Challenges and Opportunities of Extreme Scale Numerical Simulation

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

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

SEMINARUL STIINTIFIC DE MATEMATICA APLICATA
Constanta
Wednesday, October 15, 2014
Outline

- Motivation
  - Direct Numerical Simulation
- 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

<table>
<thead>
<tr>
<th>JUQUEEN</th>
<th>SuperMUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Blue Gene/Q architecture</td>
<td>• Intel Xeon architecture</td>
</tr>
<tr>
<td>• 458,752 PowerPC A2 cores</td>
<td>• 147,456 cores</td>
</tr>
<tr>
<td>• 16 cores (1.6 GHz) per node</td>
<td>• 16 cores (2.7 GHz) per node</td>
</tr>
<tr>
<td>• 16 GiB RAM per node</td>
<td>• 32 GiB RAM per node</td>
</tr>
<tr>
<td>• 5D torus interconnect</td>
<td>• Pruned tree interconnect</td>
</tr>
<tr>
<td>• TOP 500: 8, Europe‘s fastest supercomputer</td>
<td>• TOP 500: 10, World‘s fastest x86-based supercomputer</td>
</tr>
<tr>
<td>Core</td>
<td>Node</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
</tr>
<tr>
<td>✤ vectorization (SSE, AVX), i.e. vectors of 2-8 floating point numbers must be treated in blocks:</td>
<td>✤ Several CPU chips (2) may be combined with local memory to become a node</td>
</tr>
<tr>
<td>✤ may have their own cache memories</td>
<td>✤ Several cores (8) are on a CPU chip</td>
</tr>
<tr>
<td>✤ access to local (cache) mem fast</td>
<td>✤ Within a node we can use „shared memory parallelism“ e.g. OpenMP</td>
</tr>
<tr>
<td>✤ access to remote mem slow</td>
<td>✤ Several cores may share second/third level caches</td>
</tr>
<tr>
<td>✤ pipelining, superscalar execution</td>
<td>✤ Memory access bottlenecks may occur</td>
</tr>
<tr>
<td>✤ each core may need several threads to hide memory access latency</td>
<td>✤ Sometimes nodes are equipped with accelerators (i.e. graphics cards)</td>
</tr>
</tbody>
</table>
Building block II:

Granular Media Simulations with the physics engine

with T. Preclik and K. Iglberger


Newton-Euler Equations for Rigid Bodies

\[
\begin{align*}
(\dot{x}(t)) &= (v(t)) \\
(\dot{\varphi}(t)) &= (Q(\varphi(t))\omega(t)) \\
M(\varphi(t))(\dot{v}(t)) &= (f(s(t), t)) \\
\omega(t) &= (\tau(s(t), t) - \omega(t) \times I(\varphi(t))\omega(t))
\end{align*}
\]

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
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

\[
\begin{align*}
    f_i(\vec{x} + e_i, t + 1) - f_i(\vec{x}, t) &= C_i(\vec{f}(\vec{x}, t)) \\
    \begin{array}{ll}
        \text{Non-local part,} & \\
        \text{Linear propagation to neighbors} & \\
        \text{(stream step)} & \\
    \end{array}
\end{align*}
\]

\[
\begin{align*}
    \begin{array}{ll}
        \text{Local part,} & \\
        \text{Non-linear operator,} & \\
        \text{(collide step)} & \\
    \end{array}
\end{align*}
\]
**The collide step**

Compute new PDFs modelling „molecular collisions“
Most collision operators can be expressed as

$$C_i = A_{ij} \left[ f_j(\vec{x}, t) - f_j^{eq}(\vec{x}, t) \right].$$

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

$$f^{eq}(\vec{x}, t) = f^{eq}(\rho(\vec{x}, t), \vec{u}(\vec{x}, t))$$
Basic Lattice Boltzmann Method

Single Relaxation Time (SRT)

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

Macroscopic quantities

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

Equilibrium distribution function

\[ f_i^{eq}(u, \rho) = \omega_i \rho \left( 1 + \frac{e_i \cdot u}{c_s^2} + \frac{(e_i \cdot u)^2}{2c_s^4} - \frac{3u^2}{2c_s^2} \right) \]
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 \times 10^{12} \text{ cell updates per second (TLups)}
0.837 \times 10^{12} \text{ cell updates per second (GLups)}
Single Node Performance

JUQUEEN

SuperMUC

vectorized

optimized

standard
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

Computational Fluid Dynamics with waLBerla

with: F. Schornbaum, C. Godenschwager, E. Fattahi:
Simulation of phantom vocal fold geometry
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 \( V_p = \frac{\pi d^3}{6} \)
  - 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 \( \alpha \)
  - Reynolds number \( Re_p = \frac{\rho \bar{u} d}{\eta} \)

where \( \bar{u} = \frac{1}{V} \sum_{x \in \Omega} u(x) \) 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
  \[ f_t = f_d + f_b = f_d + V_p \rho a \]

- 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 (\( Re_p = 3.6 \), left), and in the transient regime (\( Re_p = 300 \), right). Blue color indicates lower, red color indicates a higher flow velocity, respectively.
Finding a new drag correlation

- 5 runs (beds) for each pair \((\varphi, R_{ep})\)
  \[ \varphi \in \{0.01, 0.03, 0.05, 0.08, 0.1, 0.2, 0.3, 0.35\} \]
  \[ R_{ep} = 0.05..300 \]
- Ansatz function:
  \[
  F(\varphi, R_{ep}) = (1 - \varphi)^a [b + c R_{ep}^d + e (1 + R_{ep})^{-f \varphi} + g (1 + R_{ep})^{-h \varphi}]
  \]
  - Undetermined: \(a, b, .., h\)
  - Optimization (conjugate gradient) to minimize
    \[
    E = \sum_{(\varphi, R_{ep})} \frac{|F - C^*|}{F},
    \]
  - 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
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 rows 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
Construction of random spherical structure for porous media with the PE
Turbulent flow over a permeable wall

- Re ≈ 3000, DNS
- 3 levels 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
- I10 cluster, 7x32x19=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)


Extreme Scale Simulation - Ulrich Rüde
Mapping Moving Obstacles into the LBM Fluid Grid

An Example
Mapping Moving Obstacles into the LBM Fluid Grid

An Example (2)

Cells with state change from Fluid to Particle

Cell change from particle to fluid
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


6-way coupling

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

Figure 13 shows the cost of fluid-simulation shares on the total runtime. This diagram is based on the results from different problem sizes. However, when doubling the problem size, CG iterations required to solve the coarsest grid problem is depicted for different BCs on the boundary. Whenever the relative proportion of Neumann BCs increases, convergence deteriorates and sometimes stay constant or have to be increased. This corresponds to the expected behaviour that the number of CG iterations approximately doubles.

Table 6 in Tab. 6. The number of CG iterations required to solve size is successively doubled in all dimensions, as shown in Fig. 13 for different problem sizes. Doubling the domain in all three dimensions, the growth in the condition number Shewchuk (1994). This corresponds to the expected behaviour that the number of CG iterations approximately doubles.

Weak scaling experiments for up to 2048 nodes. The runtimes of all parts of the algorithm are shown in Tab. 6. The number of CG iterations required to solve, measured for V(3,3)-cycles and CG coarsest grid solver the finite element MG solver in Gmeiner and Rüde.
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 $p_G = \frac{V_b}{V_b(t)}$ - Gas / Gas – Interface) 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
- Joint work with C. Körner, M. Markl, R. Ammer

Heat transfer
Flow dynamics
- Melting/solidification
- phase transition
- surface tension
- fluid flow
- wetting, capillary forces
Simulation of Electron Beam Melting

Simulating powder bed generation using the PE framework

High speed camera shows melting step for manufacturing a hollow cylinder

WalBera Simulation

Extreme Scale Simulation - Ulrich Rüde
Conclusions and Perspectives

- Supercomputer power
- Versatile tools
  - Lattice Boltzmann
  - Multibody Dynamics
  - FE and Multigrid (tomorrow)
- Limitations to pore scale resolution
  - ensemble size
  - time steps
- Gain insight
  - Multi-scale
  - Multi-physics
- Challenges
  - validation
Designing Algorithms!
with four strong jet engines

Would you want to propel a Superjumbo
or with 1,000,000 blow dryer fans?

Moderately Parallel Computing
Massively Parallel MultiCore Systems

Large Scale Simulation Software

Exa-Scale Simulation Technology - Ulrich Rüde
Thank you for your attention!

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