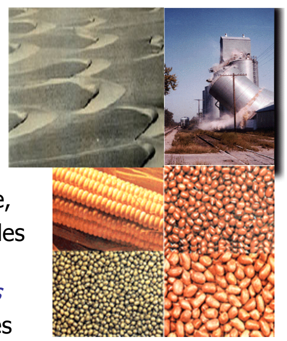
Particle Systems for Beginnners

Stefan Luding
MSM, TS, CTW, UTwente, NL

Stefan Luding, s.luding@utwente.nl
MSM, TS, CTW, UTwente, NL

Granular Materials

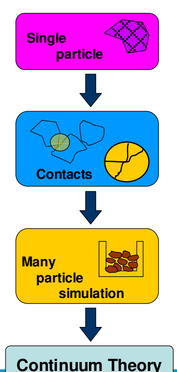


Real:
- sand, soil, rock,
- grain, rice, lentils,
- powder, pills, granulate,
- micro- and nano-particles

Model Granular Materials
- steel/aluminum spheres
- spheres with dissipation/friction/adhesion

Approach philosophy

- Introduction
- Single Particles
- Particle Contacts/Interactions
- Many particle cooperative behavior
- Applications/Examples
- Conclusion



Deterministic Models …

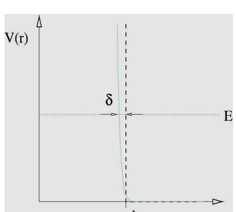
Method	Abbrev.	Theory
Molecular dynamics (soft particles)	MD	…
Event Driven (hard particles)	ED	(Kinetic Theory)
Monte Carlo (random motion)	MC	Stat. Phys.
Direct Simulation Monte Carlo	DSMC	Kinetic Theory
Lattice (Boltzmann) Models	LB	Navier Stokes

DCCSE – steps in simulation

1. Setting up a model	1. Particle model
2. Analytical treatment	2. Kinetic theory
3. Numerical treatment	3. Algorithms for MD
4. Implementation	4. FORTRAN or C++/MPI
5. Embedding	5. Linux – research codes
6. Visualisation	6. xballs X11 C-tool
7. Validation	7. theory/experiment

What is Molecular Dynamics ?

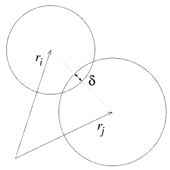
1. Specify interactions between bodies (for example: two spherical atoms)



2. Compute all forces $\mathbf{f}_{j\to i}$
3. Integrate the equations of motion for all particles (Verlet, Runge-Kutta, Predictor-Corrector, …) with fixed time-step dt

$$m\ddot{\mathbf{x}}_i = \sum_{j\neq i} \mathbf{f}_{j\to i}$$

Discrete particle model

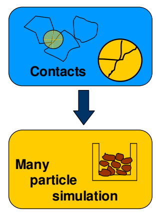


Equations of motion

$$m_i \frac{d^2 \vec{r}_i}{dt^2} = \vec{f}_i$$

Forces and torques:

$$\vec{f}_i = \sum_c \vec{f}_i^{\,c} + \sum_w \vec{f}_i^{\,w} + m_i g$$

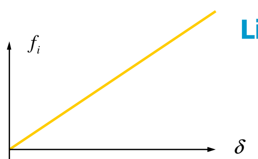


Contact if Overlap > 0

Overlap $\delta = \frac{1}{2}(d_i + d_j) - (\vec{r}_i - \vec{r}_j)\cdot\vec{n}$

Normal $\hat{n} = \vec{n}_{ij} = \frac{(\vec{r}_i - \vec{r}_j)}{|\vec{r}_i - \vec{r}_j|}$

Linear Contact model



- really simple ☺
- linear, analytical
- very easy to implement

$$f_i = -m_{ij}\ddot{\delta} = k\delta + \gamma\dot{\delta}$$

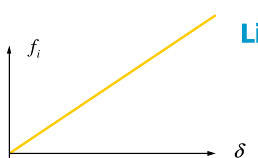
overlap $\delta = \frac{1}{2}(d_i + d_j) - (\vec{r}_i - \vec{r}_j)\cdot\vec{n}$

rel. velocity $\dot{\delta} = -(\vec{v}_i - \vec{v}_j)\cdot\vec{n}$

acceleration $\ddot{\delta} = -(\vec{a}_i - \vec{a}_j)\cdot\vec{n}$

http://www.ical.uni-stuttgart.de/~lui/PAPERS/coll2p.pdf

Linear Contact model



- really simple ☺
- linear, analytical
- very easy to implement

$$f_i = -m_{ij}\ddot{\delta} = k\delta + \gamma\dot{\delta}$$

overlap $\delta = \frac{1}{2}(d_i + d_j) - (\vec{r}_i - \vec{r}_j)\cdot\vec{n}$

rel. velocity $\dot{\delta} = -(\vec{v}_i - \vec{v}_j)\cdot\vec{n}$

acceleration $\ddot{\delta} = -(\vec{a}_i - \vec{a}_j)\cdot\vec{n} = -\left(\vec{f}_i/m_i - \vec{f}_j/m_j\right)\cdot\vec{n} \overset{(\vec{f}_j = -\vec{f}_i)}{=} -\frac{1}{m_{ij}}\vec{f}_i\cdot\vec{n}$

http://www.ical.uni-stuttgart.de/~lui/PAPERS/coll2p.pdf

Linear Contact model

$$f_i = -m_{ij}\ddot{\delta} = k\delta + \gamma\dot{\delta}$$
$$k\delta + \gamma\dot{\delta} + m_{ij}\ddot{\delta} = 0$$
$$\frac{k}{m_{ij}}\delta + 2\frac{\gamma}{2m_{ij}}\dot{\delta} + \ddot{\delta} = 0$$
$$\boxed{\omega_0^2\delta + 2\eta\dot{\delta} + \ddot{\delta} = 0}$$

- really simple ☺
- linear, analytical
- very easy to implement

elastic freq. $\omega_0 = \sqrt{k/m_{ij}}$

eigen-freq. $\omega = \sqrt{\omega_0^2 - \eta^2}$

visc. diss. $\eta = \frac{\gamma}{2m_{ij}}$

http://www.ical.uni-stuttgart.de/~lui/PAPERS/coll2p.pdf

Linear Contact model

$$f_i = -m_{ij}\ddot{\delta} = k\delta + \gamma\dot{\delta}$$
$$k\delta + \gamma\dot{\delta} + m_{ij}\ddot{\delta} = 0$$
$$\frac{k}{m_{ij}}\delta + 2\frac{\gamma}{2m_{ij}}\dot{\delta} + \ddot{\delta} = 0$$
$$\boxed{\omega_0^2\delta + 2\eta\dot{\delta} + \ddot{\delta} = 0}$$

- really simple ☺
- linear, analytical
- very easy to implement

$$\delta(t) = \frac{v_0}{\omega}\exp(-\eta t)\sin(\omega t)$$
$$\dot{\delta}(t) = \frac{v_0}{\omega}\exp(-\eta t)\left[-\eta\sin(\omega t) + \omega\cos(\omega t)\right]$$

contact duration $\boxed{t_c = \pi/\omega}$

restitution coefficient $r = -\frac{v(t_c)}{v_0} = \exp(-\eta t_c)$

elastic freq. $\omega_0 = \sqrt{k/m_{ij}}$

eigen-freq. $\omega = \sqrt{\omega_0^2 - \eta^2}$

visc. diss. $\eta = \frac{\gamma}{2m_{ij}}$

http://www.ical.uni-stuttgart.de/~lui/PAPERS/coll2p.pdf

Linear Contact model ($m_w = \infty$)

	particle-particle	particle-wall
elastic freq.	$\omega_0 = \sqrt{k/m_{ij}}$	$\omega_0^{wall} = \sqrt{k/m_i} = \omega_0/\sqrt{2}$
eigen-freq.	$\omega = \sqrt{\omega_0^2 - \eta^2}$	$\omega^{wall} = \sqrt{\omega_0^2/2 - \eta^2/4}$
visc. diss.	$\eta = \frac{\gamma}{2m_{ij}}$	$\eta^{wall} = \frac{\gamma}{2m_i} = \frac{\eta}{2}$
contact duration	$\boxed{t_c = \pi/\omega}$	$\boxed{t_c^{wall} = \pi/\omega^{wall} > t_c}$
restitution coeff.	$r = \exp(-\eta t_c)$	$r^{wall} = \exp(-\eta^{wall} t_c^{wall})$

http://www.ical.uni-stuttgart.de/~lui/PAPERS/coll2p.pdf

Time-scales

time-step $\Delta t <= {}^{t_c}\!/_{50}$

$t_n < t_c$

contact duration $\boxed{t_c = {}^{\pi}\!/_{\omega}}$ $t_c^{wall} = {}^{\pi}\!/_{\omega^{wall}} > t_c$

time between contacts

$t_n > t_c$

sound propagation $N_L t_c$ ... with number of layers N_L

experiment T

http://www.ical.uni-stuttgart.de/~lui/PAPERS/coll2p.pdf

Time-scales

time-step $\Delta t <= {}^{t_c}\!/_{50}$

$t_n < t_c$

different sized particles

contact duration $\boxed{t_c = {}^{\pi}\!/_{\omega}}$ $t_c^{large} > t_c^{small}$

time between contacts

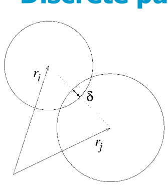
$t_n > t_c$

sound propagation $N_L t_c$ ... with number of layers N_L

experiment T

http://www.ical.uni-stuttgart.de/~lui/PAPERS/coll2p.pdf

Discrete particle model

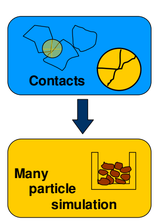


Equations of motion

$$m_i \frac{d^2 \vec{r}_i}{dt^2} = \vec{f}_i$$

Forces and torques:

$$\vec{f}_i = \sum_c \vec{f}_i^{\,c} + \sum_w \vec{f}_i^{\,w} + m_i g$$

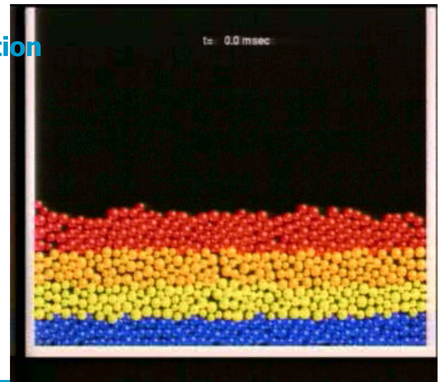


Contact if Overlap > 0

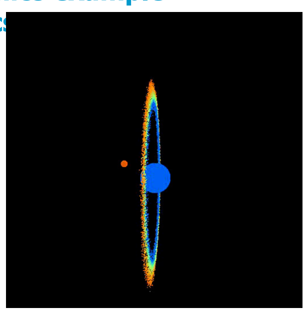
Overlap $\delta = \frac{1}{2}\left(d_i + d_j\right) - \left(\vec{r}_i - \vec{r}_j\right) \cdot \vec{n}$

Normal $\hat{n} = \vec{n}_{ij} = \frac{\left(\vec{r}_i - \vec{r}_j\right)}{\left|\vec{r}_i - \vec{r}_j\right|}$

Convection

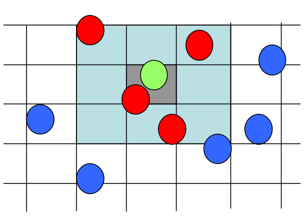


Molecular Dynamics example from astrophysics



Algorithmic trick(s) for speed-up

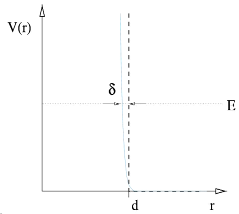
- Linked cells neighborhood search O(1) (*short range forces*)



- Linked cells update after 10-100 time-steps O(N)

What is Molecular Dynamics ?

1. Specify interactions between bodies (for example: two spherical atoms)



2. Compute all forces $\mathbf{f}_{j\to i}$

3. Integrate the equations of motion for all particles (Verlet, Runge-Kutta, Predictor-Corrector, …) with fixed time-step dt

$$m\ddot{\mathbf{x}}_i = \sum_{j\neq i} \mathbf{f}_{j\to i}$$

Rigid interaction (hard spheres)

Stiff (rigid) interactions require dt=0
Events (=collisions) occur in **zero-time** (instantaneously)
that means: Integration is *impossible* !

1. Propagate particles between collisions
2. Identify next event (collision)
3. Apply collision matrix

Why use hard spheres ?

+ advantages
- Event driven (ED) is faster than MD
- Analytical kinetic theory is available (with 99.9% agreement)

– drawback
- Implementation of arbitrary forces is expensive
- Parallelization is less successful

Why use **hard** spheres ?

+ advantages

- Event driven (ED) is faster than MD
- Analytical kinetic theory is available
  (with 99.9% agreement)

– drawback

- Implementation of arbitrary forces is expensive
- Parallelization is less successful

Algorithm (serial)

0. Initialize

- Compute all forces O(1)
- Integrate equations of motion $t+dt$
- O(N) – goto 1.

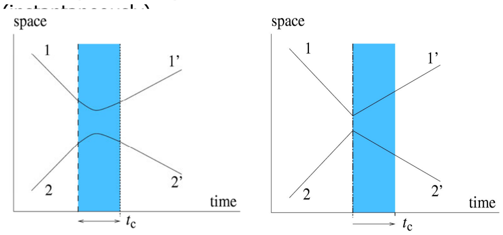
Total effort: O(N)

Rigid interaction (hard spheres)

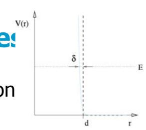
0. Stiff (rigid) interactions require dt=0
   Events (=collisions) occur in **zero-time** (instantaneously)
   Integration is *impossible* !
1. Propagate particles between collisions
2. Identify next event (collision)
3. Apply collision matrix

Rigid interaction (hard spheres)

1. Stiff (rigid) interactions require dt=0
   Events (=collisions) occur in **zero-time**



Rigid interaction (hard spheres



2. Solve equation of motion between collision

- trajectory $\mathbf{x}_i(t)=\mathbf{x}_i(0)+\mathbf{v}_i(0)t+\frac{1}{2}\mathbf{g}t^2$
- contact $\|\Delta\mathbf{x}_{ij}\| = \|\mathbf{x}_i(t)-\mathbf{x}_j(t)\| = r_1+r_2$

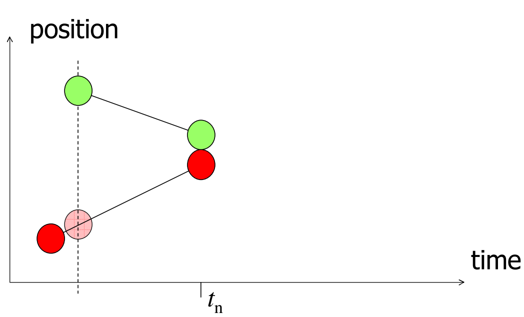
$$(\Delta\mathbf{x}_{ij}(0)+\Delta\mathbf{v}_{ij}(0)t)^2=(r_1+r_2)^2$$

$$\underbrace{\Delta\mathbf{x}_{ij}^2-(r_1+r_2)^2}_{c}+\underbrace{\Delta\mathbf{x}_{ij}\cdot\Delta\mathbf{v}_{ij}}_{b}t+\underbrace{\Delta\mathbf{v}_{ij}^2}_{a}t^2=0$$

- event-time

$$t_{1,2}=\frac{-b\pm\sqrt{b^2-4ac}}{2a}$$

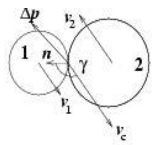
Time evolution



Rigid interaction (hard spheres)

Collision rule (translational)

$v'_{1,2} = v_{1,2} \pm (1+r)\Delta P/2m_{1,2}$

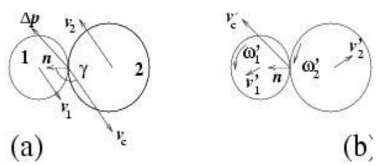


Momentum conservation + dissipation
with restitution coefficient (normal): r

Rigid interaction (hard spheres)

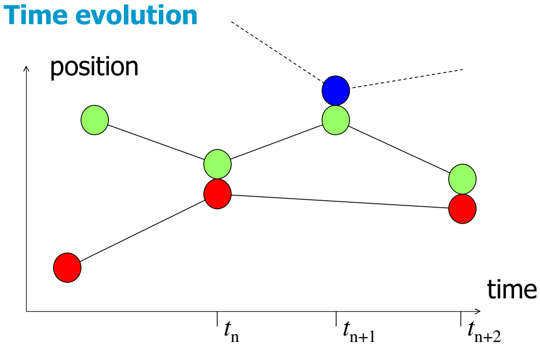
Collision rule (translational and rotational)

$v'_{1,2} = v_{1,2} \pm (1+r)\Delta P/2m_{1,2}$ $\qquad \omega'_{1,2} = \omega_{1,2} \pm (1+r_t)\Delta L/2I_{1,2}$



Restitution coefficient (normal): r (tangential) r_t

Time evolution



Algorithm (ED serial)

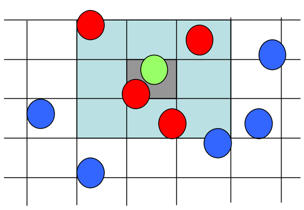
0. Initialize

- Propagate particle(s) to next event O(1)
- Compute event (collision or cell-change)
- Calculate new events and times O(1)
- Update priority queue (heap tree) O(log$N$)
- O(N) – **`goto`** 1.

Total effort: O(Nlog$N$)

Algorithmic trick(s) for speed-up

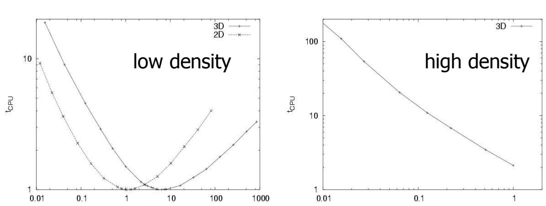
- Linked cells neighborhood search O(1) (*short range forces*)



- Linked cells update not needed !

Performance

- Short range contacts
- Linked cells neighbourhood search

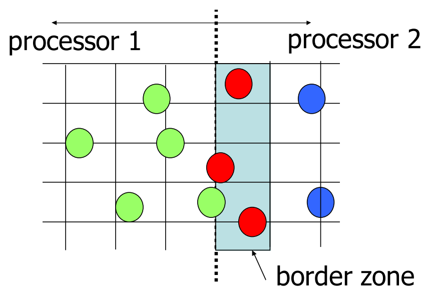


Cells per particle

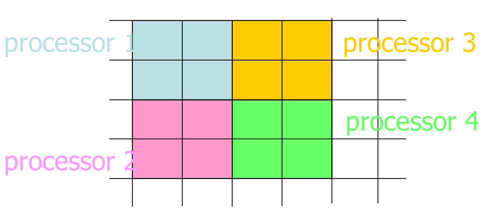
Algorithm (parallel)

0. Initialize
- Communication between processors
- Process next events t_n to t_{n+m} (see serial)
- Send and receive border-particle info
- **If** causality error **then** rollback **goto** 2.
- Synchronisation (for load-balancing and I/O)
- **goto** 1.

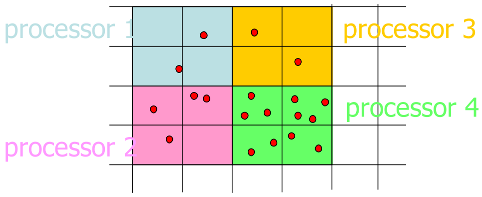
Parallelization – communication



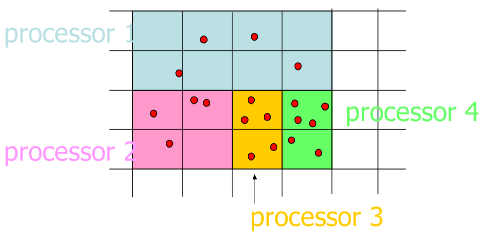
Parallelization – load balancing



Parallelization – load balancing

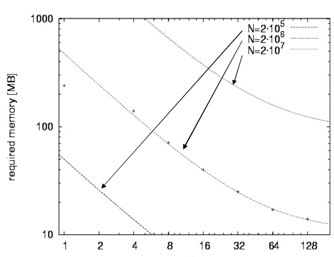


Parallelization – load balancing



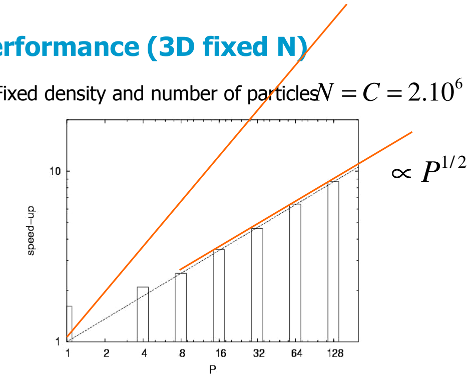
Performance (fixed N)

- Required memory per processor [MByte] $= N\left(\frac{c_1}{P} + \frac{c_2}{\sqrt[3]{P}} + c_3\right)$



Performance (3D fixed N)

- Fixed density and number of particles $N = C = 2.10^6$



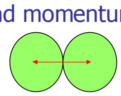
The End

From Boltzmann (low density) …

- binary collisions
- successive collisions are uncorrelated
- neglect boundary effects

… to Chapman-Enskog (high density)

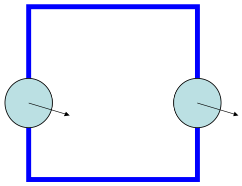
- collision rate & pressure increase with density
- add coll.-transport of energy and momentum

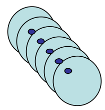


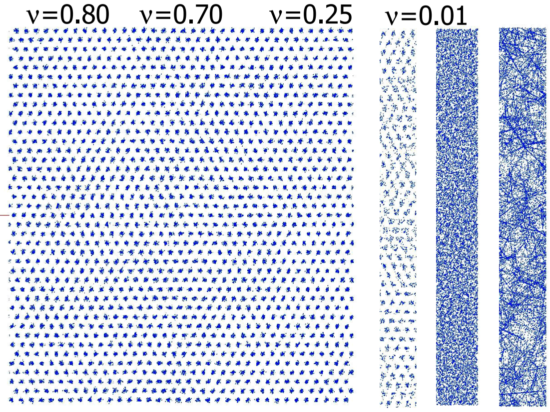
Elastic Hard Sphere Model

simulate 2000 particles
in a peridoc box

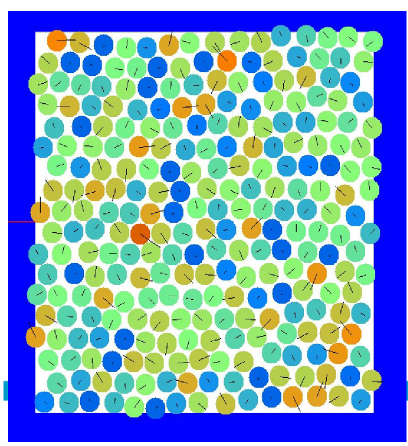
... plot streaklines





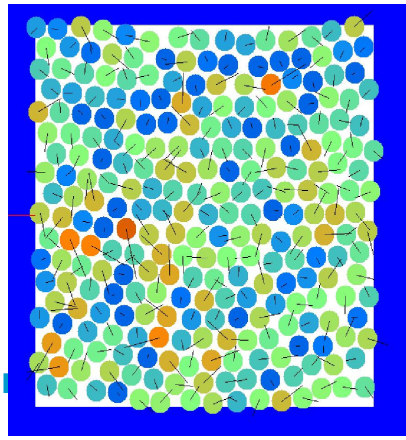


ν=0.80 ν=0.70 ν=0.25 ν=0.01

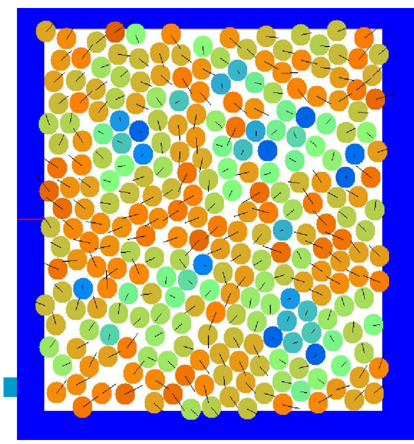


r=0.95

ν=0.70

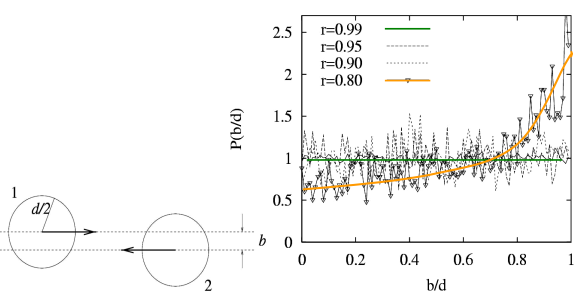


r =0.90

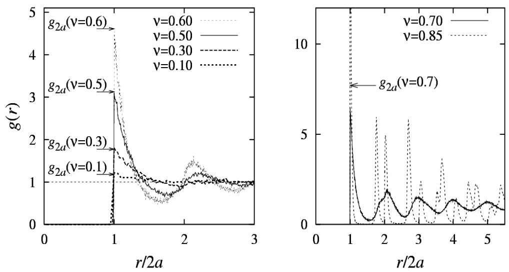


r =0.80

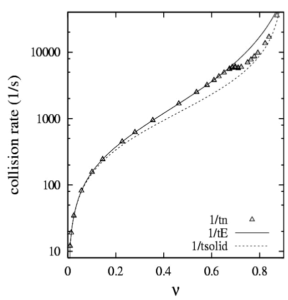
Collision parameter



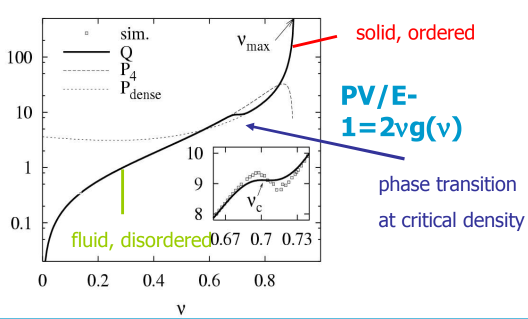
Contact probability – correlation function



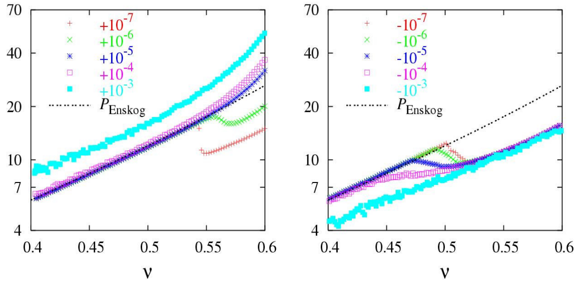
Collision rate – time scale



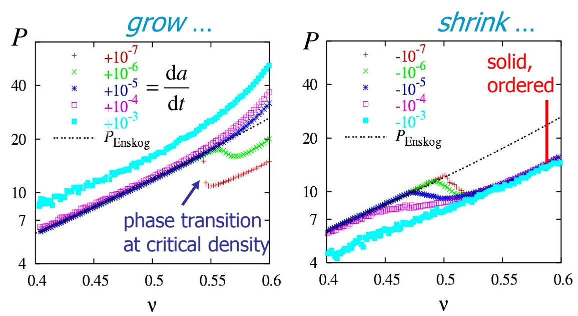
Pressure (Equation of State – 2D)



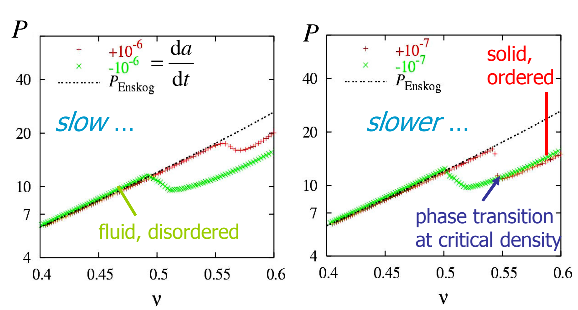
Pressure (Equation of State – 3D)



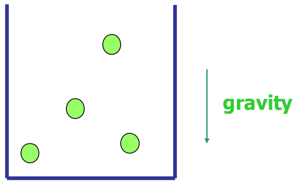
Pressure (Equation of State – 3D)



Pressure (Equation of State – 3D)

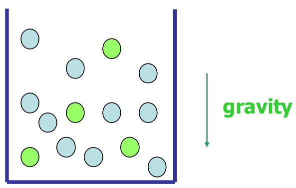


Elastic hard spheres in gravity



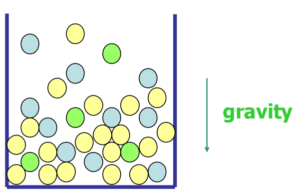
- N particles
- Kinetic Energy
- What is the *density profile* ?

Elastic hard spheres in gravity



- N particles
- Kinetic Energy
- What is the *density profile* ?

Elastic hard spheres in gravity



- N particles
- Kinetic Energy
- What is the *density profile* ?

Elastic hard spheres in gravity

elastic steady state: $\frac{\partial}{\partial t} = 0 \qquad u_i = I = 0$

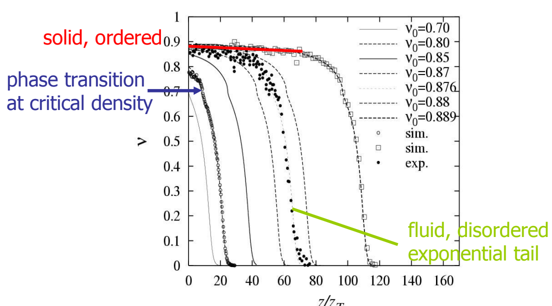
mass & energy conservation – OK

momentum balance:

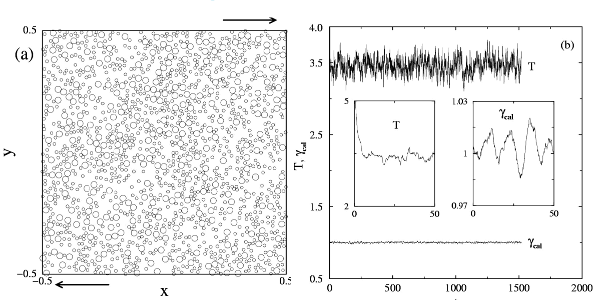
$$0 = -\frac{\partial}{\partial x_i} P + \rho g_i$$

- **Pressure *P global equation of state***
- **Shear Stress** $\sigma_{ij}^{dev} = 0$
- **Energy Dissipation Rate *I=0***

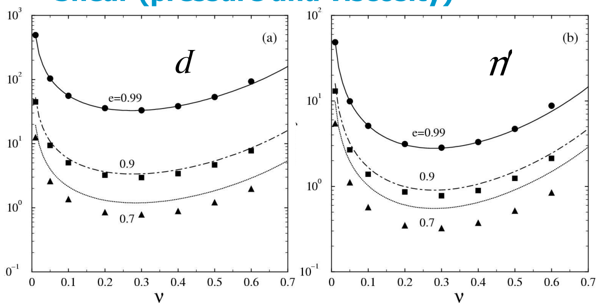
Hard sphere gas in gravity



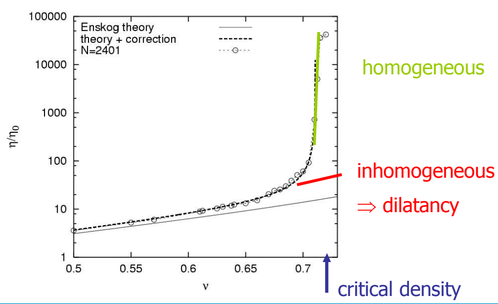
Shear (energy and rate)



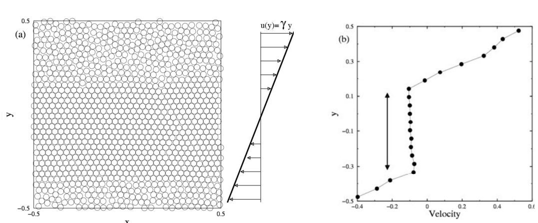
Shear (pressure and viscosity)



Shear (viscosity at high density)



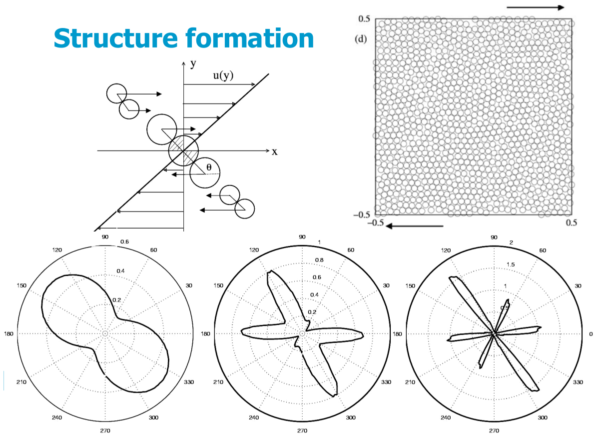
Structure formation



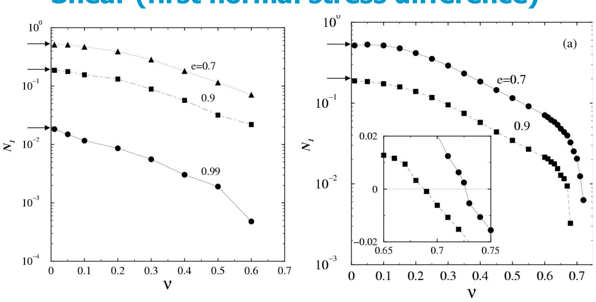
Low density -> linear velocity profile

High density -> shear localization

Structure formation

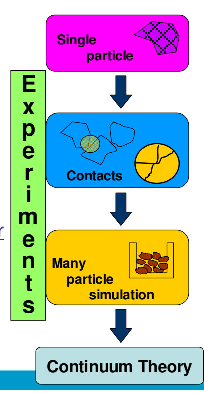


Shear (first normal stress difference)

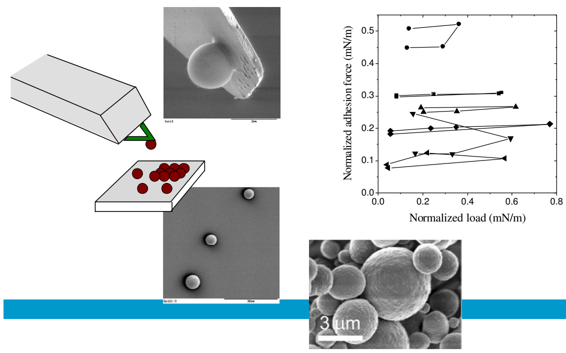


Approach philosophy

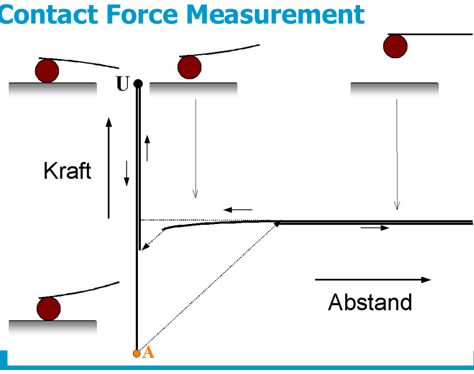
- Introduction
- Single Particles
- Particle Contacts/Interactions
- Many particle cooperative behavior
- Applications/Examples
- Conclusion



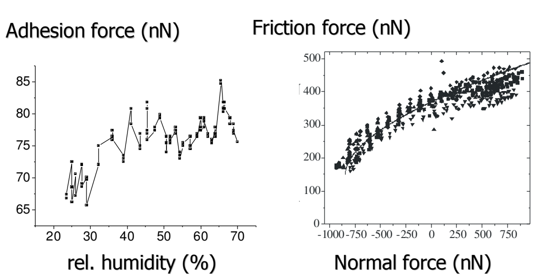
Contact force measurement (PIA)



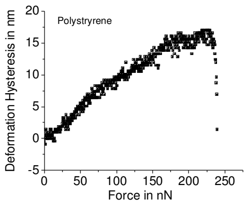
Contact Force Measurement

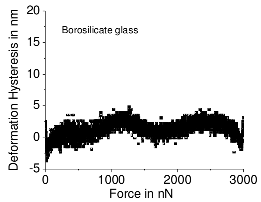


Adhesion and Friction



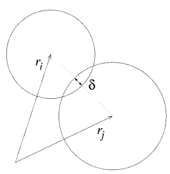
Hysteresis (plastic deformation)





Collaborations:
MPI-Polymer Science (Butt et al.)
Contact properties via AFM

Discrete particle model

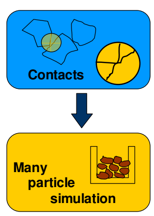


Equations of motion

$$m_i \frac{d^2 \vec{r}_i}{dt^2} = \vec{f}_i$$

Forces and torques:

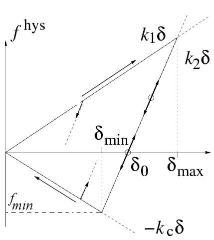
$$\vec{f}_i = \sum_c \vec{f}_i^c + \sum_w \vec{f}_i^w + m_i g$$



Contact if Overlap > 0

Overlap $\delta = \frac{1}{2}(d_i + d_j) - (\vec{r}_i - \vec{r}_j) \cdot \vec{n}$

Normal $\hat{n} = \vec{n}_{ij} = \frac{(\vec{r}_i - \vec{r}_j)}{|\vec{r}_i - \vec{r}_j|}$



Contact model

- (too) simple ☺
- piecewise linear
- easy to implement

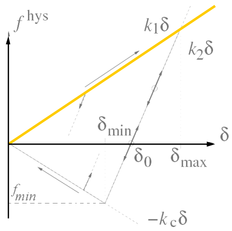
$$f_i^{hys} = \begin{cases} k_1\delta & \text{for loading} \\ k_2(\delta - \delta_0) & \text{for un-/reloading} \\ -k_c\delta & \text{for unloading} \end{cases}$$

Maximum overlap δ_{max}

stress-free overlap $\delta_0 = (1 - k_1/k_2)\delta_{max}$

strongest attraction at: $\delta_{min} = \frac{k_2 - k_1}{k_2 + k_c}\delta_{max}$

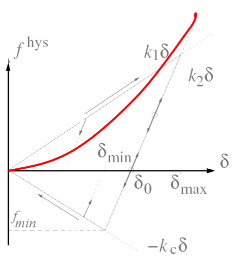
the max. attractive force: $f_{min} = -k_c\delta_{min}$



Linear Contact model

- (really too) simple ☺
- linear
- very easy to implement

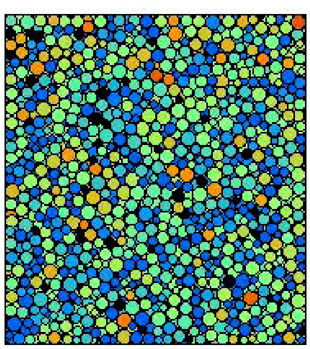
$$f_i^{hys} = \begin{cases} k_1\delta & \text{for un-/re-loading} \\ & \end{cases}$$



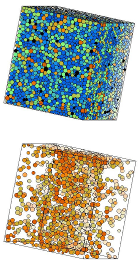
Hertz Contact model

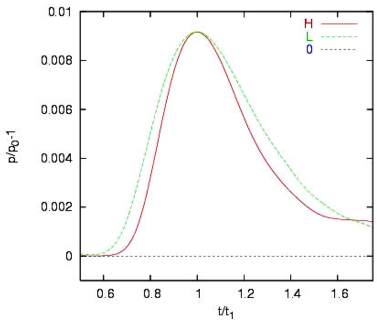
- simple ☺
- non-linear
- easy to implement

$$f_i^{hys} = \begin{cases} k_1\delta^{3/2} & \text{for un-/re-loading} \\ & \end{cases}$$



Sound





P-wave shape and speed

Stefan Luding, s.luding@tnw.tudelft.nl
Particle Technology, DelftChemTech, Julianalaan 136, 2628 BL Delft

Open questions

- Agglomeration
  - population balance – cluster evolution
  - phase transitions, cooperative behavior
- Main challenges (for modeling)
  - aero-/hydro-dynamics coupling
  - cluster stability, statistics, sintering, ...