Performance Characterization of a Molecular Dynamics Code on PC Clusters

Is there any easy parallelism in CHARMM?

M. Taufer, E. Perathoner, A. Cavalli, A. Caflisch, T. Stricker

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich
Motivation

• Demand for powerful and affordable computing platforms, such as cluster of PCs, for running molecular dynamics codes like CHARMM

• Biologists turn to computer architects for a platform recommendation

• Systematic method to find the most cost effective platform for CHARMM
Outline

• Scientific computation code CHARMM
• Systematic approach for:
  ▪ performance study of CHARMM on cluster of PCs
• Platform parameters which affect the application performance:
  ▪ Network technology
  ▪ Middleware
  ▪ Number of CPUs per node (SMPs)
• Quantify the performance impact of the chosen platform parameters on CHARMM on cluster of PCs
• Platform Recommendation
CHARMM – Chemistry at Harvard Macromolecular Mechanics

Use of CHARMM in:

- Molecular dynamics (MD) for protein folding simulation

Parallelization method:
- Replicated-Data (RD) method

Electrostatic interactions:
- Local interactions
- Long-range interactions

Nodes: p1  p2  p3  p4
CHARMM Classical Energy Calculation

Classical energy calculation is characterized by:

- Classical mechanics, i.e. Newton equations of motions
- Evaluation of the energy and its gradient (force) in an efficient way
- Accurate treatment of local electrostatic interactions
- Long-range electrostatic interactions treatment not accurate enough
Particle Mesh Ewald (PME)

Particle Mesh Ewald (PME) method:
- Enhanced energy calculation incorporating long-range electrostatic force calculations in frequency domain
- More accurate way for treating the long-range interactions
- Requires convolution on a interpolated grid using 3-D FFTs

3-D FFTs make data parallelism harder

How efficient is PME on clusters of PCs?
CHARMM on Different PC Clusters

• Systematic experimental design with different:
  ▪ Software configurations
  ▪ Hardware configurations

• Questions to be answered quantitatively:
  ▪ The benefits of data parallelism on different cluster platforms
  ▪ Scalability limits depending on cluster configuration

• Attempt to gather the maximum performance information with the minimum number of experiments:
  ▪ Response variables: measured performance of the system
  ▪ Factors: the parameters which affect the response variables
  ▪ Levels: the values of the factors
Performance of CHARMM: Response Variables

Classical energy calculation time: using classical mechanics in time domain for energy calculation

PME energy calculation time: additional computation in frequency domain and related FFTs due to PME approach

Time components measured for each of two energy calculations:

- Computation time
- Communication time
- Synchronization time
Characteristics of the Platforms:
Space of Factors

- **Network technology factor:**
  - **MPICH** using **TCP/IP** over Gigabit Ethernet
  - **SCORE** over Gigabit Ethernet
  - **MPICH-GM** over Myrinet

- **Middleware factor:**
  - MPI calls with point-to-point **blocking** communication
  - CHARMM MPI calls with **non-blocking** communication

- **Number of CPUs per node:**
  - Single CPU (**non-SMP**)
  - Dual CPUs (**SMP**)
Calculation Times at the Focal Point

**Total energy calculation:** MPI with TCP/IP on Ethernet (non-SMP)

**Total calculation time:**
- Classical energy calculation
- PME energy calculation

In sequential CHARMM:
- PME time slightly less than half of total calculation time

In parallel CHARMM:
- PME time becomes 2/3 of total calculation time

PME dominates performance with increasing number of nodes
Resource Usage at the Focal Point

Classical energy calculation: MPI with TCP/IP on Ethernet (non-SMP)

PME energy calculation: MPI with TCP/IP on Ethernet (non-SMP)

Communication overhead in PME explains inefficiency of parallel version
Performance Impact of Network Technology

Total energy calculation

Middleware

Number of CPUs per node

SMP

non-SMP

MPI

CMPI

Networking

Focal Point
Performance Impact of Middleware

Number of CPUs per node

SMP

non-SMP

MPI

CMPI

Networking

Focal Point

Total energy calculation

speed-up

slow-down

MPI

CMPI

number of processors

1 2 4 8

1 2 4 8

classic calculation

pme calculation

13
Resource Usage with CMPI

Classical energy calculation: CMPI with TCP/IP on Ethernet (non-SMP)

PME energy calculation: CMPI with TCP/IP on Ethernet (non-SMP)

CMPI synchronization is inefficient on clusters of PCs
Performance Impact of Multiprocessor Nodes (I)

Total energy calculation: MPI with TCP/IP on Ethernet

- speed-up
- slow-down

Focal Point

<table>
<thead>
<tr>
<th>number of processors</th>
<th>non-SMP</th>
<th>SMP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>classic calculation</td>
<td>pme calculation</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>8</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>1</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td>6</td>
<td>6</td>
</tr>
</tbody>
</table>
Performance Impact of Multiprocessor Nodes (II)

Total energy calculation: MPICH-GM on Myrinet

Focal Point
Platform Recommendation

CHARMM with classical energy calculation:
- Clusters with Ethernet and MPICH:
  - 4-8 nodes

CHARMM with PME energy calculation:
- Clusters with Ethernet and enhanced message passing:
  - 8-16 nodes
- Clusters with Myrinet (Score)
  - >16 nodes

- Clusters with Ethernet and enhanced message passing (Score) systems:
  - 16-32 nodes

- Problem of robustness and portability
  - High cost network
  - Low cost network

- Cannot take advantage from SMP nodes

CHOICE:
- Clusters with Ethernet and enhanced message passing:
Conclusion and Outlook

• Systematic approach to performance evaluation allows quantifying:
  - Scalability limits of CHARMM on commonly used cluster configurations due to resource usage

• Our systematic performance evaluation can be extended to other applications and architectures (e.g., the grid):