastic. Thus, from the observation made in [1], the incompressibility solution can be expressed as a minimization of kinetic energy. To clarify, since interactions are inelastic, any force, in this case stress, will attempt to minimize the total energy of the system, which here is the kinetic energy.

Thus, our problem can be expressed as a Quadratic minimization problem with linear constraints. However, methods for solution of general quadratic problems are slow, as noted in [1], where the KKT based PATH solver was used. Instead, we can attempt to solve the problem in an iterative fashion, where we solve for each of the constraints independently using different methods, and recurse. In such a method, a LCP (Linear Complementarity Problem) solver can be used for constraints 1 and 2, while we can apply constraint 3 as a post process and iterate. The overall problem is not an LCP problem as the frictional yield constraints are not orthogonal to the other constraints, as is required for the LCP formulation.

4.2 Progress Made

For this approach to be tested out, I started out trying to develop a 3-D fluid simulator. I used the variational approach outlined in [1] as a base model, i.e. instead of using a conventional incompressibility solve formulation, I used a variational formulation based on kinetic energy formulation. This model allows for better solid-fluid interaction, and it can handle sub-grid interactions, which can result in stair-case artifacts in conventional incompressibility solvers.

4.2.1 Coding effort

I reused the earlier implemented particle system for use in the this solver. In addition, I used a Conjugate Gradient solver developed by Bridson et. al. for solving sparse linear systems.

Apart from that, the system was developed from scratch. The following list outlines the overall coding effort made:

- 3D grid based simulator developed

- Uses variational incompressibility solver, and grid based advection

- Rigidity projection: for projecting velocity fields of rigid components to the space of rigid body motions

  - All rigid body motions must be of the form $v_p = v_C + \omega \times (r_p - r_C)$, where C is the center of mass
  - Accumulate mass(M), momentum(M_v) and angular momentum(M_ω) for each set of connected rigid cells

- Determine v_C and ω as $Mv_C = M_v$, $I\omega = M_\omega$. The inertia tensor can be constructed as well in the previous step

- Particle system coupled with the grid simulator

  - Density field built each frame from particle distribution using linear interpolation weights
  - Particles are tracers, i.e. they get velocity from grid

- Other minor components like code to check the yield condition after evaluating stress, rendering etc.

4.2.2 Results

I developed a simple example of falling sand to demonstrate the simulator. It demonstrates 10k grains of loose sand falling under the action of gravity. Notice how even the sand eventually ends up spread on a larger area on the floor than the cross-sectional area of the initial pile. The figures below show some samples of the example: The example runs in real time with 10k particles, on a 10x10x10 grid.
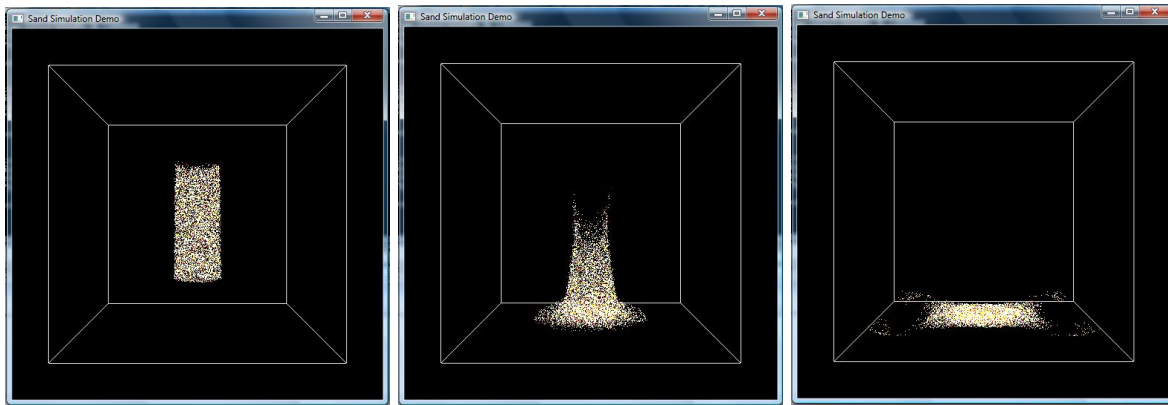

Figure 1: Falling sand example

4.3 Future Work and Conclusion

Due to problems faced in the theoretical formulation and the implementation of a fluid simulator, I was not able to complete all of my set goals for the project. I have the following targets for the near future:

- Remove any remaining bugs in the simulator

- Add an LCP solver to test out the constraints listed in previous sections

Over a longer time frame, I aim to develop a better numerical solution method for sand, which would solve the minimization satisfying all of the constraints. This includes testing with a KKT solver for the quadratic problem, and attempting an LCP formulation.

In addition, I came across an interesting problem while trying to implement the solver. In involves trying to calculate the fluid mass in any grid cell, which is essentially the fraction of the cell occupied by fluid. To do so, we need to find the volume of the tetrahedron generated by the intersection of arbitrary tetrahedra with this cuboidal grid cell. I tried searching for possible efficient solutions to this problem, all to no avail, so I aim to try to find an efficient method to do this, which would let my solver interact with arbitrary solids.

6

References

[1] BATTY, C., BERTAILS, F., AND BRIDSON, R. A fast variational framework for accurate solid-fluid coupling. *ACM Trans. Graph. 26*, 3 (2007), 100.

[2] BELL, N., YU, Y., AND MUCHA, P. J. Particle-based simulation of granular materials. In *SCA '05: Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation* (New York, NY, USA, 2005), ACM, pp. 77–86.

[3] CARLSON, M., MUCHA, P. J., AND TURK, G. Rigid fluid: animating the interplay between rigid bodies and fluid. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Papers* (New York, NY, USA, 2004), ACM, pp. 377–384.

[4] JAEGER, H. M., AND NAGEL, S. R. Granular solids, liquids, and gases. *Reviews of Modern Physics* (1996).

[5] ZHU, Y., AND BRIDSON, R. Animating sand as a fluid. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Papers* (New York, NY, USA, 2005), ACM, pp. 965–972.