Supercomputing for Simulation in Science and Engineering


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

Lehrstuhl für Informatik 10 (Systemsimulation)
Universität Erlangen-Nürnberg
www10.informatik.uni-erlangen.de

May 11, 2010
Peking University
Beijing, China
Where is Erlangen?
Overview

- Motivation (developments in computers)
- Towards Scalable Multilevel FE Solvers
  - Multigrid
  - HHG
  - Performance Results
- Flow Simulation with Lattice Boltzmann Methods
  - The LBM
  - Rigid Body Dynamics
  - Fluid-Structure Interaction with Moving Objects
  - Bubbly Flows and Foams
  - Animations
- Conclusions
Motivation
**How much is a PetaFlops?**

- $10^6 = 1$ MegaFlops: Intel 486
  - 33MHz PC (~1989)
- $10^9 = 1$ GigaFlops: Intel Pentium III
  - 1GHz (~2000)
  - If every person on earth computes one operation every 6 seconds, all humans together have ~1 GigaFlops performance (less than a current laptop)
- $10^{12} = 1$ TeraFlops: HLRB-I
  - 1344 Proc., ~ 2000
- $10^{15} = 1$ PetaFlops
  - 294 912 Cores (Jugene, 2009)
  - If every person on earth runs a 486 PC, we all together have an aggregate Performance of 6 PetaFlops.
- $10^{18} = 1$ ExaFlops (around 2020)?

HLRB-I: 2 TFlops

HLRB-II: 63 TFlops
IBM Blue Gene
- 0.825 petaflop/s performance speed running the Linpack benchmark.
- theoretical peak capability 1.0027 Petaflop/s
- 294 912 cores
- #4 on TOP 500 List
- Nov 2009

Björn Gmeiner (LSS) visiting Jülich Supercomputing Center
Picture taken March 2010
What’s the problem?

replacing 4 strong jet engines

Would you want to propel a Super Jumbo with 300,000 blow dryer fans?
Towards Scalable FE Software

Scalable Algorithms
and Data Structures
How fast are our algorithms (multigrid) on current CPUs

Assumptions:
- Multigrid requires 27.5 Ops/unknown to solve an elliptic PDE (Griebel ’89 for Poisson)
- A modern laptop CPU delivers >10 GFlops peak

Consequence:
- We should solve one million unknowns in 0.00275 seconds
- ~3 ns per unknown

Revised Assumptions:
- Multigrid takes 500 Ops/unknown to solve your favorite PDE
- you can get 5% of 10 Gflops performance

Consequence: On your laptop you should
- solve one million unknowns in 1.0 second
- ~1 microsecond per unknown

Consider Banded Gaussian Elimination on the Play Station (Cell Processor), single Prec. 250 GFlops, for 1000 x 1000 grid unknowns
- ~2 Tera-Operations for factorization - will need about 10 seconds to factor the system
- requires 8 GB Mem.
- Forward-backward substitution should run in about 0.01 second, except for bandwidth limitations
Multigrid: V-Cycle

Goal: solve $A^h u^h = f^h$ using a hierarchy of grids

Relax on

$A^h u^h = f^h$

Correct

$u^h \leftarrow u^h + e^h$

Residual

$r^h = f^h - A^h u^h$

Restrict

$r^H = I^H_h r^h$

Interpolate

$e^h = I^h_H e^H$

Solve

$A^H e^H = r^H$

by recursion

...
Parallel High Performance FE Multigrid

- Parallelize "plain vanilla" multigrid
  - partition domain
  - parallelize all operations on all grids
  - use clever data structures

- Do not worry (so much) about Coarse Grids
  - idle processors?
  - short messages?
  - sequential dependency in grid hierarchy?

- Multigrid vs. Domain Decomposition
  - DD without coarse grid does not scale (algorithmically) and is inefficient for large problems/many processors
  - DD with coarse grids is like multigrid and is as difficult to parallelize
  - We get good results for parallel multigrid ...

Bey's Tetrahedral Refinement
HHG refinement example

Input Grid
HHG Refinement example

Refinement Level one
HHG Refinement example

Refinement Level Two
HHG Refinement example

Structured Interior
HHG Refinement example

Structured Interior
HHG Refinement example

Edge Interior
HHG Refinement example

Edge Interior
Parallel HHG - Framework
Design Goals

To realize good parallel scalability:

- Minimize latency by reducing the number of messages that must be sent
- Optimize for high bandwidth interconnects ⇒ large messages
- Avoid local copying into MPI buffers
HHG for Parallelization

Use regular HHG patches for partitioning the domain
HHG Parallel Update Algorithm

for each vertex do
    apply operation to vertex
end for

update vertex primary dependencies

for each edge do
    copy from vertex interior
    apply operation to edge
    copy to vertex halo
end for

update edge primary dependencies

for each element do
    copy from edge/vertex interiors
    apply operation to element
    copy to edge/vertex halos
end for

update secondary dependencies
<table>
<thead>
<tr>
<th>#Proc</th>
<th>#unkn. x 10^6</th>
<th>Ph.1: sec</th>
<th>Ph.2: sec</th>
<th>Time to sol.</th>
</tr>
</thead>
<tbody>
<tr>
<td>23</td>
<td>268.4</td>
<td>3.27</td>
<td>6.67*</td>
<td>39.3</td>
</tr>
<tr>
<td>6</td>
<td>536.9</td>
<td>3.35</td>
<td>6.75*</td>
<td>40.3</td>
</tr>
<tr>
<td>16</td>
<td>1,073.7</td>
<td>3.38</td>
<td>6.80*</td>
<td>40.6</td>
</tr>
<tr>
<td>64</td>
<td>2,147.5</td>
<td>3.53</td>
<td>4.92</td>
<td>42.3</td>
</tr>
<tr>
<td>128</td>
<td>4,295.0</td>
<td>3.60</td>
<td>7.06*</td>
<td>43.2</td>
</tr>
<tr>
<td>252</td>
<td>8,455.7</td>
<td>3.87</td>
<td>7.39*</td>
<td>46.4</td>
</tr>
<tr>
<td>504</td>
<td>16,911.4</td>
<td>3.96</td>
<td>5.44</td>
<td>47.6</td>
</tr>
<tr>
<td>2040</td>
<td>68,451.0</td>
<td>4.92</td>
<td>5.60</td>
<td>59.0</td>
</tr>
<tr>
<td>3825</td>
<td>128,345.7</td>
<td>6.90</td>
<td></td>
<td>82.8</td>
</tr>
<tr>
<td>4080</td>
<td>136,902.0</td>
<td></td>
<td>5.68</td>
<td></td>
</tr>
<tr>
<td>6102</td>
<td>205,353.1</td>
<td></td>
<td>6.33</td>
<td></td>
</tr>
<tr>
<td>8152</td>
<td>273,535.7</td>
<td></td>
<td>7.43*</td>
<td></td>
</tr>
<tr>
<td>9170</td>
<td>307,694.1</td>
<td></td>
<td>7.75*</td>
<td></td>
</tr>
</tbody>
</table>

Parallel scalability of scalar elliptic problem in 3D discretized by tetrahedral finite elements.

Times to solution on SGI Altix: Itanium-2 1.6 GHz.

Largest problem solved to date: 3.07 x 10^{11} DOFS (1.8 trillion tetrahedra) on 9170 Procs in roughly 90 secs.

B. Bergen, F. Hülsemann, U. Rüde, G. Wellein: ISC Award 2006, also: „Is 1.7 x 10^{10} unknowns the largest finite element system that can be solved today?“, SuperComputing, Nov’ 2005.
Computational Fluid Dynamics with the Lattice Boltzmann Method

Falling Drop with Turbulence Model (slow motion)
The Lattice-Boltzmann-Method

- Discretization in squares or cubes (cells)
- 9 numbers per cell (or 19 in 3D)
  = number of particles traveling towards neighboring cells
- Repeat (many times)
  - stream
  - collide
The stream step

Move particle (numbers) into neighboring cells
The collide step

Compute new particle numbers according to the collisions
LBM in Equations

Stream/Collide:

\[ F_i(x + c_i \Delta t, t + \Delta t) - F_i(x, t) = -\frac{1}{\tau} \left( F_i(x, t) - F_i^{(0)}(x, t) \right) \]

Equilibrium DF:

\[ F_i^{(0)}(x, t) = \frac{1}{3} \rho(x, t) \left( 1 - \frac{3}{2} \frac{\langle u(x, t), u(x, t) \rangle}{c^2} \right) \]

for \( i = C \),

\[ F_i^{(0)}(x, t) = \frac{1}{18} \rho(x, t) \left( 1 + 3 \frac{\langle c_i, u(x, t) \rangle}{c^2} + \frac{9}{2} \frac{\langle c_i, u(x, t) \rangle^2}{c^4} - \frac{3}{2} \frac{\langle u(x, t), u(x, t) \rangle}{c^2} \right) \]

for \( i \in \{N, E, S, W, T, B\} \),

\[ F_i^{(0)}(x, t) = \frac{1}{36} \rho(x, t) \left( 1 + 3 \frac{\langle c_i, u(x, t) \rangle}{c^2} + \frac{9}{2} \frac{\langle c_i, u(x, t) \rangle^2}{c^4} - \frac{3}{2} \frac{\langle u(x, t), u(x, t) \rangle}{c^2} \right) \]

for \( i \in \{TN, TS, BN, BS, TE, TW, BE, BW, NE, NW, SE, SW\} \).
Rigid Multibody Dynamics
What is rigid body dynamics?
**Weak Scaling**
up to 9120 processor cores
more than one billion geometric objects

**HLRB-II: SGI Altix**
Leibniz Computing Center Garching

Itanium based
63 TFlop Peak
40 TByte memory

**K. Iglberger (LSS)**
PRACE Award 2010

<table>
<thead>
<tr>
<th># Cores</th>
<th># Spheres</th>
<th>Partitioning</th>
<th>Runtime [s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>128</td>
<td>16,000,000</td>
<td>$128 \times 1 \times 1$</td>
<td>1059.73</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$16 \times 8 \times 1$</td>
<td>1139.79</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$8 \times 4 \times 4$</td>
<td>1400.87</td>
</tr>
<tr>
<td>256</td>
<td>32,000,000</td>
<td>$256 \times 1 \times 1$</td>
<td>1056.23</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$16 \times 16 \times 1$</td>
<td>1145.34</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$8 \times 8 \times 4$</td>
<td>1406.18</td>
</tr>
<tr>
<td>448</td>
<td>56,000,000</td>
<td>$448 \times 1 \times 1$</td>
<td>1049.47</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$28 \times 16 \times 1$</td>
<td>1141.81</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$8 \times 8 \times 7$</td>
<td>1405.81</td>
</tr>
<tr>
<td>960</td>
<td>120,000,000</td>
<td>$960 \times 1 \times 1$</td>
<td>1051.76</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$32 \times 30 \times 1$</td>
<td>1148.84</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$15 \times 8 \times 8$</td>
<td>1408.58</td>
</tr>
<tr>
<td>1920</td>
<td>240,000,000</td>
<td>$1920 \times 1 \times 1$</td>
<td>1065.06</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$60 \times 32 \times 1$</td>
<td>1157.27</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$16 \times 15 \times 8$</td>
<td>1414.46</td>
</tr>
<tr>
<td>3840</td>
<td>480,000,000</td>
<td>$3840 \times 1 \times 1$</td>
<td>1055.39</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$64 \times 60 \times 1$</td>
<td>1157.28</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$16 \times 16 \times 15$</td>
<td>1418.86</td>
</tr>
<tr>
<td>9120</td>
<td>1,140,000,000</td>
<td>$9120 \times 1 \times 1$</td>
<td>1066.73</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$96 \times 95 \times 1$</td>
<td>1359.29</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$24 \times 20 \times 19$</td>
<td>1613.73</td>
</tr>
</tbody>
</table>
Massively Parallel Particulate Flows
Mapping Moving Obstacles into the LBM Fluid Grid

(a) Initial setup: The velocities $u$ of the object cells $x_b$ are set to the velocity $u_w(x_b)$ of the object. In this example the object only has a translational velocity component. Fluid cells are marked with $x_f$.

(b) Updated setup: Two fluid cells have to be transformed to object cells and for two object cells the PDFs have to be reconstructed.

Figure 1: 2D mapping example.
Fluid-Structure Interaction

- Physics Engine
- Walbera

- Collision detection
- (frictional) collision response
- Time integration
- Rigid bodies act as obstacles
- Fluid results in external forces
- Update of fluid nodes: stream/collide
- Calculation of hydrodynamic forces (momentum exchange)
Algorithm 2  Coupled LBM-PE solver

1: *MPI communicate ghost layer of velocity and density*
2: for each body $B$ do
3:   Map $B$ to lattice grid
4: end for
5: *MPI communicate ghost layer of PDFs*
6: for each lattice cell do
7:   Stream and collide
8: end for
9: for each surface cell do
10:   Add forces from fluid to rigid objects
11: end for
12: *Time step in the rigid body simulation*
Parallelization of Particle-laden Flows

<table>
<thead>
<tr>
<th>Module</th>
<th>% of compute time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object mapping</td>
<td>13.0</td>
</tr>
<tr>
<td>LBM solver</td>
<td>53.2</td>
</tr>
<tr>
<td>Force calculation</td>
<td>15.6</td>
</tr>
<tr>
<td>PE solver</td>
<td>1.1</td>
</tr>
<tr>
<td>LBM communication</td>
<td>3.2</td>
</tr>
<tr>
<td>PE communication</td>
<td>14.0</td>
</tr>
</tbody>
</table>

xsize: 540, ysize: 500, zsize: 500

135 x 10^6 lattice cells

129 processes/cores

2,500 objects

27,000 time steps

12:54h
Coupling the physics engine to the Lattice Boltzmann solver

**Example**

1. Set boundary conditions
2. LBM stream collide
3. Add forces from fluid to obstacles
4. Move and collide obstacles
   - send to instance
   - move and collide locally
5. Move and collide obstacles on instance
6. Send values back
7. Receive and update values from instance
8. If obstacle getting near border, then send to neighbor
Fluidized Beds
Virtual Fluidized Bed

512 processors of HLRB2:
Size of Simulation Domain
400x400x480 cells of LBM
number of rigid objects: 25,000
number of time steps: 252,000
Run time:
30h 4min
corresponds to 15,000 core hours.
Virtual Fluidized Bed

512 Processors
HLRB-II

Simulation Domain
Size: 180x198x360 cells of LBM

900 capsules and 1008 spheres = 1908 objects

Number time steps: 252,000

Run Time: 07h 12 min
Parallel Performance of Fluid-Structure Interaction with Multibody Dynamics

Largest simulation to date: 622 Billion unknowns per time step (LBM alone) 12 TByte

HLRB-II
SGI Altix 4700
LRZ Garching

4 million fluid lattice cells per core
32 billion cells max in total
spherical moving objects of diameter 6 lattice cells
37 million moving objects max in total
Segregation simulation of 12,013 objects with two different shapes in different time steps simulated on 2,048 cores in a box. Density values of 0.8 kg/dm$^3$ and 1.2 kg/dm$^3$ are used for the objects in water with density 1 kg/dm$^3$ and a gravitation field. Lighter particles are rising to the top of the box, while heavier particle sink to the bottom.
Lattice Boltzmann Methods

Free Surface Flow Simulation

for foams, fuel cells, food processing, etc.
The interface between liquid and gas

- Volume-of-Fluids like approach
- Flag field: Compute only in fluid
- Special “free surface” conditions in interface cells
Simulation of Metal Foams

- Example application:
  - Engineering: metal foam simulations

- Based on LBM:
  - Free surfaces
  - Surface tension
  - Disjoining pressure to stabilize thin liquid films
  - Parallelization with MPI and load balancing

- Collaboration with C. Körner (Dept. of Material Sciences, Erlangen)

- Other applications:
  - Food processing
  - Fuel cells
Larger-Scale Computation: 1000 Bubbles

Simulation
1000 Bubbles
510x510x530 = $1.4 \times 10^8$ lattice cells
70,000 time steps
77 GB
64 processes
72 hours
4,608 core hours

Visualization
770 images
Approx. 12,000 core hours for rendering

Best Paper Award for Stefan Donath (LSS Erlangen) at ParCFD, May 2009 (Moffett Field, USA)
Numerical Experiment: Single Rising Bubble

Comparison to (rotationally symmetric) 2D level-set volume-of-fluid method and experimental results (T. Pohl, D. Gerlach, F. Durst (Erlangen), G. Biswas (IIT Kanpur), more in the future jointly with V. Buwa, IIT Delhi)

Modified parameter: surface tension
Flow Simulation

Visualization and Animation
Simulations with Fluid Control
Part IV

Conclusions
The Two Principles of Science

Three

Theory
Mathematical Models, Differential Equations, Newton

Experiments
Observation and prototypes, empirical Sciences

Computational Science
Simulation, Optimization, (quantitative) virtual Reality
# CE Graduate (Master) Program

## CE Master Degree

<table>
<thead>
<tr>
<th>Regular CE Master Studies</th>
<th>BGCE Elite program (with TU Munich)</th>
<th>Double Degree (with KTH Stockholm)</th>
<th>ERASMUS MUNDUS Program</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>COSSE</td>
</tr>
</tbody>
</table>


Double Degree Program

1st year

- Students beginning at Erlangen
- Students beginning in Stockholm

FAU Erlangen-Nürnberg

KTH Stockholm

Core Courses (taught at both universities)
- 15 ECTS Numerical Analysis
- 15 ECTS Applied Mathematics
- 15 ECTS Scientific Computing

Joint Workshop – in depth advising of students to prepare the transfer
- 15 ECTS Preparatory Courses for Specialization (partly given by Guest Prof. from KTH)
- 15 ECTS Preparatory Courses for Specialization (partly given by Guest Prof. from FAU)

Transfer

2nd year

Choice of Specialization in
- 15 ECTS Visualization and Image Processing
- 15 ECTS High Performance Computing
- 15 ECTS Electives

Choice of Specialization in
- 15 ECTS Bio-Modelling
- 15 ECTS Computational Fluid Dynamics
- 15 ECTS Electives

30 ECTS Master Thesis
conducted at host university
co-supervised by instructor from home-university
Frauen: 96 + 11*)
Männer: 256 + 29*)

Nordamerika:
- Kanada 1
- USA 4

Westeuropa:
- Deutschland 40 + 12
- Frankreich 1
- Griechenland 2
- Irland 1
- Österreich 1
- Spanien 2 + 1
- Italien 1

Osteuropa / Zentralasien:
- Bulgarien 14
- Estland 1
- Kasachstan 1
- Rumänien 4 + 2
- Russland 8 + 1
- Serbien 2
- Ukraine 10
- Ungarn 9
- Usbekistan 1

Lateinamerika:
- Bolivien 1 + 1
- Brasilien 4
- Costa Rica 1
- Ecuador 1
- Kolumbien 4
- Kuba 1
- Mexiko 3
- Nicaragua 1
- Peru 1
- Venezuela 1
- Chile + 1

Afrika:
- Ägypten 4
- Algerien 1
- Eritrea 1
- Gambia 1
- Kenia 1
- Kamerun 4
- Mauritius 1
- Senegal 1
- Sudan 1
- Tunesien 2
- Zimbabwe 1
- Uganda 1

Naher Osten:
- Iran 2 + 2
- Israel 1
- Jordanien 2
- Libanon 5 + 1
- Syrien 7 + 1
- Türkei 15 + 1

Südostasien:
- Indonesien 4
- Philippinen 1
- Thailand 1
- Vietnam 2

Südostasien:
- Bangladesch 1
- Indien 64 + 7
- Nepal 1
- Pakistan 10 + 3

Ostasien:
- China 94 + 6
- Südkorea 1

Σ = 352 + 40*)
aus 56 Ländern

*) Started in 2009

International CE Students 1997-2009
Examples of CE (Master) Thesis

- B. Gmeiner: Extension of a Software Package for Hierarchical Hybrid Grids
- S. Strobl: GPU-Based Rigid Body Dynamics
- J. Halwai: Performance Analysis of Hybrid Parallelization Techniques for Large Scale Lattice Boltzmann Simulations on Multicore Platforms
- T. Heller: SSE Optimierung of the PE Physics Engine
- D. Ritter: A Fast Multigrid Solver for Molecular Dynamics on the Cell Broadband Engine
- L. Yi: Efficient GPU Implementation of an MIR PPA Reconstruction Algorithm
- D. Bartuschat: A parallel patch-based approach for the reduction of quantum noise in CT-images
- D. Brinkers: Algorithms for sparse matrices on the CBEA
- Y. Sun: Parallel Solution for Differential-Algebraic Equations in power plant simulation
- S. Geißelsöder: Shared memory parallelization of the pe physics engine
- J. Habich: Performance Evaluation of Numeric Compute Kernels on NVIDIA GPUs
- M. Kavasoglu: Simulating Transformer Noise
- Yuanjun Zhang: Numerical Methods for Simulating Transformer Noise
Acknowledgements

Collaborators

- In Erlangen: WTM, LSE, LSTM, LGDV, RRZE, LME, Neurozentrum, Radiologie, Applied Mathematics, Theoretical Physics, etc.
- Especially for foams: C. Körner (WTM)
- International: Utah, Technion, Constanta, Ghent, Boulder, München, Zürich, Delhi, ...

Dissertationen Projects

- N. Thürey, T. Pohl, S. Donath, S. Bogner (LBM, free surfaces, 2-phase flows)
- M. Kowarschik, J. Treibig, M. Stürmer, J. Habich (architecture aware algorithms)
- K. Iglberger, T. Precklik, K. Pickel (rigid body dynamics)
- J. Götz, C. Feichtinger (Massively parallel LBM software, suspensions)
- C. Mihoubi, D. Bartuschat (Complex geometries, parallel LBM)

(Long Term) Guests in summer/fall 2009/10:

- Dr. S. Ganguly, IIT Kharagpur (Humboldt) - Electroosmotic Flows
- Prof. V. Buwa, IIT Delhi (Humboldt) - Gas-Fluid-Solid flows
- Felipe Aristizabal, McGill Univ., Canada (LBM with Brownian Motion)
- Prof. Popa, Constanta, Romania (DAAD) Numerical Linear Algebra
- Prof. N. Zakaria, Universiti Petronas, Malaysia
- Prof. Hanke, KTH Stockholm (DAAD), Mathematical Modelling

~25 Diplom- /Master- Thesis, ~30 Bachelor Thesis

Funding by KONWIHR, DFG, BMBF, EU, Elitenetzwerk Bayern
Thank you for your attention!

Questions?

Slides, reports, thesis, animations available for download at:
www10.informatik.uni-erlangen.de