Scalable and Highly SIMD-vectorized Molecular Dynamics Simulation Involving Multiple Bubble Nuclei

Hiroshi Watanabe[1]
hwatanabe@issp.u-tokyo.ac.jp

Satoshi Morita[1]
morita@issp.u-tokyo.ac.jp

Inaoka Hajime[2]
hajime.inaoka@riken.jp

Haruhiko Matsuo[3]
halm@rist.or.jp

Synge Todo[4,1]
wistaria@phys.s.u-tokyo.ac.jp

Nobuyasu Ito[5,2]
ito@ap.t.u-tokyo.ac.jp

[1] The Institute for Solid State Physics, The University of Tokyo
[2]RIKEN Advanced Institute for Computational Science
[3]Research Organization for Information Science and Technology
[4]Department of Physics, School of Science, The University of Tokyo
[5]Department of Applied Physics, School of Engineering, The University of Tokyo

ABSTRACT

While understanding the behaviors of gas-liquid multiphase systems is important for the field of engineering, the simulation of systems involving multiple bubble nuclei is challenging since (1) the system involves travels, creations and annihilations of phase boundaries, and (2) a huge system is required in order to investigate the interactions between bubbles. We overcame the above difficulties by implementing a scalable molecular dynamics code adopting hybrid parallelism which allows us to perform simulations involving 13.7 billion Lennard-Jones atoms on the full nodes of the K computer. The simulation code is highly tuned to SPARC architecture and 98.8% of arithmetic operations are SIMDized. A performance of 2.44 PFlops representing 23.0% of peak is achieved. We present an unprecedented simulation of cavitation involving multiple bubble nuclei. The detailed behavior of bubbles in the early stage of cavitation is captured for the first time.

Keywords

Time-to-solution, scalability, molecular dynamics simulations, cavitation, bubble nucleation, Ostwald ripening

1. BACKGROUND

Understanding the behavior of bubbles is difficult, since it involves various kinds of physics such as phase transitions, heat conduction, interaction between bubbles, and so forth. Especially, the very moment of the multiple bubble nuclei is veiled in mystery. In order to tackle this problem, we adopt the molecular dynamics method (MD). With MD, we can investigate nuclei and interactions of bubbles from the atomic scale. However, a huge number of atoms are required. 10,000 atoms are required to express a bubble and 10,000 bubbles are required to investigate their interactions. As the result, a simulation should involve at least a hundreds of millions of atoms. Since it is impossible on a single computer, parallelization is required. In developing parallelized MD, a programer will face many dilemmas. To obtain better performance on the recent computers, SIMD-vectorization is unavoidable. While it may be possible to perform SIMD-vectorization and thread-parallelization simultaneously, the code would be excessively complex. IO is also troublesome since a huge run involves huge IO. An appropriate compression of data is required which drastically reduces the size of a file keeping the important information. In the present poster, we propose a simple and efficient implementation which achieves scalability and computational efficiency simultaneously [4]. Data compression method is also described.

2. MOLECULAR DYNAMICS METHOD

We performs MD simulation with the truncated LJ potential [2]. The list of neighboring atoms is used to reduce the computational cost to search interacting atoms. Since the construction of the list is expensive, we adopt the well-known bookkeeping technique [3] in order to reuse the same list for several time steps. We apply the bookkeeping method for parallel computations [5].

3. SPECIFIC OPTIMIZATION FOR THE K COMPUTER

The K computer consists of 82,944 nodes (the total number of core is 665,552), and its theoretical peak performance is 10.6 PFlops. In order to increase the computational efficiency, we write the force calculation kernel explicitly by intrinsic functions. Since the floating-point divide instruction (`fdivd`) cannot be specified as a SIMD instruction, we use the floating-point reciprocal approximation (RCPA) instruction (`frcpad`) with corrections of accuracy. With the optimization described above, 98.8% of the arithmetic operations, which are 69.5% of the total executed instructions, are SIMDized.

4. PSEUDO-FLAT MPI APPROACH

	SISD	FMA	SIMD	SIMD-FMA	Others
all	0.84	0.46	48.4	19.8	30.5
force	1.66	0.84	44.2	18.1	35.2

Table 1: Breakdowns of instructions for the single-node run (percentages).

We adopt a pseudo-flat MPI approach, *i.e.*, we use threads as if they are processes by using thread IDs as ranks of the MPI [1]. A domain assigned to a node (process) is divided into subdomains and a subdomain is assigned to each CPU-core (thread). While the pseudo-flat MPI approach increases the complexity of communications, we do not have to take into account load imbalance of threads and SIMDization simultaneously.

5. SCALABILITY ON THE K COMPUTER

Weak scaling performance for 154 thousand atoms per node was measured on the K computer. Elapsed time for 1000 steps (open symbols) and memory usage per node (filled symbols) are shown. The largest run involves 12.8 billion atoms. While the runs of flat-MPI are always faster than those of hybrid, memory usage increases drastically when the number of node is 32768.

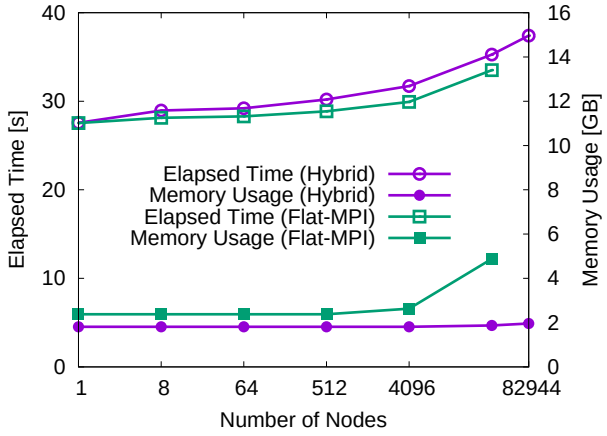

Figure 1: Weak scaling performance on the K computer.

6. MULTIPLE BUBBLE NUCLEI SIMULATION

We perform a huge simulation of cavitation involving 13.7 billion atoms on the full nodes of the K computer. The size of the system box is $2880 \times 3240 \times 1920$ corresponding to $1.0\ \mu\text{m} \times 1.1\ \mu\text{m} \times 0.6\ \mu\text{m}$. The performance of the whole run including DATA IO is 1.81 PFlops which is 17.0% of peak. We also perform data compress of the data in order to reduce the time for DATA IO [6].

7. CONCLUSIONS

We have developed MD code which works effectively on the K computer. Large scale MD simulations allows us to capture the behavior of bubbles in the very moment of nucleation which will offers an important insight for developing more efficient turbines in power plants or screw propellers of ships.

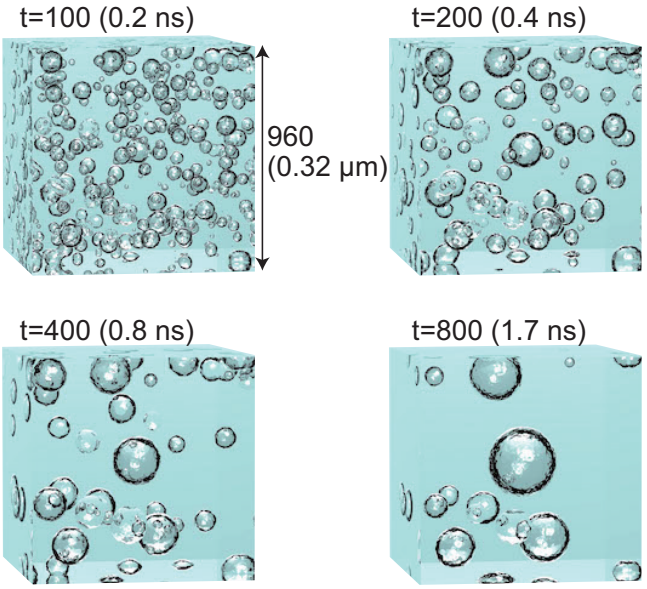

Figure 2: Coarsening of bubbles. After sudden decrease of pressure, many bubbles appear and coarsening of them follows. A small system calculated on 4096 nodes of the K computer is shown for visibility.

8. ACKNOWLEDGMENTS

The computation was carried out on the K computer at the RIKEN AICS (proposal number hp140214, hp130047, and hp120305). We would like to thank N. Kawashima for fruitful discussions and to K. Nitadori and N. Yoshioka for helpful comments, and the help desk of HPCI and RIST for technical supports. This work was partially supported by Grants-in-Aid for ScientificResearch (Contract No. 23740287).

9. REFERENCES

[1] Berger, M. J., Aftosmis, M. J., Marshall, D. D., and Murman, S. M. Performance of a New CFD Flow Solver using a Hybrid Programming Paradigm. *Journal of Parallel and Distributed Computing*, 65(4):414–423, 2005.

[2] S. D. Stoddard and J. Ford. Numerical experiments on the stochastic behavior of a lennard-jones gas system. *Phys. Rev. A*, 8:1504–1512, 1973.

[3] L. Verlet. Computer "experiments" on classical fluids. i. thermodynamical properties of lennard-jones molecules. *Phys. Rev.*, 159:98–103, 1967.

[4] H. Watanabe, M. Suzuki, and N. Ito. `http://mdacp.sourceforge.net/`.

[5] H. Watanabe, M. Suzuki, and N. Ito. Efficient implementations of molecular dynamics simulations for lennard-jones systems. *Progress of Theoretical Physics*, 126(2):203–235, 2011.

[6] H. Watanabe, M. Suzuki, and N. Ito. Huge-scale molecular dynamics simulation of multibubble nuclei. *Comput. Phys. Commun.*, 184(12):2775 – 2784, 2013.