Fast Parallel Algorithms for Computational Bio-Medicine

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

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

25. Juni 2009
Overview

- Intro: Who we are
- Example 1: Towards Near Real Time Simulation of Blood Flow on the IBM Cell Processor
- Example 2: Parallel Numerical Algorithms for Advanced Image Processing
- Conclusions
Introduction
The **LSS Mission**

Development and Analysis of Computer Methods for Applications in Science and Engineering

Applications from Physical and Engineering Sciences

**LSS**

Computer Science

Mathematics
What We Do

Simulation Applications in Science and Engineering
- Material Sciences
- Bio-medical Models and Simulation
- Chemical Engineering
- Computational Fluid Dynamics
- Electrical Engineering
- Power Systems
- Medical Imaging

Algorithms for Numerical Simulation
- Parallel Iterative Solvers (Multigrid)
- Parallel Lattice Boltzmann Methods

High Performance Computing
- Supercomputer Architecture
- Architecture-Aware and Parallel Programming, Cache Optimization
- Software Engineering for Technical Applications
Who is at LSS (and does what?)

**Complex Flows**
K. Iglberger
- C. Feichtinger
- S. Donath
- C. Mihoubi
- P. Neumann
- D. Geuss

**Numerical Algorithms**
H. Köstler
- T. Dreher
- T. Preclik
- D. Bartuschat
- ★ T. Preclik
- ★ C. Popa

**Laser Simulation**
Prof. Dr. C. Pflaum
- B. Heumann
- M. Wohlmuth
- ★ T. Dreher
- ★ T. Preclik
- ★ D. Bartuschat
- ★ C. Popa

**Supercomputing**
J. Götz
- T. Gradl
- M. Stürmer
- F. Deserno
- D. Ritter

**Alumni**
- Prof. G. Horton (Univ. of Magdeburg)
- Prof. El Mostafa Kalmoun (Cadi Ayyad University, Marocco)
- Dr. M. Kowarschik (Siemens Health Care)
- Dr. M Mohr (Geophysik, TU München)
- Dr. F. Hülsemann (EDF, Paris)
- Dr. B. Bergen (Los Alamos, USA)
- Dr. N. Thürey (ETZH Zürich)
- Dr. J. Härdtlein (Bosch GmbH)
- C. Möller (Navigon)
- Dr. U. Fabricius (Elektrobit)
- Dr. Th. Pohl (Siemens Health Care)
- J. Treibig (RRZE)
- C. Freundl
Industrial Collaboration Projects

- With Siemens
  - Industry
    - Dissertation T. Dreher (Siemens Simulation Center)
    - Temperature Simulation for Rolling Process Control
  - Energy
    - MSc Thesis Yongqi Sun (Power Plant Modelling)
  - Corporate Technology
    - MSc Thesis Rishi Patil (Coal Gasification)
  - Health Care
    - Dr. Campagna (MR reconstruction with GPUs)
    - Dr. Pohl (GPU accelerated image processing)
    - Dr. Kowarschik (Component Evaluation)

- Other industrial collaborations
  - Areva (Thermohydraulics)
  - Opel (Fuel Cells)
  - BASF (materials processing)
Example 1

Towards Near Real Time Flow Simulation on the IBM Cell for Bio-Medical Applications
**Computing Equipment**

**LSS Cluster**
- 8x4 (+ 9x2) Nodes
- CPU: AMD Opteron 848
- RAM: 8x16 Gbyte
- High-Speed Network InfiniBand

**Erlangen Computing Center (RRZE) Cluster**
- 217 compute nodes
- each with two Xeon 5160 "Woodcrest" chips (4 cores)
- High-Speed Network InfiniBand

**Bavarian Academy of Sciences (LRZ) HLRB-II Supercomputer**
- SGI Altix 4700
- 9728 Itanium Cores, 62 TFlops
- 40 Tbyte memory
Simulation chain

Patient → Angiography → 3D data → Simulation → Interpretation
Is Flow Simulation Possible in Real Time?

... at acceptable cost?

not today!

But ....

- Commercial CFD Software has tremendous overheads
- Grid generation/ interface to geometry description
- General purpose vs specialised solution
- Existing CFD software does not exploit modern (multi-core hardware)
- 4-D data sets (moving geometry)?
IBM Cell Processor

- Available cell systems:
  - Roadrunner
  - Blades
  - Playstation 3
Cell Architecture: 9 cores on a chip

- **Element Interconnect Bus**
  - 4 rings
  - up to 204.8 GB/s

- **PPU**
  - Power Processor Unit
  - simplified PowerPC
  - similar to PPC970
  - 2x SMT, but in order

- **MIC**

- **EIB**

- **BEI**
  - Broadband Engine Interface
  - coherent connection to 2nd CPU
  - IO devices

- **SPU**

- **Synergistic Processor Unit**
  - Memory Flow Controller
    - DMA transfer into and out of LS
    - channel interface
    - "interface controller" of the SPU
  - Local Store
    - "main memory" of SPE
    - 256 kB only
    - 16 B/s bandwidth
    - 128 B/s if all 8 banks accessed concurrently

- **Memory Interface Controller**
  - Rambus eXTreme Data Rate
  - 2 channels
  - 25.6 GB/s bandwidth

- **Synergistic Processor Element**
  - 128 SIMD registers (128 bit)
  - two SIMD pipelines
  - 204.8 GFlop/s max. for 8 SPEs

1 @ 3.2GHz
The Lattice-Boltzmann-Method

- An alternative to Finite Volumes or Finite Elements
- Discretization in cubes (cells)
- 9 (or 19) numbers per cell
  - = number of particles traveling towards neighboring cells
- Repeat (many times)
  - stream
  - collide
The stream step

Move particle (numbers)
into neighboring cells

- a single cell at timestep $t$ after collision
- a single cell at timestep $t+1$ after streaming
- four cells at timestep $t$ after collision
- four cells at timestep $t+1$ after streaming
The collide step

Compute new particle numbers according to the collisions
Results

Velocity near the wall in an aneurysm

Oscillatory shear stress near the wall in an aneurysm
Simulation of clotting processes using non-Newtonian blood models

- Motivation
- Knowledge about the clotting of blood is important for many medical applications:
  - Diagnostics
  - Surgery
Aging Model for blood clotting

Simulation of clot formation within the aneurysm
Performance Results

LBM performance on a single core (8x8x8 channel flow)

- Xeon 5160: 10.4
- PPE: 4.8
- SPE*: 2.0

*on Local Store without DMA transfers

**straight-forward C code**

**SIMD-optimized assembly**

---

Friedrich-Alexander-Universität Erlangen-Nürnberg

LEHRSTUHL FÜR INFORMATIK 10 (SYSTEMSIMULATION)
Performance Results

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>42</td>
<td>81</td>
<td>93</td>
<td>94</td>
<td>94</td>
<td>95</td>
</tr>
</tbody>
</table>
Example 2

Parallel Multilevel Image Processing Algorithms
(H. Köstler)
What we work on

Goals:

- deal with variational models in medical image processing and computer vision
- develop a software framework for their efficient numerical treatment by multigrid methods
- use common parallel hardware and fast numerical algorithms in order to achieve real-time processing

Applications:

- image denoising, image inpainting, image compression, image segmentation
- optical flow, image registration, motion blur
Variational Model
- construct energy functional using variational calculus
- done by hand

Application
- denoising of CT images
- segmentation of images
- registration of images
- video coding
- ...

Computation
- (automatically) optimize parameters
- optimize algorithms (e.g. use multigrid)
- optimize implementation
- choose suitable hardware and programming language

Tuning
- choose model parameters
- minimize energy functional
- discretize model
- solve (non)linear system
- (often) done by matlab

Simplify real world using certain assumptions

Check requirements, e.g., real time

Check model assumptions

Merge by using suitable tools

Adapt model

Check requirements, e.g., real time

Check model assumptions
Image Denoising Result: 3D CT Volume

Figure: Noisy image slice of a 3D CT volume (data: Siemens AG, Healthcare Sector) and one slice of denoised image [MBK$^{+}07$].
Figure: Reference and template image of an abdominal scan (data: Prof. Dr. med. K.-U. Eckardt, Nephrologie und Hypertensiologie, Universitätsklinikum Erlangen).
Figure: Registration of abdominal scan using an elastic regularizer (data: Prof. Dr. med. K.-U. Eckardt, Nephrologie und Hypertensiologie, Universitätsklinikum Erlangen).
Image processing at LSS

- Variational models in image processing (non-rigid registration, etc.)
  - + some compressed sensing algorithms
- Fast (multilevel algorithms)
- Parallel implementations
- Unconventional Hardware
  - different GPU types (ATI vs. Nvidia)
  - Cell
  - systematic comparisons
  - hybrid programming
Conclusions

- Computational models of bio-medical systems as future tools for diagnosis and therapy planning
  - from research tools to products in clinical practice
- Advanced image processing algorithms based on parallel multilevel algorithms
- Multi-Core accelerators as cost-efficient performance booster