Modeling of Granular Materials

Abhinav Golas

COMP 768 - Physically Based Simulation

Motivation

- Movies, games

Spiderman 3

The Mummy

- Engineering design – grain silos
- Avalanches, Landslides

www.stheoutlawtorn.com

Overview

- What are Granular Materials?
- Simulation
- Rendering

Overview

- What are Granular Materials?
- Simulation
- Rendering

What are Granular Materials?

- A granular material is a conglomeration of discrete solid, macroscopic particles characterized by a loss of energy whenever the particles interact (Wikipedia)
- Size variation from 1µm to icebergs
- Food grains, sand, coal etc.
- Powders – can be suspended in gas

What are Granular materials?

- Can exist similar to various forms of matter
  - Gas/Liquid – powders can be carried by velocity fields
    - Sandstorms
  - Liquid/Solid – similar to liquids embedded with multiple solid objects
    - Avalanches, landslides
    - Hourglass
- Similar to viscous liquids

Why the separate classification?

- Behavior not consistent with any one state of matter

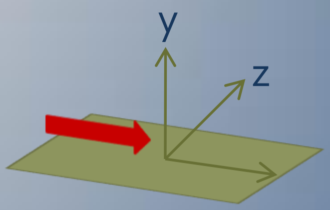
1. Can sustain small shear stresses – stable piles
    - Hydrostatic pressure achieves a maximum
2. Particle interactions lose energy
    - Collisions approach inelastic
    - Infinite collisions in finite time – inelastic collapse
3. Inhomogeneous and anisotropic
    - Particle shape and size inhomogeneous

Granular solids, liquids, and gases – Jaeger et al.

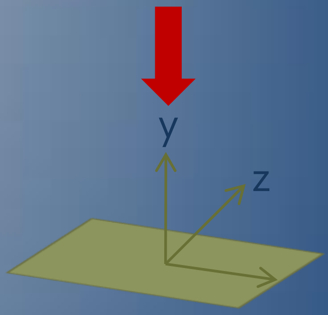
Understanding the behavior - Stress

- Stress $\sigma = \frac{F}{A}$

$$\sigma = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zy} & \sigma_{zz} \end{bmatrix}$$



Shear



Normal

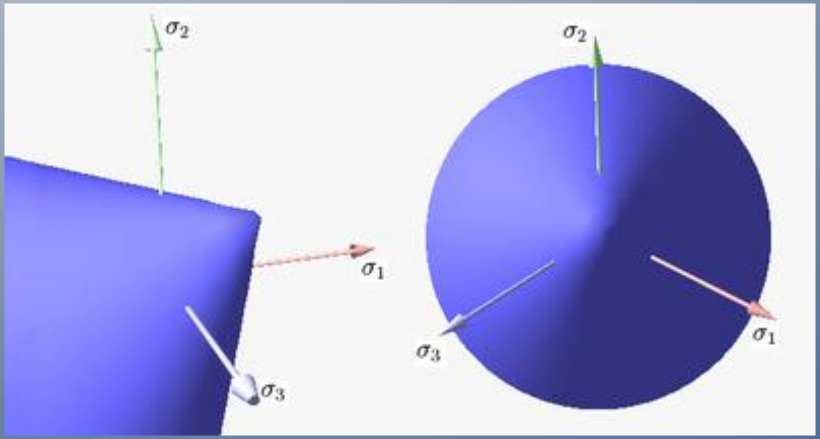
- At equilibrium – matrix is symmetric – 6 degrees of freedom
- Pressure for fluids – tr(σ)/ 3

Stress

- Different matrix for different basis – need invariants
  - Pressure! – I_0
  - Deviatoric invariants – Invariants based on $\sigma - I_0\delta$ $\Longrightarrow J_1, J_2$
- Eigen values? – called principle stresses

Understanding the behavior

- Why can sand sustain shear stress?
  - Friction between particles
- When does it yield? – yield surface/condition



Yield surface

- Many surfaces – suitable for different materials
- Mohr Coulomb surface with Von-Mises equivalent stress – f(I_0, J_1)

$$I_0 = \sigma_m = \frac{tr(\sigma)}{3} \qquad J_1 = \bar{\sigma} = \frac{\left\| \sigma - \sigma_m \delta \right\|_F}{\sqrt{2}}$$

- Condition for stability/rigidity:

$$\sqrt{3} J_1 < I_0 \sin \Phi$$

- sinΦ – coefficient of friction

So why is it difficult to simulate?

- Scale - >10M particles
- Nonlinear behavior – yield surface
- Representation – discrete or continuum?

Overview

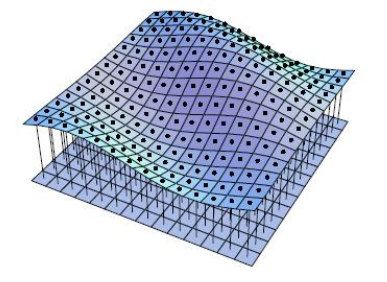
- What are Granular Materials?
- Simulation
- Rendering

Simulation

- Depends on what scenario to simulate
  - 2 dimensional – Animating Sand, Mud, and Snow, Sumner et al.
  - Discrete particles – Particle-Based Simulation of Granular Materials, Bell et al.
  - Continuum – Animating Sand as a Fluid, Zhu et al.

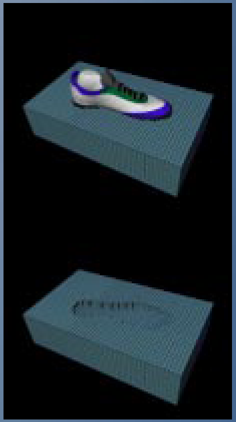
Animating Sand, Mud, and Snow

- Model deformations on a 2D height field surface caused by rigid bodies
  - Hash based grid – space saving

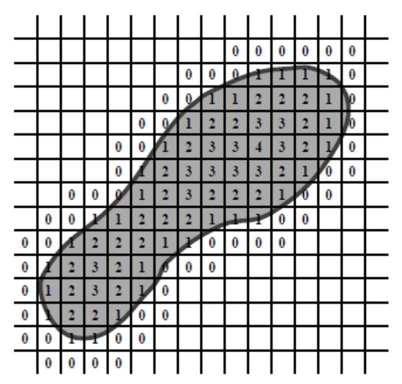


- Model features
  - Material redistribution, compression
  - Particles that get stuck to rigid body

Animating Sand, Mud, and Snow



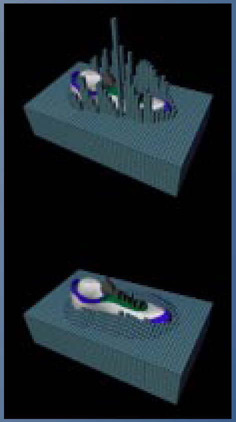
- Rigid body intersection check – ray casting
- Extra material
  - Displaced
  - Compressed
- Transferring extra material
  - Construct distance field to nearest clear cell
  - Transfer material to that cell



									0	0	0	0	0	0
							0	0	0	1	1	1	1	0
						0	0	1	1	2	2	2	1	0
					0	0	1	2	2	3	3	2	1	0
				0	0	1	2	3	3	4	3	2	1	0
				0	1	2	3	3	3	3	2	1	0	0
		0	0	0	1	2	3	2	2	2	1	0	0	
	0	0	1	1	2	2	2	1	1	1	0	0		
0	0	1	2	2	2	1	1	0	0	0	0			
0	1	2	3	2	1	0	0	0						
0	1	2	3	2	1	0								
0	1	2	2	1	0	0								
0	0	1	1	0	0									
	0	0	0	0										

Redistribution of material

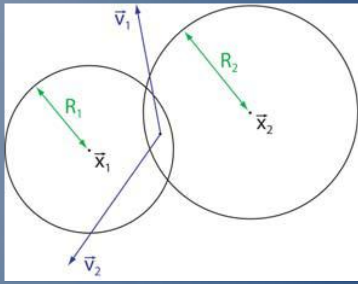
- Erosion
  - Distribute material equally to all neighboring lower height cells – if slope > threshold
- Particle Generation
  - Material may get stuck to bottom of body
  - Seed particle system from each contact triangle on rigid body – volume = c * area of triangle
    - Volume lost per time step – exponential decay



Particle-Based Simulation of Granular Materials

- Use a particle system with collision handling
- Define objects in terms of spheres
  - Need to define per sphere pair interaction forces
- Collision system based on Molecular Dynamics
  - Allow minor spatial overlap between objects

Sphere pair interaction



- Define overlap(ξ), relative velocity(V), contact normal(N), normal and tangential velocities(V_n, V_t), rate of change of overlap(V.N)
- Normal forces

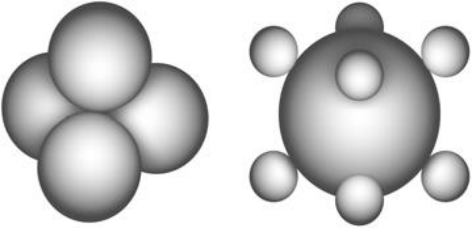
$$\vec{F}_n = f_n \vec{N} \qquad f_n + k_d \xi^{\alpha} \dot{\xi} + k_r \xi^{\beta} = 0$$

- k_d : dissipation during collisions, k_r : particle stiffness
- Best choice of coefficients: α=1/2, β=3/2
- Given coefficient of restitution ε, and time of contact t_c, we can determine k_d and k_r

Sphere pair interaction

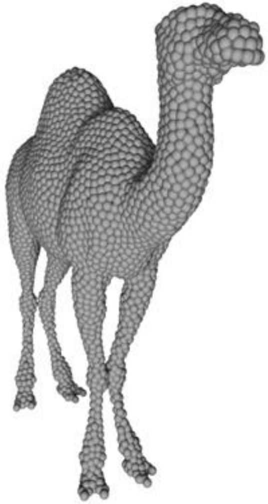
- Tangential forces

$$\vec{F}_t = -\min\left(\mu f_n, k_t \left\|\vec{V}_t\right\|\right)\frac{\vec{V}_t}{\left\|\vec{V}_t\right\|}$$



- These forces cannot stop motion – require true static friction
  - Springs between particles with persistent contact?
  - Non-spherical objects

Solid bodies



- Map mesh to structure built from spheres
  - Generate distance field from mesh
  - Choose offset from mesh to place spheres
  - Build iso-surface mesh (Marching Tetrahedra)
  - Sample spheres randomly on triangles
  - Let them float to desired iso-surface by repulsion forces
- D=sphere density, A=triangle area, R=particle radius, place $\left\lfloor \frac{DA}{\pi r^2} \right\rfloor$ particles, 1 more with fractional probability

Solid bodies

$$\vec{V} = \vec{V}_F + \vec{V}_R \qquad V_F = -\Phi\left(\vec{P}\right)\vec{N} \qquad \vec{V}_R = \sum_{P_i \in S \setminus P} K\left(\left\|\vec{P} - \vec{P}_i\right\|\right)\frac{\vec{P} - \vec{P}_i}{\left\|\vec{P} - \vec{P}_i\right\|}$$

- K – interaction kernel, P – Position of particle, V – velocity of particle, Φ – distance field
- Rigid body evolution
  - Overall force = Σ forces
  - Overall torque = Σ torques around center of mass

Efficient collision detection

- Spatial hashing
  - Grid size = 2 x Maximum particle radius
- Need to look at 27 cells for each particle ⟹O(n)
- Not good enough, insert each particle into not 1, but 27 cells ⟹check only one cell for possible collisions
- Why better?
  - Spatial coherence
  - Particles moving to next grid cell, rare (inelastic collapse)
  - Wonderful for stagnant regions

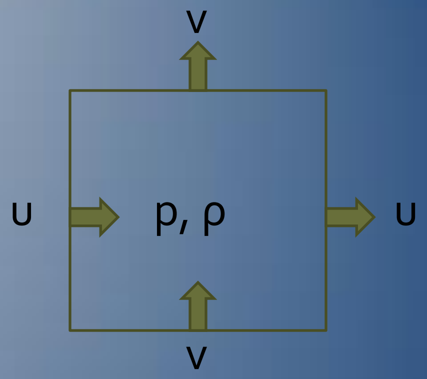
Advantages/Disadvantages

- The Good
  - Faithful to actual physical behavior
- The Bad and the Ugly
  - Computationally intensive
    - Small scale scenes
    - Scenes with some “control” particles

Animating Sand as a Fluid

- Motivation
  - Sand ~ viscous fluids in some cases
- Continuum simulation
- Bootstrap additions to existing fluid simulator
- Why?
  - Simulation independent of number of particles
  - Better numerical stability than rigid body simulators

Fluid simulation? what’s that?



- Discretize 3D region into cuboidal grid
- 3 step process to solve Navier Stokes equations
  - Advect
  - Add body forces
  - Incompressibility projection
- Stable and accurate under CFL condition

Extending our fluid simulator

- Extra things we need for sand
  - Friction (internal, boundary)
  - Rigid portions in sand
- Recall
  - Stress
  - Yield condition

Calculating stress

- Exact calculation infeasible
- Smart approximations
- Define strain rate – D = d/dt(strain)

$$D = \left(\nabla u + \nabla u^T\right) \qquad D_{i,j} = \frac{1}{2}\left(\frac{\partial u_i}{\partial j} + \frac{\partial u_j}{\partial i}\right)$$

- Approximate stresses
  - Rigid $\sigma_{rigid} = -\frac{\rho D \Delta x^2}{\Delta t}$
  - Fluid $\sigma_f = -p \sin\phi \frac{D}{\sqrt{1/3}|D|_F}$

The algorithm in a nutshell

- Calculate strain rate
- Find rigid stress for cell
- Cell satisfies yield condition?
  - Yes – mark rigid, store rigid stress
  - No – mark fluid, store fluid stress
- For each rigid connected component
  - Accumulate forces and torques
- For fluid cells, subtract friction force

Yield condition

- Recap

$$I_0 = \sigma_m = \frac{tr(\sigma)}{3} \qquad J_1 = \bar{\sigma} = \frac{\|\sigma - \sigma_m \delta\|_F}{\sqrt{2}}$$

$$\sqrt{3} J_1 < I_0 \sin \Phi$$

- Can add a cohesive force for sticky materials

$$\sqrt{3} J_1 < I_0 \sin \Phi + c$$

Rigid components

- All velocities must lie in allowed space of rigid motion (D=0)
- Find connected components – graph search
- Accumulate momentum and angular momentum

$$M_i \hat{v}_i = \int_{R_i} \rho_i \hat{u} dV_i \qquad I_i \omega_i = \int_{R_i} r_i \times \rho_i \hat{u} dV_i$$

- R_i – solid region, u – velocity, ρ – density, I – moment of inertia

Rigid Fluid: Animating the Interplay Between Rigid Bodies and Fluid, Carlson et al.

Friction in fluid cells

- Update cell velocity $u += \Delta t / \rho \nabla \cdot \sigma_f$
- Boundary conditions
  - Normal velocity: $u \cdot n \geq 0$
  - Tangential velocity: $u_T = \max\left(0, 1 - \frac{\mu |u \cdot n|}{|u_T|}\right) u_T$

Representation

- Defining regions of sand
  - Level sets
  - Particles
    - Allow improved advection
    - Hybrid simulation
      - PIC – Particle In Cell
      - FLIP – FLuid Implicit Particle

Advection

- Semi – Lagrangian advection
  - Dissipative
  - Relies on incompressibility – volume conservation
- Hybrid approach
  - Use grids coupled with particles
  - Advect particles – no averaging losses!

PIC and FLIP methods

- PIC – Particle in cell
  - Particles support grid
  - Particles take velocity from grid
- FLIP – Fluid Implicit Particle
  - Grid supports particles
  - Particles take acceleration from grid

PIC and FLIP methods

- No grid based advection
  - Lesser dissipation
  - Particle advection is simpler
- No need for a level set
- PIC, more dissipative – suited for viscous flows, FLIP for inviscid flows
- Surface reconstruction?

Surface reconstruction

- Surface Φ(x) – define using all particles i

$$\Phi(x) = |x - \bar{x}| - \bar{r} \qquad \bar{x} = \sum_i w_i x_i \qquad \bar{r} = \sum_i w_i r_i$$

$$w_i = \frac{k(|x - x_i| / R)}{\sum_j k(|x - x_j| / R)}$$

- k(s) – kernel function, R – 2 x average particle spacing, x – position, r – radius, w_i – weight for particle
- Suitable choice of kernel?
  - Must provide flat surfaces for what should be flat
  - $k(s) = \max(0, (1-s^2)^3)$

Surface reconstruction

- Issues
  - Concave regions – centre might lie outside region
    - Smoothing pass
  - Radii must be close approximation to distance from surface
    - Non-trivial, constant particle radius assumed
- Advantages
  - Fast
  - No temporal interdependence

Advantages/Disadvantages

- Advantages
  - Fast & stable
  - Independent of number of particles – large scale scenes possible
- Disadvantages
  - Not completely true to actual behavior
  - Detail issues – smoothing in simulation, surface reconstruction

Overview

- What are Granular Materials?
- Simulation
- Rendering

Rendering

- Non-trivial due to scale and visual complexity
- Surface based rendering
  - Use volumetric textures
  - Texture advected by fluid velocity
- Particle rendering

Particle Rendering

- Level of detail necessary

Rendering Tons of Sand, Sony Pictures Imageworks

- Different approaches for different levels

Particle Rendering

- Sand clouds
  - Light Reflection Functions for Simulation of Clouds and Dusty Surfaces, Blinn
  - Defines lighting and scattering functions for such materials
  - Suitable options for dust, clouds

Rendering Tons of Sand

- Surfaces with sand
  - Generate particles on mesh at runtime with temporal coherence
- Sand particles
  - Generate required number for visual detail from base “control” particles
- Rendering level of detail

Polygon mesh → Camera aligned patches → Point clusters → Single points

- Peak of 480 million particles at render time

Conclusions

- Interesting, albeit difficult problem
- Models not perfect
  - Speed vs. scale/realism tradeoff
- Similar tradeoff in rendering

References

1. Granular Solids, Liquids, and Gases – Jaeger et al.
   Review of Modern Physics ’96
2. Particle-Based Simulations of Granular Materials – Bell et al., Eurographics ‘05
3. Animating Sand as a Fluid – Zhu et al.
   SIGGRAPH ‘05
4. Rigid Fluid: Animating the Interplay Between Rigid Bodies and Fluid – Carlson et al.
   SIGGRAPH ’04

References

5. Rendering Tons of Sand – Allen et al. Sony Pictures Imageworks
   SIGGRAPH 2007 sketches
6. A Fast Variational Framework for Accurate Solid-Fluid Coupling – Batty et al.
   SIGGRAPH 2007
7. Animating Sand, Mud, and Snow – Sumner et al.
   Eurographics 1999