A CPU/MIC Collaborated Parallel Framework for GROMACS on Tianhe-2 Supercomputer

Wenhe Su, Shaoliang PENG, Shunyun Yang, Xiaoyu Zhang, Tengililang Zhang, Weiguo Liu, Xingming Zhao

Supported by: NSFC Grant 61272056, U1435222, and 1133005

National University of Defense Technology

Shandong University

Tongji University
About us

- School of Computer Science of NUDT
- Hometown of Supercomputers: Tianhe-2
  - No. 1 in TOP500 (2013 – 2015)
  - 33.86 PFLOPS, 32,000 CPUs+48,000 MICs
Outline

• Background
• Motivation & Challenge
• Design
• Evaluations
• Conclusions
Molecular Dynamics

Molecular dynamics (MD) is a computer simulation method for studying the **physical movements of atoms and molecules**.

The atoms and molecules are allowed to interact for a fixed period of time, giving a view of the dynamical evolution of the system.
GROMACS is a versatile package for performing molecular dynamics simulation systems with hundreds to millions of particles.

GROMACS is primarily designed for biochemical molecules like proteins, lipids, and nucleic acids that have many complicated bonded interactions.
A typical GROMACS MD run of a protein in a box of water
Motivation

• With the rapid advance of the pharmaceutical industry, there is a continuous demand for molecular dynamics, which is a key procedure in pharmaceutical research.

• In China, more than 20% of the computing resources in national supercomputer centers are used for GROMACS simulations.
Challenge

• Although GROMACS is highly efficient, it does not meet the rapidly increasing demand of molecular dynamics, owing to the limits of the algorithms and models.

• GROMACS need to simulate for a long period of time to get a valuable result, which always takes millions of iterations. As a consequence, it usually takes a few months to accomplish a simulation, which is intolerable.
Motivation & Challenge

CPU/MIC Collaborated Parallel GROMACS need to be developed
Design

• Transfer computing-intensive part of work onto MIC with offload mode
  • Decrease the workload on CPU, implement the offload-mode version of GROMACS
• Collaborated parallelization using CPU-MIC
  • Accelerating GROMACS, achieve workload balance
• Implemented the method on multiple computation nodes
  • Adapt to larger scale of data, acquire better scalability
Why offload mode

• There are three modes on MIC: native mode, symmetric mode, offload mode
• Native mode: only MIC works.
• Symmetric mode: bad allocation, low efficiency due to Intel’s dereliction.
• Offload mode: Flexible and more efficient
  • Need more artful design.
  • Extra overhead.
According to Amdahl’s law: Accelerate the process of none-bonded force calculation may bring us remarkable result.
Step1: Offload-mode acceleration

We carry on inter-atoms non-bonded force calculation, which takes most of time, on MIC to accelerate the process of simulation. We eliminate data dependency and solve problems such as write-collision and communication overhead to ensure the validity and achieve better performance.
Step1: Offload-mode acceleration

The **write-collision** occurs when two different threads attempt to modify one force in the list at the same time.

Our strategy is to **keep multiple copy of the data to increase reliability**: Each thread keeps one force_list copy and run a reduction function at the end of program to gather all force_lists.
Step1: Offload-mode acceleration

To decrease the communication overhead caused by keeping the redundancy, we reconstruct and trim the memory segment. Then transfer the continuous data to reduce the time cost.
Step2: Collaborated parallelization using CPU-MIC

The program may wait the slowest thread (process) to halt.

Balance the workload by experiment, which can optimize our method.
Step 3: Multiple-nodes acceleration

The task-partition cuts an atomic system into grids.
Step 3: Multiple-nodes acceleration

The bottleneck for multiple-nodes acceleration is the communication which may decrease the holistic performance and damage the scalability.

Create different flows for different types of data to decrease the communication overcome among nodes.
Evaluations: experiments setup

- Tianhe-2 supercomputer:

<table>
<thead>
<tr>
<th>Items</th>
<th>Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processors</td>
<td>32,000 Intel Xeon CPUs + 48,000 Xeon Phis + 4,096 FT CPUs</td>
</tr>
<tr>
<td></td>
<td>Peak performance is 54.9PFLOPS</td>
</tr>
<tr>
<td>Interconnect</td>
<td>Proprietary high-speed interconnection network TH Express-2</td>
</tr>
<tr>
<td>Memory</td>
<td>1PB in total</td>
</tr>
<tr>
<td>Storage</td>
<td>Global shared parallel storage system, 12.4PB</td>
</tr>
</tbody>
</table>
Improvement of CPU&MIC Collaboration

Speedup 1.51 to 1.82

The calculation time cost of the parallel new data flow is much less than that in the original version, which shows a good acceleration improvement.
Improvement of new data flow

Speedup: 1.59

The new data flow with MIC parallel acceleration has the best performance, which shows a good acceleration improvement.
Improvement of Multi-nodes

We set 2,000 steps and 100,000 atoms to test the performance of multi-node GROMACS.

The speed of accelerated version is greatly improved. Overall, compared to the original GROMACS, the accelerated version GROMACS achieves a significant acceleration effect.

The total speed-up is over 3.7
Conclusions

• Main contributions: three-level acceleration framework
  • Transfer computing-intensive part of work onto MIC with offload mode
  • Collaborated parallelization using CPU-MIC
  • Implemented the method on multiple computation nodes
• The source code is in https://github.com/tianhe2/gromacs-mic
• We ensured the acceleration effect and practicability by experiments carried out on the Tianhe2.
The Tianhe-2 supercomputer is available online. All the scientists can collaborate with us to develop new software and access Tianhe-2 using the Internet.

Welcome to contact us!

Email: pengshaoliang@nudt.edu.cn