Perspectives of Granular Dynamics and Lattice Boltzmann Methods in Porous Media Applications

Ulrich Rüde
(LSS Erlangen, ruede@cs.fau.de)

Lehrstuhl für Simulation
Universität Erlangen-Nürnberg
www10.informatik.uni-erlangen.de

Reactive Flows in Deformable, Complex Media
Oberwolfach
September 22-27, 2014
Outline

- Motivation
  - Is direct numerical simulation of the pore scale feasible?

- Building Blocks
  1. Supercomputing: scalable algorithms, efficient software
  2. Solid phase - rigid body dynamics
  3. Fluid phase - Lattice Boltzmann method
  4. Electrostatics: finite volume multigrid
  5. Gas phase - free surface tracking, volume of fluids

- Multi-physics applications
  - Coupling elementary models
  - Examples

- Perspectives
Mega=10^6, Giga=10^9, Tera=10^{12}, Peta=10^{15}, Exa=10^{18}

- World has a population of 7 \times 10^9 humans
- Earth is 4.6 \times 10^9 years old
  - the oceans together have ca. 1.3 \times 10^9 km^3
  - the mantle has 0.91 \times 10^{12} km^3
  - 10^{12} finite elements can resolve the volume of the mantle with ca. 1 km resolution
- Number of stars in the galaxy: 10^{11}
- Avogadro’s constant: 6 \times 10^{23} mol^{-1}
- The recirculatory system contains 2.5 \times 10^{13} red blood cells
- The brain has ca. 10^{11} Neurons
- Processor chip has
  - 5 \times 10^9 transistors
  - 3 GHz clock rate: 3 \times 10^9
  - can perform 10^{11} (= 100 Giga) Flops
  - Supercomputer at 2020: 10^{18} Flops

An exa-scale system with 10^{18} Flops should suffice to resolve the meso-scale
Building Block I:

Current and Future High Performance Supercomputers
Two Multi-PetaFlops Supercomputers

JUQUEEN
- Blue Gene/Q architecture
- 458,752 PowerPC A2 cores
- 16 cores (1.6 GHz) per node
- 16 GiB RAM per node
- 5D torus interconnect
- TOP 500: 8, Europe‘s fastest supercomputer

SuperMUC
- Intel Xeon architecture
- 147,456 cores
- 16 cores (2.7 GHz) per node
- 32 GiB RAM per node
- Pruned tree interconnect
- TOP 500: 10, World‘s fastest x86-based supercomputer
Computer Architecture is Hierarchical

Core
- vectorization (SSE, AVX), i.e. vectors of 2-8 floating point numbers must be treated in blocks:
- may have their own cache memories
- access to local (cache) mem fast
- access to remote mem slow
- pipelining, superscalar execution
- each core may need several threads to hide memory access latency

Node
- Several CPU chips (2) may be combined with local memory to become a node
- Several cores (8) are on a CPU chip
- Within a node we can use „shared memory parallelism“
  e.g. OpenMP
- Several cores may share second/third level caches
- Memory access bottlenecks may occur
- Sometimes nodes are equipped with accelerators (i.e. graphics cards)

Cluster
- thousands of nodes are connected by a fast network
- different network topologies
- between nodes message passing must be used
  e.g. MPI
- high latency
- low bandwidth
Building block II:

Granular Media Simulations

with the physics engine

with T. Preclik and K. Iglberger


1250000 spherical particles, 256 CPUs, 300300 time steps, runtime: 48h (including data output), texture mapping, ray tracing
Newton-Euler Equations for Rigid Bodies

\[
\begin{pmatrix}
\dot{x}(t) \\
\dot{\phi}(t)
\end{pmatrix} =
\begin{pmatrix}
v(t) \\
Q(\phi(t))\omega(t)
\end{pmatrix}
\]

\[
M(\phi(t))
\begin{pmatrix}
\dot{v}(t) \\
\dot{\omega}(t)
\end{pmatrix} =
\begin{pmatrix}
f(s(t), t) \\
\tau(s(t), t) - \omega(t) \times I(\phi(t))\omega(t)
\end{pmatrix}
\]

Contact detection
- minimizing signed distance function
- Time-continuous non-penetration constraint for hard contacts

Coulomb friction

Parallelization via domain partitioning
- Advanced parallel data structures for contact detection
- Sophisticated protocol for synchronization
Scaling Results

- The solver is not optimal, hence cannot scale unconditionally.
- Strong and weak scaling results for a constant number of iterations performed SuperMUC and Juqueen.

Largest runs:
- $2.8 \times 10^{10}$ non-spherical particles
- $1.1 \times 10^{10}$ contacts

hexagonal close packing of spheres, gravitational acceleration as a test problem: scaling results

![Graphs showing scaling results with speedup, parallel efficiency, average time per time step in seconds, and number of nodes.](image)
Shaker scenario with sharp edged hard objects

Department of Computer Science
Chair for System Simulation
University of Erlangen-Nürnberg

Tobias Preclik

Sharp-edged Granular Matter in a High Amplitude Horizontal Shaker

31.3.2014
Building Block III:

Scalable Flow Simulations with the Lattice Boltzmann Method
The Lattice-Boltzmann-Method

- Discretization in squares or cubes (cells)
- Particles Distribution Functions (PDF) \( \vec{f} = (f_0, f_1, \ldots, f_{N-1}) \)
  - 9 numbers in 2D (2DQ9)
  - in 3D: D3Q19 (alternatives D3Q27, etc)
- Repeat (many times)
  - stream
  - collide
The stream step

Move PDFs into neighboring cells

\[ f_i(\vec{x} + \vec{e}_i, t + 1) - f_i(\vec{x}, t) = C_i(\vec{f}(\vec{x}, t)) \]

- **Non-local part**, **Linear** propagation to neighbors 
  (stream step)
- **Local part**, **Non-linear** operator, 
  (collide step)
The collide step

Compute new PDFs modelling „molecular collisions“

Most collision operators can be expressed as

\[ C_i = A_{ij} [f_j(x^*, t) - \tilde{f}^{eq}_j(x^*, t)]. \]

The *equilibrium function* is a non-linear function, depending on the conserved momenta \( \rho, \vec{u}, \) and \( \vec{f}. \)

\[ \tilde{f}^{eq}(x^*, t) = \tilde{f}^{eq}(\rho(x^*, t), \vec{u}(x^*, t)) \]
Basic Lattice Boltzmann Method

Single Relaxation Time (SRT)

\[ f_i(x + e_i \delta t, t + \delta t) = f_i(x, t) - f_i(x, t) - f_i^{eq}(u(x, t), \rho(x, t)) \]

\[ \tau \]

Macroscopic quantities

\[ \rho = \sum f_i \quad \rho u = \sum e_i f_i \]

Equilibrium distribution function

\[ f_i^{eq}(u, \rho) = \omega_i \rho \left( 1 + \frac{e_i \cdot u}{c_s^2} + \frac{(e_i \cdot u)^2}{2c_s^4} - \frac{3u^2}{2c_s^2} \right) \]
MRT, TRT, SRT

\[ C'_i = A_{ij} [f_j(x, t) - f^eq_j(x, t)]. \]

- Family of most common collision operators
  - *Relaxation* towards equilibrium
  - The *generalized LBE* (→ *d'Humieres '92*) is a way to construct collision operators with a maximal degree of freedom (i.e., maximal number of relaxation times).
    - MRT: Multiple Relaxation Times model
    - TRT: Two Relaxation Times model
    - SRT: Single Relaxation Times model (*BGK - Model*)
- Which relaxation parameters do I need?
- How many relaxation parameters should I use?
Important Subclasses

- SRT:
  - Only use 1 relaxation time
    $$\omega := -s_\nu = -s_e = -s_\epsilon = -s_q$$
    → \textit{Lattice BGK model} (Bhatnagar, Gross, Krook)
  - Extremely easy to implement and computationally efficient

- TRT (→ Ginzburg et al. 2008)
  - Use a pair $\lambda^+$, $\lambda^-$ for the relaxation of moments of \textit{even order} (here: $s_\nu$, $s_e$, $s_\epsilon$) and of \textit{odd order} (here: $s_q$)
  - Implementation can be made equally efficient (*) as SRT.

- One relaxation time (SRT) is often used but may exhibit artifacts
waLBerla

Massively Parallel Multiphysics Framework

- Focus on lattice Boltzmann method
- Hybridly parallelized (MPI + OpenMP)
- Painstakingly optimized:
  - Machine-specific kernels for max performance
  - All data structures exa-scalable

Weak scaling (Lid Driven Cavity) TRT

JUQUEEN
16 processes per node
4 threads per process

SuperMUC
4 processes per node
4 threads per process

2.1 × 10^{12} cell updates per second (TLups)

0.837 × 10^{12} cell updates per second (GLups)
Summary of Performance Evaluation on Coronary Geometry

- Weak scaling on JUQUEEN with 458,752 processes
  - over a trillion ($10^{12}$) fluid lattice cells
  - Cell sizes of 1.27µm (diameter of red blood cells about 7µm)
  - 2.1 $10^{12}$ cell updates per second
  - 0.41 PFlops

- Strong scaling at cell sizes of 0.1 mm
  - In excess of 6000 time steps per second on 32,768 cores of SuperMUC
  - 2.1 million fluid cells

Boundary Conditions

Dirichlet boundaries for velocity

Most popular: *Bounce Back* rule

\[ f_i(x_b, t + 1) = f_i(x_b, t) + 6 \rho w_i \vec{e}_i u_w \]

- Boundary condition is imposed approx. *half-way* between boundary node and wall node
- Simple implementation (no neighborhood information needed)

Approximate porous medium as *flag field* (0: fluid, 1: solid)

- LB scheme does not require high resolutions to cover hydrodynamics correctly
- 4 lattice sites per pore diameter usually sufficient? \((\rightarrow\text{ Succi 2001})\)
Porous Media

- Benchmark:
  Body Centered Array of Spheres (3D)

- *Pan et al. 2005* suggest a discretization with $dx = \frac{L}{32}$;

- Periodic domain; flow driven by body force $g$

- Estimation of permeability $k$ from the simulated average flow velocity $u_d$

\[ u_d = \frac{k}{\mu} \left[ \nabla P + \rho \cdot g \right] \quad \text{(Darcy's law)} \]

- Study of collision models and boundary conditions: SRT (BGK) vs. TRT
  - SRT: $\lambda = \lambda^+ = \lambda^- = \frac{-2}{6\nu + 1}$
  - TRT: $\lambda^+ = \frac{-2}{6\nu + 1}$; $\lambda^- = \ldots$

  *(find out optimal choice)*
Porous Media (cont.)

- Optimal choice for second relaxation time
- Numerical artifact: Permeability obtained from simulation depends on relaxation times
- BGK: no correction possible

\[ \lambda = \lambda^+ = \lambda^- = \frac{-2}{6\nu + 1} \]

- TRT (*): optimal choice for second parameter

\[ \lambda^+ = \frac{-2}{6\nu + 1}; \quad \lambda^- = -8\frac{\lambda^+ + 2}{\lambda^+ + 8} \]
Deriving macroscopic closure laws

Lattice Boltzmann Study of the Drag Correlation in Fluid-Particle Systems


Drag Correlations

• Fluid-solid systems
• Important in chemical engineering (fluidized beds, hydrocyclone, thickener, flotation columns)
• Relate the drag force $f_d$ per particle to the
  • local particle *Reynolds number* $Re_p$ (relative velocity) and
  • *solid volume fraction* $\varphi$

$$f_d \sim F(Re_p, \varphi)$$

• Examples: Wen & Yu (1966), Ergun (1952)
Simulation setup - Drag Computation

- **Random particle beds (homogeneous):**
  - Number of particles $N = 27$
  - Spherical particles of volume $V_p = \pi d^3 / 6$
  - Solid volume fraction $\varphi = \frac{N V_p}{V}$

- Cubic domain length $L$ with $V = L^3$.
- Periodic boundaries along $x$, $y$ and $z$ direction!

- **Simulation of flow:**
  - Accelerating gravity $a$
  - Reynolds number $Re_p = \frac{\rho \bar{u} d}{\eta}$,

where $u = \frac{1}{V} \sum_{x \in \Omega} u(x)$ is the average flow rate.
Flow field and vorticity

- 2D slice visualized
- Domain size: $L = 450^3$
- Re = 300
- Volume fraction: $\varphi = 0.01$
Drag Computation (2)

- **Average drag:**
  \[ f_d = \frac{1}{N} \sum_{p=1}^{N} f_p \]
  
  ... is estimated for each simulation (steady state).

- **The (total) hydrodynamic force is**
  \[ f_t = f_d + f_b = f_d + V_p \rho \alpha \]

- **The dimensionless drag** is defined by
  \[ C = \frac{f_t}{3\pi \eta d\bar{u}} \]
  
  (normalized using the Stokes drag).

---

Figure 1: Streamlines in bed of solid volume fraction \( \varphi = 0.35 \) in the laminar \( (Re_p = 3.6, \text{ left}) \), and in the transient regime \( (Re_p = 300, \text{ right}) \). Blue color indicates lower, red color indicates a higher flow velocity, respectively.
Finding a new drag correlation

- 5 runs (beds) for each pair \((\varphi, \text{Re}_p)\)
  \[\varphi \in \{0.01, 0.03, 0.05, 0.08, 0.1, 0.2, 0.3, 0.35\}\]
  \[\text{Re}_p = 0.05..300\]
- Ansatz function:
  \[F(\varphi, \text{Re}_p) = (1 - \varphi)^a [b + c\text{Re}_p^d + e(1 + \text{Re}_p)^{-f}\varphi + g(1 + \text{Re}_p)^{-h}\varphi]\]
  - Undetermined: \(a, b, .., h\)
  - Optimization (conjugate gradient) to minimize
    \[E = \sum_{(\varphi,\text{Re}_p)} \frac{|F - C^*|}{F},\]
  - where \(C^*\) is the averaged dimensionless drag obtained from simulations.
New drag correlation

(a) Normalized drag force based on total hydrodynamic force as a function of solid volume fraction for different particle Reynolds number.

(b) The normalized drag force as a function of the Reynolds number $Re_p$ for various solid volume fractions.

- Finally, the drag correlation reads

$$F = (1 - \varphi)^{-5.726} \left[ 1.751 + 0.151 \, Re_p^{0.684} \right.$$

$$\left. - 0.445 \, (1 + Re_p)^{1.04\varphi} - 0.16 \, (1 + Re_p)^{0.0003\varphi} \right].$$

- Average absolute percentage error: 9.7 %
Direct Numerical Simulation for periodic and random dense sphere packings

joint work with Ehsan Fattahi, Christian Waluga, Barbara Wohlmuth
Densily packed periodic sphere beds

- Periodic boundary with spheres of equal radius
- Flow driven with acceleration as a source term
- Validation against semi-analytic solution of Sangani and Acrivos (1982)

\[
C_D = \frac{F_D}{6\pi \mu u_D r}
\]

\[
C_D = \sum_{n=0}^{30} \alpha_n \chi^n, \quad \chi = \left( \frac{c}{c_{max}} \right)^{1/3}, \quad c = \frac{4\pi r^3}{3L^3}, \quad c_{max} = \frac{\pi}{6}
\]

\[
K = \frac{\mu u_D}{dP/dx}
\]
Drag Force, Discretization Effect

- Different collision operator,
- Different boundary condition scheme
Flow over a „periodic“ porous media

- Radius of each sphere is changed to change the porosity and the permeability. (6<r<12)
- Simple cubic sphere pack is used to fill the porous part.
- 5 row of spheres are used in z direction
- x and y direction periodic
- One column of sphere is used
- The Stokes flow simulation is done with TRT-LIMR
- The velocity is averaged over a x-y plane
Different Re Number

- Radius of each sphere is set to 6 cells.
- $1 \times 10^{-5} < \text{Re} < 1 \times 10^{1}$
- The Reynolds Number changed by changing the pressure gradient in the channel
- The resolution and the simulation setup is the same as previous slide
Construction of random spherical structure for porous media with the PE
Pore scale simulation
Free flow over a porous region

- The result of velocity magnitude
- on planar cut
- sphere color encodes different sphere sizes
- clipped on the top

- Streamline of the DNS solution
- Stokes flow
- Colored by velocity magnitude
Turbulent flow over a permeable wall

- $\text{Re} \approx 3000$, DNS
- 3 level of refinement used
- Volume rendering of velocity magnitude
- Periodic in X and Y direction
- Flow driven by periodic-pressure boundary condition
- Non-uniform pressure gradient
- 6 mega cells in the simulation
- 300 loops of flow over periodic channel
- Video from transition regime
- LR10 cluster, $7 \times 32 \times 19 = 4256$ core hours
- 1,300,000 timesteps
- 8 times more timestep on the finest level
Multi-Physics Simulations for Particulate Flows

Parallel Coupling with waLBerla and PE

with D. Bartuschat and K. Gustavsson
(KTH Stockholm)
Fluid-Structure Interaction
direct simulation of Particle Laden Flows (4-way coupling)

Mapping Moving Obstacles into the LBM Fluid Grid

An Example
Mapping Moving Obstacles into the LBM Fluid Grid

An Example (2)

Cell change from fluid to particle

Perspectives of LBM for PM  -  Ulrich Rüde
Heterogenous CPU-GPU Simulation
with C. Feichtinger, H. Köstler, J. Habich, G. Wellein, T. Aoki (Tokyo Tech)

Fluidized Beds:

Direct numerical simulation
fully resolved particles

Fluid-structure-interaction

4-way-coupling

Particles: 31250, Domain: 400x400x200, Timesteps: 400 000
Devices: 2 x M2070 + 1 Intel „Westmere“, Runtime: 17.5 h
Direct numerical simulation of charged particles in flow


Positive and negatively charged particles in flow subjected to transversal electric field

Building Block IV (electrostatics)
6-way coupling

- electrostatic fields
  - FV-discretization - cell centered multigrid solver
  - lubrication correction
Separation experiment

![Separation experiment diagram](image)

### Runtimes of charged particle algorithm sweeps

<table>
<thead>
<tr>
<th>Number of nodes</th>
<th>Total runtime</th>
<th>LBM</th>
<th>Map</th>
<th>Lubr</th>
<th>HydrF</th>
<th>ptCm</th>
<th>ElectF</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>2</td>
<td>200</td>
<td>200</td>
<td>200</td>
<td>200</td>
<td>200</td>
<td>200</td>
<td>200</td>
</tr>
<tr>
<td>4</td>
<td>400</td>
<td>400</td>
<td>400</td>
<td>400</td>
<td>400</td>
<td>400</td>
<td>400</td>
</tr>
<tr>
<td>8</td>
<td>800</td>
<td>800</td>
<td>800</td>
<td>800</td>
<td>800</td>
<td>800</td>
<td>800</td>
</tr>
<tr>
<td>16</td>
<td>1600</td>
<td>1600</td>
<td>1600</td>
<td>1600</td>
<td>1600</td>
<td>1600</td>
<td>1600</td>
</tr>
<tr>
<td>32</td>
<td>3200</td>
<td>3200</td>
<td>3200</td>
<td>3200</td>
<td>3200</td>
<td>3200</td>
<td>3200</td>
</tr>
<tr>
<td>64</td>
<td>6400</td>
<td>6400</td>
<td>6400</td>
<td>6400</td>
<td>6400</td>
<td>6400</td>
<td>6400</td>
</tr>
<tr>
<td>128</td>
<td>12800</td>
<td>12800</td>
<td>12800</td>
<td>12800</td>
<td>12800</td>
<td>12800</td>
<td>12800</td>
</tr>
<tr>
<td>256</td>
<td>25600</td>
<td>25600</td>
<td>25600</td>
<td>25600</td>
<td>25600</td>
<td>25600</td>
<td>25600</td>
</tr>
</tbody>
</table>

### Weak scaling performance of MG and LBM

<table>
<thead>
<tr>
<th>Number of nodes</th>
<th>LBM Perform.</th>
<th>MG Perform.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>2</td>
<td>200</td>
<td>200</td>
</tr>
<tr>
<td>4</td>
<td>400</td>
<td>400</td>
</tr>
<tr>
<td>8</td>
<td>800</td>
<td>800</td>
</tr>
<tr>
<td>16</td>
<td>1600</td>
<td>1600</td>
</tr>
<tr>
<td>32</td>
<td>3200</td>
<td>3200</td>
</tr>
<tr>
<td>64</td>
<td>6400</td>
<td>6400</td>
</tr>
<tr>
<td>128</td>
<td>12800</td>
<td>12800</td>
</tr>
<tr>
<td>256</td>
<td>25600</td>
<td>25600</td>
</tr>
</tbody>
</table>

Overall the coupled multiphysics algorithm achieves 83% parallel efficiency on 2048 nodes. Since most time is used for the Poisson equation (SetRHS), communication of the electric potential (HydrF), and the two weak scaling sweeps, such as LBM, moving obstacle mapping (Map), Hydrodynamic force computation (HydrF), and the lubrication correction (Lubr), this leads to a speedup of 64%. The LBM performs 95,372 MLUPS, with 91% parallel efficiency. The MG solver in Gmeiner et al. (2014) corresponds to 170 MLUPS. This performance is significantly lower than the LBM performance.
Building Block V

Volume of Fluids Method for Free Surface Flows

joint work with Regina Ammer, Simon Bogner, Martin Bauer, Daniela Anderl, Nils Thürey, Stefan Donath, Thomas Pohl


Free Surface Flows

- Volume-of-Fluids like approach
- Flag field: Compute only in fluid
- Special “free surface” conditions in interface cells
- Reconstruction of curvature for surface tension
Free Surface Bubble Model

Data of a Bubble:
- Initial Volume (Density=1)
- Current Volume
- Density/Pressure = initial volume / current volume

Update Management
- Each process logs change of volume due to cell conversions and mass variations in Interface cells
- All volume changes are added to the volume of the bubble at the end of the timestep (which also has to be communicated)
Bubble data includes

- Process list
- Merge information


Donath, Feichtinger, Pohl, Götz, UR: *Localized Parallel Algorithm for Bubble Coalescence in Free Surface Lattice-Boltzmann Method*, in LNCS 5704.
Additive Manufacturing
Fast Electron Beam Melting


Motivating Example: Simulation of Electron Beam Melting Process (Additive Manufacturing)

- EU-Project Fast-EBM
  - ARCAM (Sweden)
  - TWI (Cambridge)
  - WTM (FAU)
  - ZISC (FAU)

- Generation of powder bed
- Energy transfer by electron beam
  - modeling penetration depth

- heat transfer
- Flow dynamics
- Melting/solidification
- phase transition
- surface tension
- fluid flow
- wetting, capillary forces

Joint work with C. Körner, M. Markl, R. Ammer
Simulation of Electron Beam Melting

Simulating powder bed generation using the PE framework

WalBerla Simulation

High speed camera shows melting step for manufacturing a hollow cylinder

Perspectives of LBM for PM - Ulrich Rüde
Conclusions and Perspectives

- Supercomputer power
- Versatile tools
  - Lattice Boltzmann
  - Multibody Dynamics
- Limitations to pore scale resolution
  - ensemble size
  - time steps
- Gain insight
  - Multi-scale
  - Multi-Physics
- Challenges
  - validation
  - software
Thank you for your attention!

Videos, preprints, slides at https://www10.informatik.uni-erlangen.de