Parallel Computing and Many Body Problems

George Biros       Bastiaan Braams

Class info

- Office hours
  - G. Biros: Monday 4-6pm
  - B. Braams: Tuesday 4-6pm

- Class home page
  - http://cs.nyu.edu/courses/spring03/G22.2945-001/index.htm
Class info - Requirements

- Homeworks
  - Algorithm design
  - Shared memory programming
  - MPI programming
- Semester project

Class Info - Topics

- Parallel computing
  - Algorithmic primitives
  - Shared memory
  - Distributed memory
- Scientific computing
  - N-Body algorithms
  - Multigrid, FFT
  - Linear Algebra
- Statistical physics
  - Monte Carlo simulations
  - Ising model
Class info – computing

- Six 4 CPU Sun workstations
  Bionum{1,6}.cims.nyu.edu
- One 8 CPU SGI Origin,
  spectrum.cims.nyu.edu
- Two 4 CPU SGI Origin,
  {septum,stratum}.cims.nyu.edu

Introduction
Why parallel computing?
1. Emergence of Computational Science

- Science
  - Analysis
    - Restricted to model problems, simple geometries
  - Experiments
    - Expensive (crash worthiness, aerodynamics)
    - Impossible (Astrophysics, Earthquakes, Global climate)
    - Dangerous (Medical devices, Nuclear devices)
    - Difficult to reproduce

- New Venue: Computational Science
  - Direct modeling of physical phenomena for
    - Scientific discovery
    - Optimal design
    - Engineering and Industry

Why parallel computing?
2. Sequential computing is slow

- To get faster has to get smaller, but
- Physics limitations
- Parallel is faster by definition

- But tuning software to run fast in single CPU
- is very important
Earth simulator – Fastest (silicon-based) machine as of 2002, 40Tflops ~20,000 P4s

Von Neumann computing model

Memory, CPU, I/O

In practice several memory hierarchies
Rule: Large memory is slow
Small memory is fast
Parallelism

Basic Definitions

- **Speedup**
  - best sequential / time on \( p \) processors
- **Efficiency**
  - Speedup/\( p \) (\(< 1\))
- **Latency**
  - time to initiate communication channel
- **Bandwidth**
  - capacity of communication channel
**Efficiency**

- Algorithmic scalability (sequential complexity)
  - How the algorithm scales with the increasing problem size and fixed number of processors
- Architecture scalability (fixed size scalability)
  - How the algorithm scales with fixed problem size and fixed number of processors
- Overall scalability (iso-granular scalability)
  - Fixed grain size (work per processor). Both work and $p$ increase. The most important in applications

**Amdahl’s law**

- Sequential bottleneck will ruin scalability
- $s$ is the sequential part percentage on the overall work.

\[
E = \frac{1}{s + \frac{(1 - s)}{p}} \leq \frac{1}{s}
\]

- Fix: sequential part should be independent of problem size, and $s$ will decrease as problem becomes bigger.
Basic definitions - continued

- Coordination
  - Synchronous vs. Asynchronous

- Scalability
  - Number of processors

- Granularity
  - Single processor work

- Interconnection network ($p^2$ is too expensive)
  - Ring, Bus, Mesh, Torus, Star, Hypercube, Butterfly, Fat trees

- Memory
  - Registers, Cache (L1, L2, L3), RAM, Discs

Basic models

- Machine models
  - Single Instruction Single Data (SISD)
  - Data parallel (Vector) (SIMD)
  - Shared memory (SMP)
  - Distributed memory (MIMD, SPMD)

- Programming models
  - Compilers (HPF, HPC++)
  - Threads, OpenMP
  - Message Passing MPI, PVM

- Best platforms combine everything (SMP Clusters)
- "Efficient" software should combine OpenMP + MPI
Basic steps in writing programs

- Partition work
- Determine communication
- Agglomeration to number of available processors
- Map to processors

- Goals
  - Minimize communication
  - Maximize concurrency of communication
  - Minimize synchronizations
  - Overlap computation with communication
  - Load balance
  - Avoid Amdahl law (sequential part that scales with input size)

Basic work partitioning techniques

- Divide and conquer
  - Important applications in N-Body algorithms
- Pipelining
  - Overlapping similar computation phases
- Domain decomposition
  - Partition of work is based on input data
- Functional decomposition
  - Partition is based on computation
- Embarassingly parallel
  - Independent tasks are readily identified
Models for algorithm evaluation

- Work/Depth models
  - Vector, Language, Graphs
- PRAM (shared memory)
  - Access to memory takes unit time
  - Variants to support exclusive reads and writes
- BSP - Bulk, Synchronous, Parallel (distributed)
  - Local/Remote memory
  - Uniform times to access remote memory
- LogP
  - Latency, Overhead, Gap (communication bandwidth), Processors

Practical goals

- Numerical Algorithms must be
  - highly concurrent and straightforward to load balance
  - latency tolerant
  - cache friendly (temporal and spatial locality of reference)
  - highly scalable (in the sense of algorithm convergence)

- Goal for algorithmic scalability: fill up memory of arbitrarily large machines while preserving constant running times with respect to proportionally smaller problem on one processor
Importance of optimal algorithms

- M1 runs in $O(N^2)$ in 1 CPU
- M2 runs in $O(N)$ in 1 CPU

- On 1000 CPUs M1 solves problem of size 30 x N
- On 1000 CPUs M2 solves problem of size 1000 x N

Software engineering

- Programming is more difficult
  - Deadlocks
  - System level requirements
  - Check-pointing
- Unreliable OS
- Unreliable I/O
- Unreliable software
- Unreliable hardware
- Debugging is painful
Things have changed

- Robust platforms for “small” $p$
  - 1-256 CPU. Shared memory, SMPs and Beowulf clusters
  - MPI libraries *de facto* standard
  - Portability is possible
  - Debugging, performance monitoring, and software libraries readily available

---

Gordon Bell Prize winners

![Gordon Bell Prize winners graph](chart.png)
Bell prizes history

<table>
<thead>
<tr>
<th>Year</th>
<th>Type</th>
<th>Application</th>
<th>Gflop/s</th>
<th>System</th>
<th>No. Procs</th>
</tr>
</thead>
<tbody>
<tr>
<td>1988</td>
<td>PDE</td>
<td>Structures</td>
<td>1.0</td>
<td>Cray Y-MP</td>
<td>8</td>
</tr>
<tr>
<td>1989</td>
<td>PDE</td>
<td>Seismic</td>
<td>5.6</td>
<td>CM-2</td>
<td>2,048</td>
</tr>
<tr>
<td>1990</td>
<td>PDE</td>
<td>Seismic</td>
<td>14</td>
<td>CM-2</td>
<td>2,048</td>
</tr>
<tr>
<td>1992</td>
<td>NB</td>
<td>Gravitation</td>
<td>5.4</td>
<td>Delta</td>
<td>512</td>
</tr>
<tr>
<td>1993</td>
<td>MC</td>
<td>Boltzmann</td>
<td>60</td>
<td>CM-5</td>
<td>1,024</td>
</tr>
<tr>
<td>1994</td>
<td>IE</td>
<td>Structures</td>
<td>143</td>
<td>Paragon</td>
<td>1,904</td>
</tr>
<tr>
<td>1995</td>
<td>MC</td>
<td>QCD</td>
<td>179</td>
<td>NWT</td>
<td>128</td>
</tr>
<tr>
<td>1996</td>
<td>PDE</td>
<td>CFD</td>
<td>111</td>
<td>NWT</td>
<td>160</td>
</tr>
<tr>
<td>1997</td>
<td>NB</td>
<td>Gravitation</td>
<td>170</td>
<td>ASCI Red</td>
<td>4,096</td>
</tr>
<tr>
<td>1998</td>
<td>MD</td>
<td>Magnetism</td>
<td>1,020</td>
<td>T3E-1200</td>
<td>1,536</td>
</tr>
<tr>
<td>1999</td>
<td>PDE</td>
<td>CFD</td>
<td>627</td>
<td>ASCI BluePac</td>
<td>5,832</td>
</tr>
<tr>
<td>2000</td>
<td>NB</td>
<td>Gravitation</td>
<td>1,349</td>
<td>GRAPE-6</td>
<td>96</td>
</tr>
</tbody>
</table>

2002 Gordon Bell awards  [www.supercomp.org]

- Astrophysics* (26.5 Tflops)
  - GRAPE 6, Japan
- Structural mechanics (1.16 Tflops 3K CPUs)
  - Salinas, Sandia Labs
- Atmospheric simulations (26.5 Tflops 5K CPUs)
  - Earth simulator, Japan
- Turbulence simulations (16.4 Tflops)
  - Earth simulator, Japan
- Nuclear Fusion
  - Earth simulator, Japan (14.9 Tflops)
- Biomolecular simulation*
  - NAMD, UIUC

*N-body algorithm
Fastest academic computer in US (2002)

Summary
- 750 Compute Nodes
- 3000 EV68 processors
- 6 TF (peak; est ~4 TF on LISR)
- 3.0 TB memory
- 40 TB local disk (sys + tmp)
- Multi-rack fat-tree network
- Redundant monitor/control
- WAN/LAN accessible
- Parallel visualization
- File servers:
  - 50TB, ~32 GB/s
  - Mass store, ~1 TB/h

Control of flow around a Boeing 707 wing

Optimal control of laminar viscous flow
- Optimization variables are surface suction/injection
- Objective is minimum drag
- 700,000 states; 4,000 controls
- 128 Cray T3E processors
- ~5 hrs for optimal solution (~1 hr for analysis)

Suction/Injection control
Simulation of a 1994 Earthquake at Northridge

Efficiencies and required memory

- Largest current (complete) simulation:
  - 1Hz source takes 6h on 512PEs @ 231 Gflops/s
- Target simulation:
  - 2Hz, estimated 13.5h on 3000 PEs @ 1.2 Tflops/s
Unstructured grids and partitioning

Simulation, 100^6 elements
Cray T3E at PSC

www.cs.cmu.edu/~oghattas
N-Body Algorithms - Integral equations

N-Body algorithms – Integral equations
<table>
<thead>
<tr>
<th>Rank</th>
<th>Manufacturer/Computer System</th>
<th>Rmax</th>
<th>Installation Site</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>NEC Earth Simulator/5120</td>
<td>35868.00</td>
<td>Earth Simulator Center</td>
</tr>
<tr>
<td></td>
<td></td>
<td>40960.00</td>
<td>Japan/2002</td>
</tr>
<tr>
<td>2</td>
<td>Hewlett-Packard ASCI Q - AlphaServer SC ESP/1.25 GHz/4096</td>
<td>7727.00</td>
<td>Los Alamos National Laboratory</td>
</tr>
<tr>
<td></td>
<td></td>
<td>10240.00</td>
<td>USA/2002</td>
</tr>
<tr>
<td>3</td>
<td>Hewlett-Packard ASCI Q - AlphaServer SC ESP/1.25 GHz/4096</td>
<td>7727.00</td>
<td>Los Alamos National Laboratory</td>
</tr>
<tr>
<td></td>
<td></td>
<td>10240.00</td>
<td>USA/2002</td>
</tr>
<tr>
<td>4</td>
<td>IBM ASCI White, SP Power3 3.75 MHz/8192</td>
<td>7226.00</td>
<td>Lawrence Livermore National Laboratory</td>
</tr>
<tr>
<td></td>
<td></td>
<td>12288.00</td>
<td>USA/2002</td>
</tr>
<tr>
<td>5</td>
<td>Linux Network: PACE Linux Cluster: Xeon 2.4 GHz - Quadrics/2304</td>
<td>5084.00</td>
<td>Lawrence Livermore National Laboratory</td>
</tr>
<tr>
<td></td>
<td></td>
<td>10000.00</td>
<td>USA/2003</td>
</tr>
<tr>
<td>6</td>
<td>Hewlett-Packard AlphaServer SC ES45/1 GHz/3016</td>
<td>4463.00</td>
<td>Pittsburgh Supercomputing Center</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6032.00</td>
<td>USA/2001</td>
</tr>
<tr>
<td>7</td>
<td>Hewlett-Packard AlphaServer SC ES45/1 GHz/2560</td>
<td>3980.00</td>
<td>Commissariat à l'Energie Atomique (CEA)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5120.00</td>
<td>France/2001</td>
</tr>
<tr>
<td>8</td>
<td>HP Teraflop Xeon 2.2 GHz - MyriNet/2000/1536</td>
<td>3237.00</td>
<td>Forecast Systems Laboratory - NOAA</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6758.00</td>
<td>USA/2002</td>
</tr>
<tr>
<td>9</td>
<td>IBM pSeries 690 Turbo 1.3GHz/1280</td>
<td>3241.00</td>
<td>IBM/2001</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6036.00</td>
<td>UK/2002</td>
</tr>
<tr>
<td>10</td>
<td>IBM pSeries 690 Turbo 1.3GHz/1216</td>
<td>2164.00</td>
<td>NCAR (National Center for Atmospheric Research)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3122.00</td>
<td>USA/2002</td>
</tr>
</tbody>
</table>