Huge-Scale Molecular Dynamics Simulation of Multi-bubble Nuclei

H. Watanabe ISSP, The University of Tokyo

M. Suzuki Kyushu University

H. Inaoka RIKEN AICS

N. Ito The University of Tokyo, RIKEN AICS

Outline

1. Introduction
2. Benchmark results
3. Cavitation
4. Future of HPC
5. Summary and Discussion

May. 15th, 2014@NCSA Blue Waters Symposium

University of Tokyo

Introduction (1/2)

Gas-Liquid Multi-Phase Flow

Multi-Scale and Multi-Physics Problem

Numerical Difficulties

Moving, Annihilation and Creation of Boundary

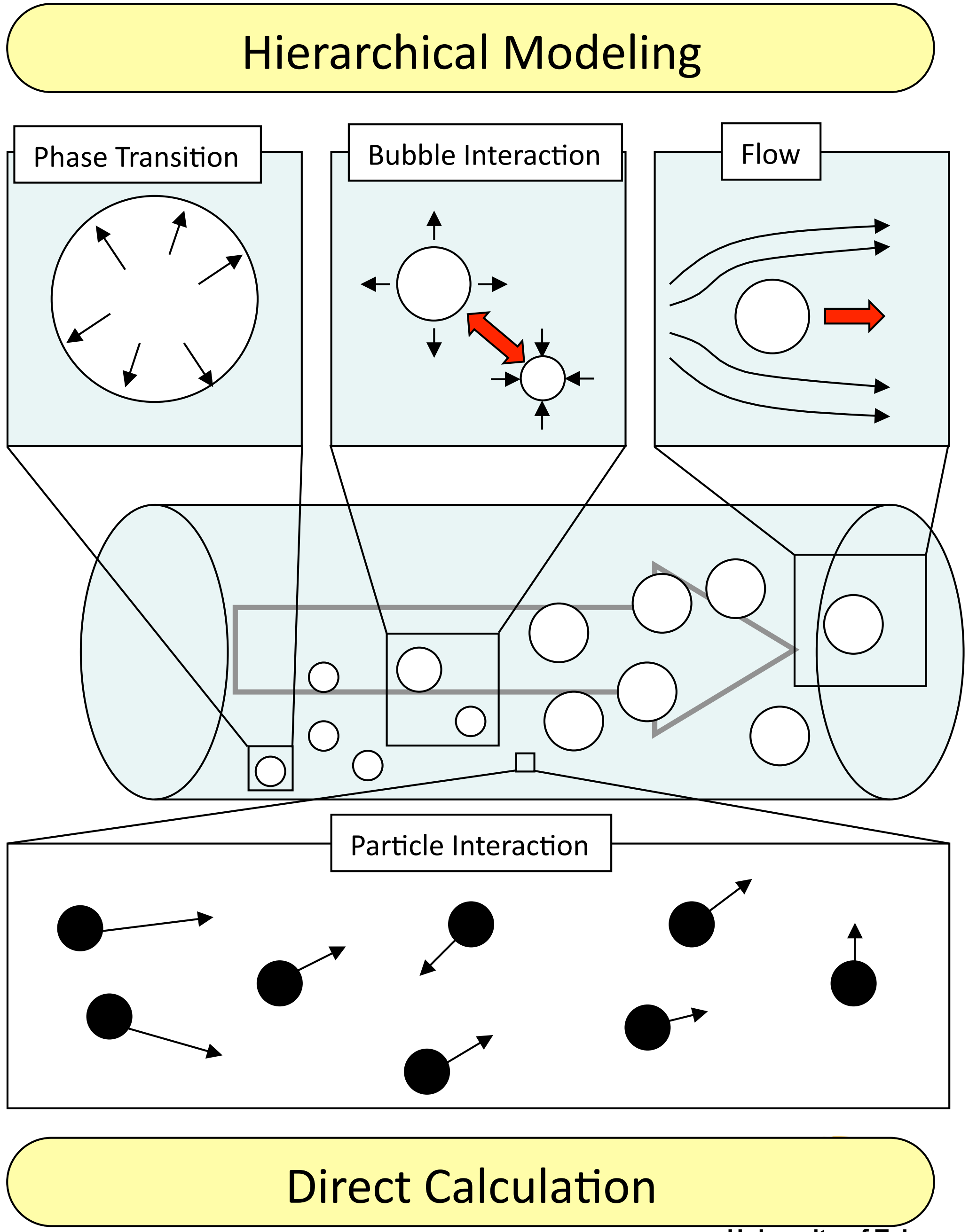
Hierarchical Modeling

Divide a problem into sub problems
Governing Equations for each scale

Artificial Hierarchy
Validity of Governing Equation

→ Direct Simulation

Movie



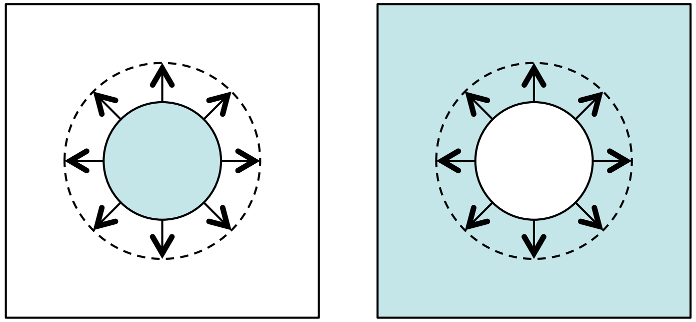
Introduction (2/2)

Classical Nucleation Theory

Classical nucleation theory (CNT) predicts a nucleation rate of clusters.
OK for droplet nuclei, bad for bubble nuclei.

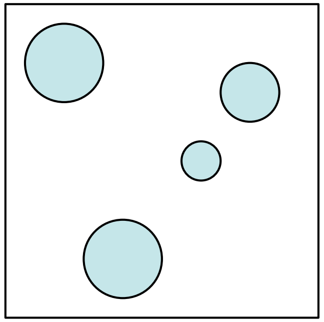
Droplets and Bubbles

Additional work to create a bubble

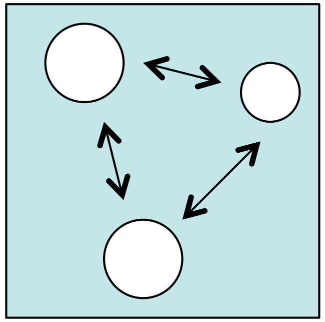


$$W = \underbrace{\gamma A - V\rho\Delta\mu}_{\text{Same as Droplet}} - \underbrace{V\Delta P}_{\text{Work by Bubble}}$$

Interaction between clusters



Interact via gas (diffusive)



Interact via liquid (ballistic)

It is difficult to investigate microscopic behavior directly.

Huge scale molecular dynamics simulation
to observe interaction between bubbles.

Parallelization (1/3)

Simulation Method

Molecular Dynamics (MD) method with Short-range Interaction

General Approach

MPI: Domain Decomposition
OpenMP: Loop Decomposition

```
DO I=1,N          <------------ Loop Decomposition here
  DO J in Pair(I)   <-------- or here
    CalcForce(I,J)
  ENDDO
ENDDO
```

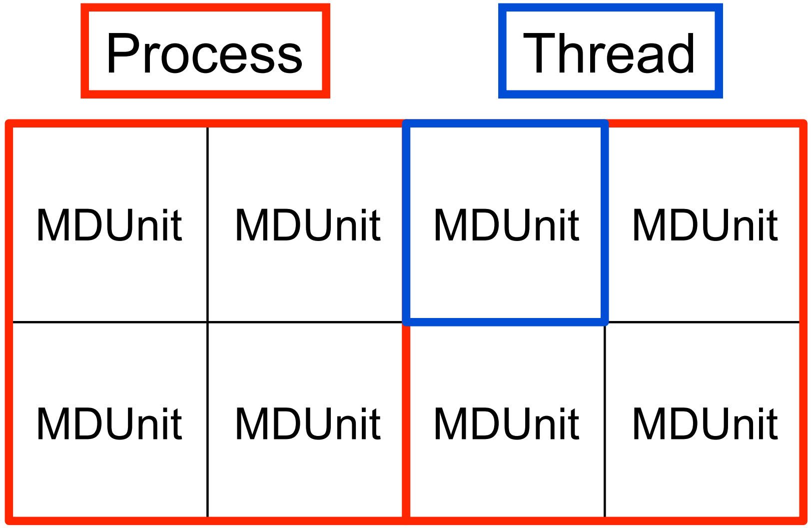
Problems

Conflicts on write back of momenta
→ Prepare temporary buffer to avoid conflicts
→ Do not use Newton’s third law (calculate every force twice)
We have to thread parallelize other parts, such as pair-list construction.
We have to treat SIMDization and thread parallelization simultaneously.

Parallelization (2/3)

Pseudo-flat MPI

Full domain decomposition for both intra- and inner-nodes.



MDUnit class :
- Represent an OpenMP thread
- Responsible for calculation

MDManager class:
- Represent an MPI process
- Responsible for communication

```
DO I=1,THREAD_NUM  ← Loop parallelization
  CALL MDUnit[i]->Calculate()
ENDDO
```

Merits

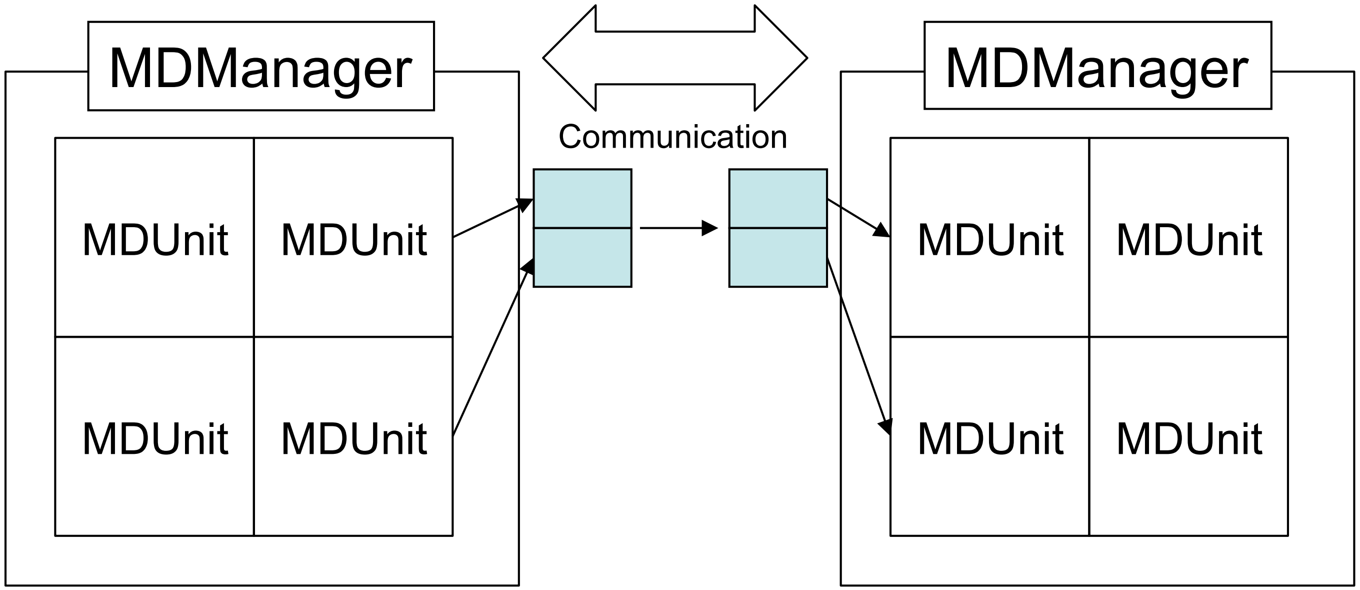
It is straightforward to implement pseudo-flat MPI codes from flat MPI codes.
Memory locality is automatically satisfied. (not related to the K computer)
We have to take care SIMDization only at the hot-spot (force calculation).
→ We do not have to take care two kinds of load balance simultaneously.

Parallelization (3/3)

Demerits

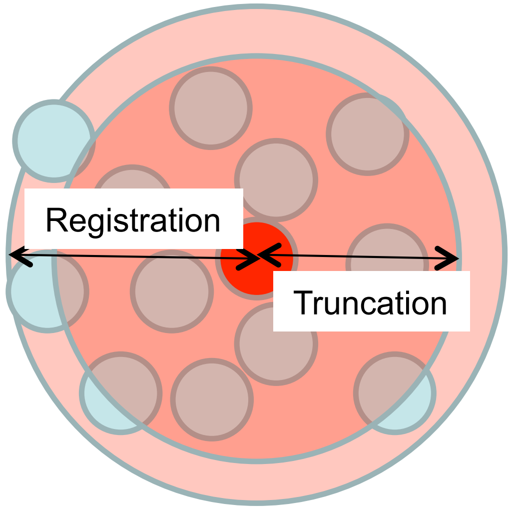
Two-level communication is required.
→ Limitation of MPI call in thread.
→ Data should be packed before being sent by a master thread.

It is troublesome to compute physical quantities such as pressure, and so on.



Algorithms for MD

Pair-list Construction

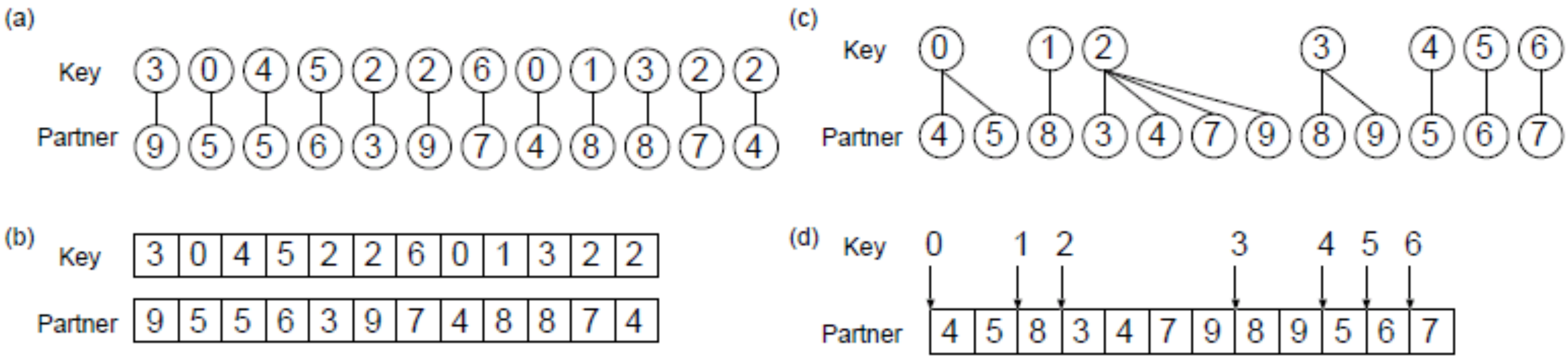


Grid Search
O(N) method to find interacting particle-pairs

Bookkeeping method
Reuse the pair-list by registering pairs in a length longer than truncation length.

Sorting of Pair-list

Sort by i-particle in order to reduce memory access



SIMDization

Non-use of Newton's third law

We find that store operations involving indirect addressing is quite expensive for the K computer.

→ We avoid indirect storing by not using Newton's third law.
Computations becomes double, but speed up is more than twofold.

The value of FLOPS is meaningless.

Hand-SIMDization

Compiler cannot generate SIMD operations efficiently.

→ We use intrinsic SIMD instructions explicitly. (emmintrin.h)

Divide instruction (fdivd) cannot be specified as a SIMD instruction.
We use a reciprocal approximation (frcpd) which is fast, but accuracy is only 8bit.
We improve the precision by additional calculations.

The loop is unrolled by four times to enhance the software pipelining.

You can see the source codes from: http://mdacp.sourceforge.net/

Benchmark (1/3) - Conditions -

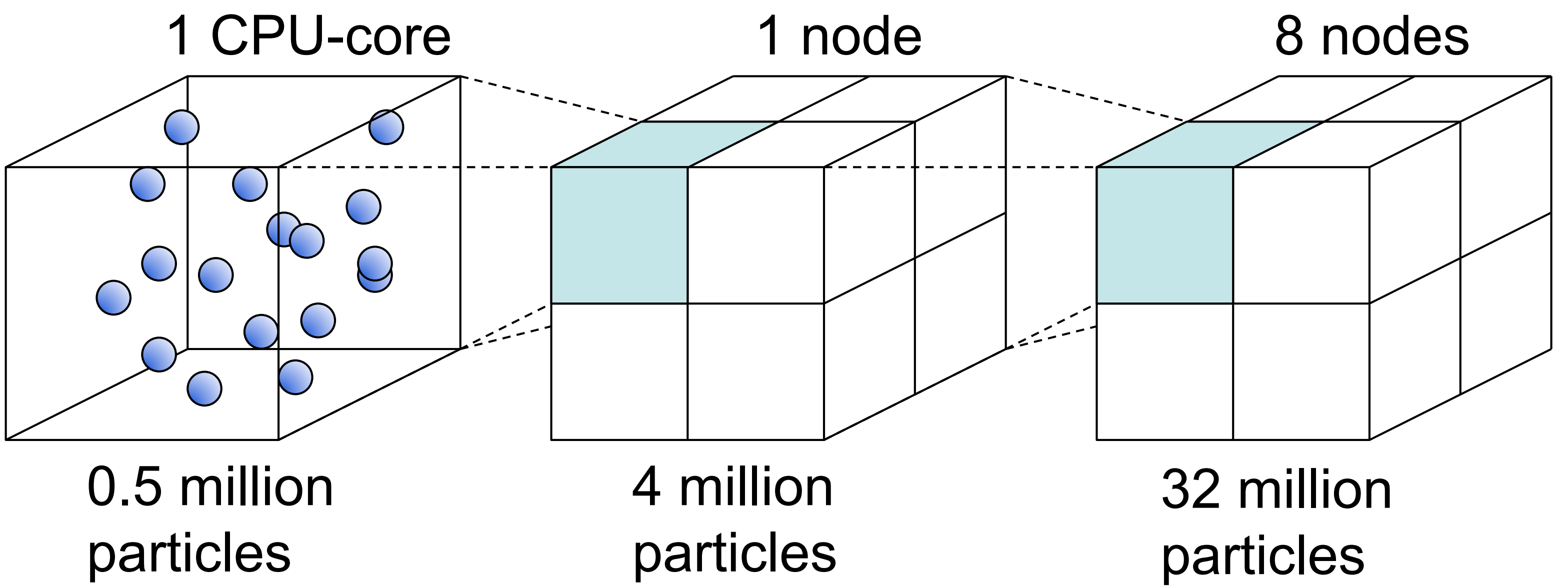
Conditions

Truncated Lennard-Jones potential with cutoff length 2.5 (in LJ units)
Initial Condition is FCC and density is 0.5
Integration Scheme: 2^{nd} order symplectic with dt = 0.001
0.5 million particles on a CPU-core (4 million particles on a node)
After 150 steps, measure time required for 1000 steps.

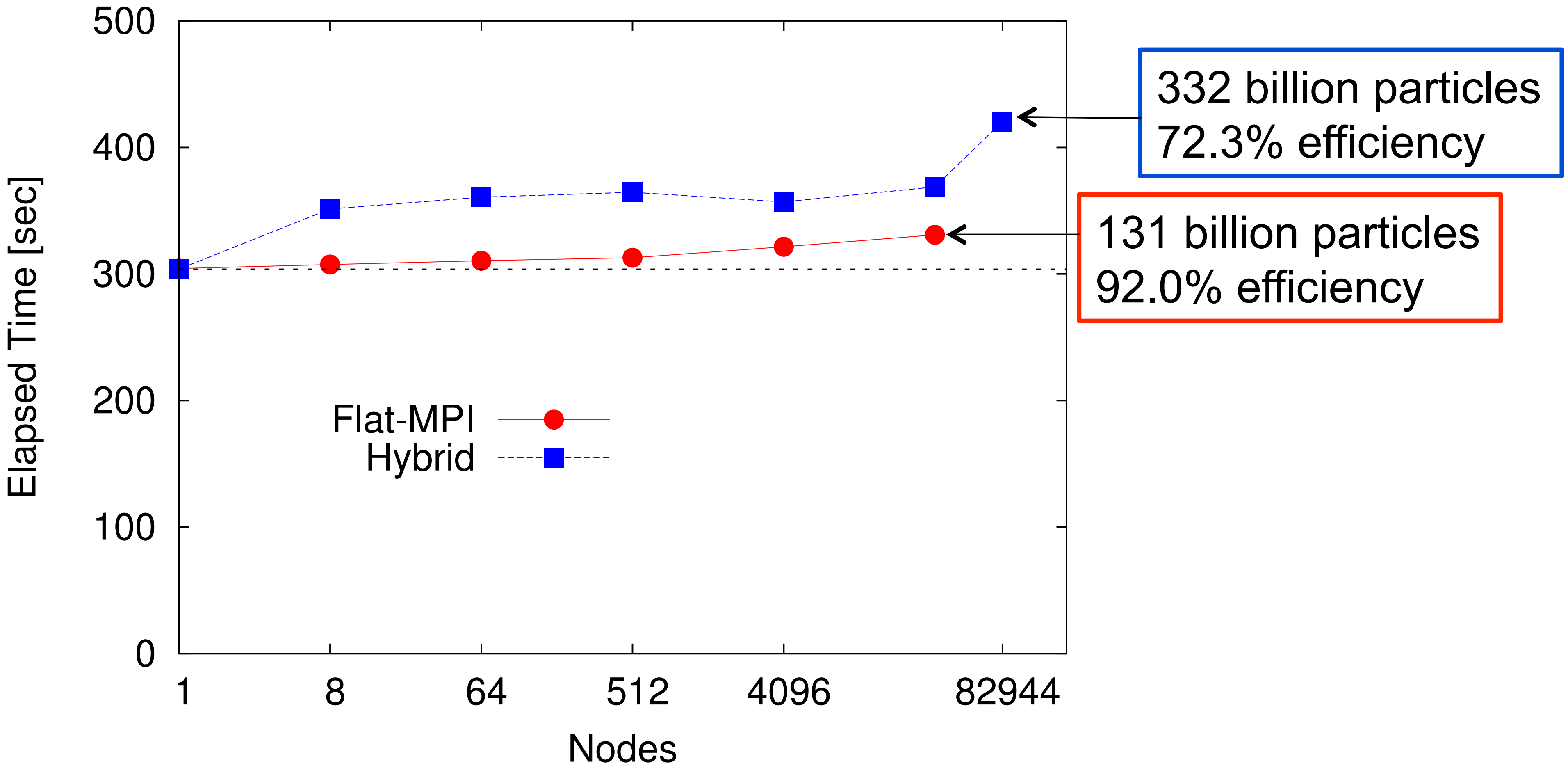
Parallelization

Flat MPI : 8 processes /node up to 32768 nodes (4.2 PF)
Hybrid : 8 threads/node up to 82944 nodes (10.6 PF)

Computations assigned to each core are perfectly identical for both scheme.



Benchmark (2/3) - Elapse Time -



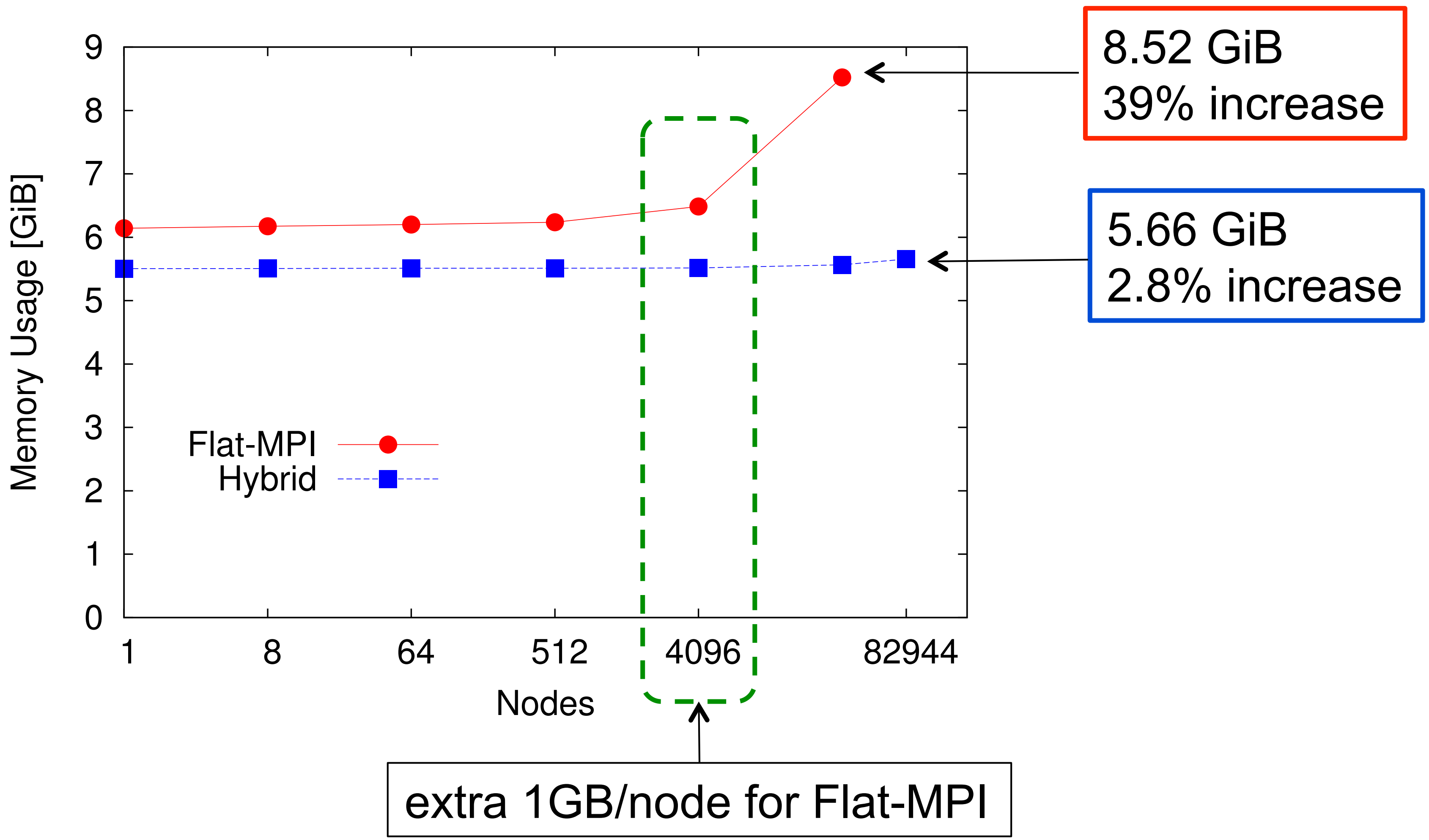
Nice scaling for the flat MPI (92% efficiency compared with a single node)

Performance of the largest run: 1.76 PFLOPS (16.6%)

University of Tokyo

Benchmark (3/3) - Memory Usage -

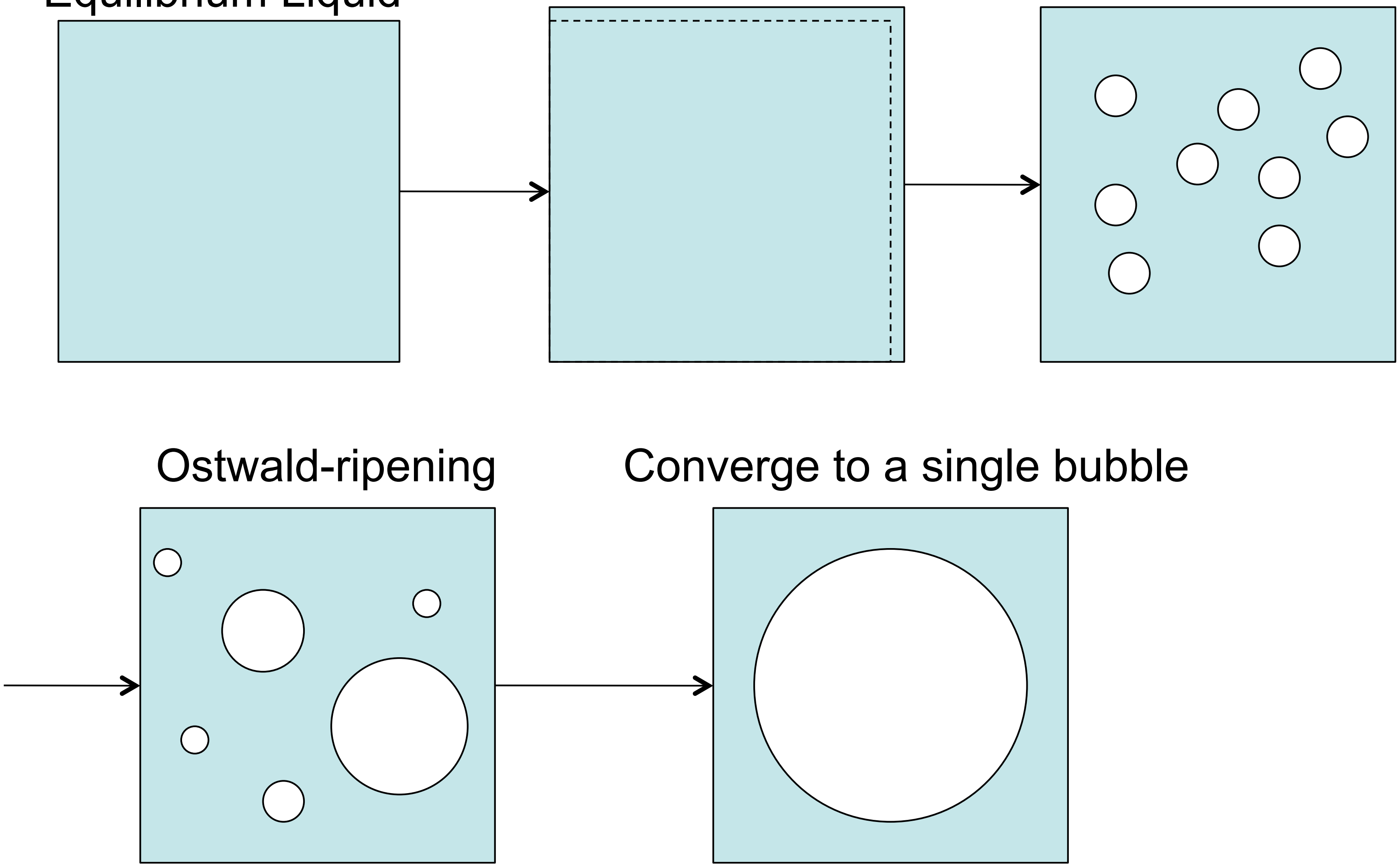
Usage of memory is almost flat for hybrid.



We performed product runs using 4096 nodes adopting flat-MPI.

University of Tokyo

Time Evolution of Multi-bubble nuclei

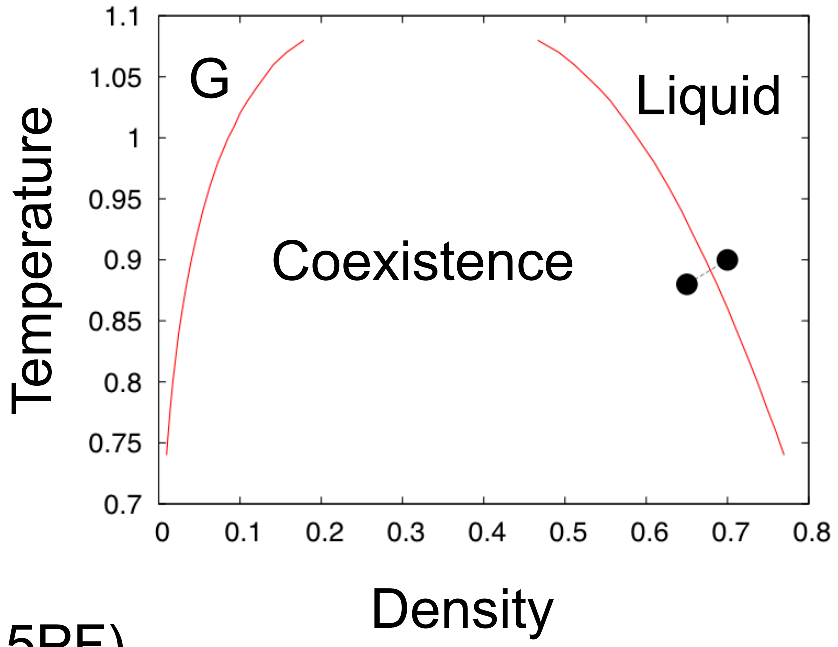


Larger bubbles becomes larger,
smaller bubbles becomes smaller

Details of Simulation

Condition

Initial Condition : Pure Liquid
Truncation Lenght : 3.0
Periodic Boundary Conditon:
Time Step : 0.005
Integration : 2^{nd} order Symplectic (NVE)
2^{nd} order RESPA (NVT)



Computational Scale

Resource : 4096 nodes of K computer (0.5PF)
Flat-MPI: 32768 processes
Parallelization : Simple Domain Decomposition
Particles : 400 ～ 700 million particles
System Size: L = 960 (in LJ units)
Thermalize: 10,000 steps + Observation : 1,000,000 steps

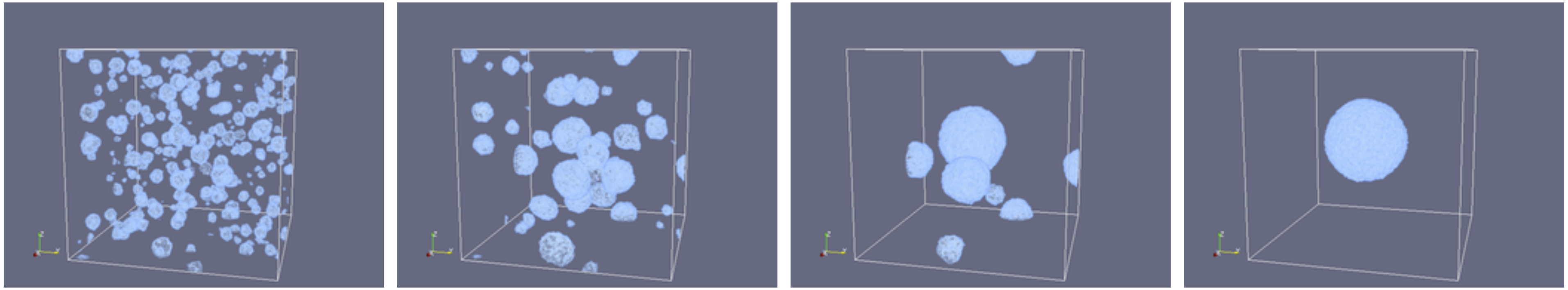
Movie

24 hours / run
1 run ～ 0.1million node hours
10 samples ＝1 million node hours

Multi-bubble Nuclei and Ostwald-like Ripening

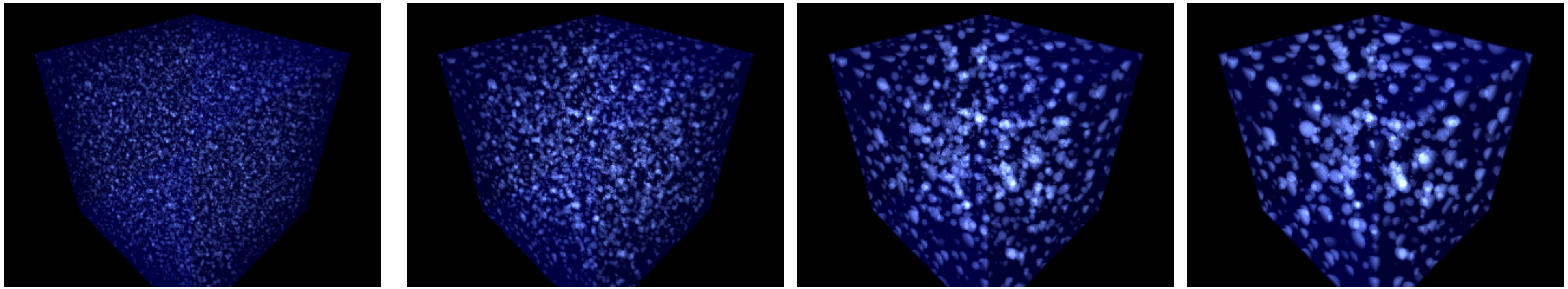
23 million particles

Initial stage | intermediate stage | Final stage



Multi nuclei | Ostwald-like ripening | One bubble survives

1.5 billion particles



Bubbles appear a the results of particle interactions.
Ostwald-like ripening is observed as the results of bubble interactions.
→ Direct simulations of multi-scale and multi-physics phenomena.

University of Tokyo

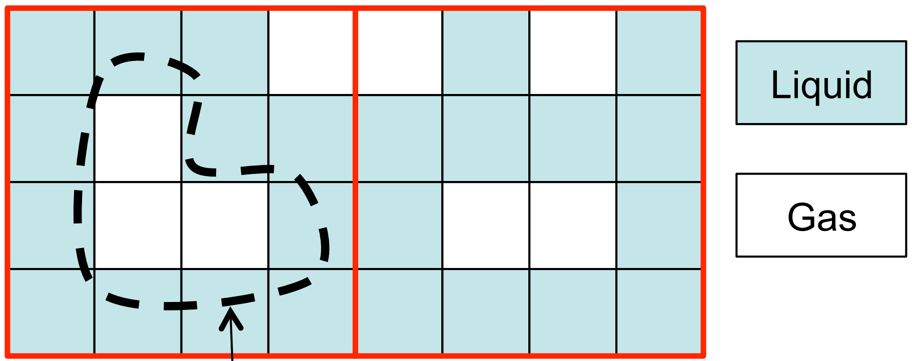
Identification of Bubbles

Definition of Bubble

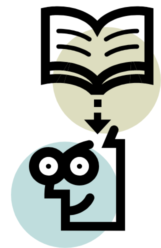
Divide a system into small cells.
A cell with density smaller than some threshold is defined to be gas phase.
Identify bubbles on the basis of site-percolation criterion (clustering).

A type of binarization

Process A Process B



Bubble

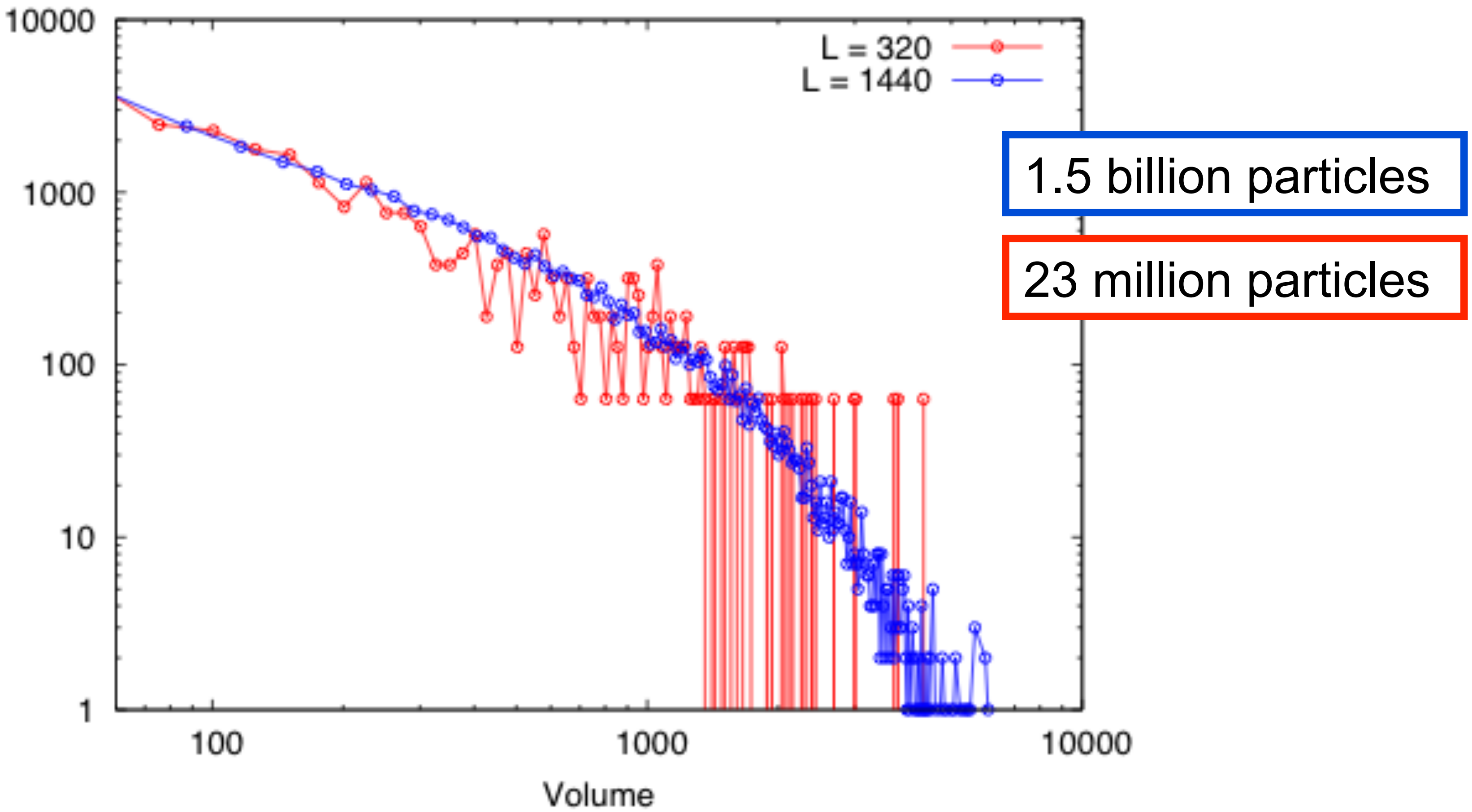


We do not have to take care of byte-order!

Data Format

A number of particles in a cell is stored as unsigned char (1 Byte).
A system is divided into 32.7 million cells → 32.7 MB / snap shots.
1000 frames/run → 32.7GB/run.
Clustering is performed in post process.

Bubble-size distribution



Bubble size distribution at the 10,000th step after the expansion.
Accuracy of the smaller run is insufficient for further analysis.
→ We need 1 billion particles to study population dynamics of bubbles which is impossible without Peta-scale computers.

Lifshitz-Slyozov-Wagner (LSW) Theory

Definitions

$f(v,t)$ a number of bubbles which has volume v at time t.

$$\frac{\partial f}{\partial t} = -\frac{\partial}{\partial v}(\dot{v} f)$$ Governing Equation

$\dot{v}(v,t)$ Kinetic Term (Microscopic Dynamics)

Assumptions

- Mean field approximation
- Mechanical equilibrium

} shared with the classical nucleation theory

- Conservation of the total volume of gas phase
- Self-similarity of the distribution function $f(v,t) \sim t^y \tilde{f}(vt^{-x})$

Predictions

Power-law behaviors

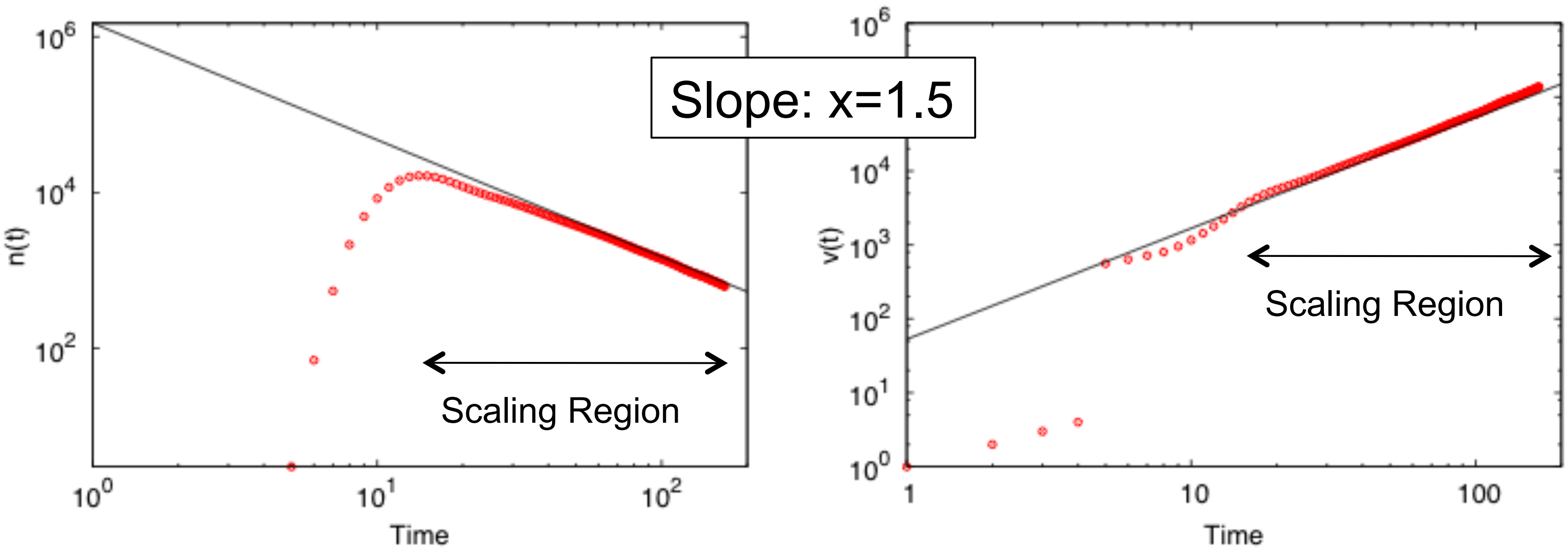
$n(t) \sim t^{-x}$ The total number of bubbles

$\bar{v}(t) \sim t^x$ The average volume of bubbles

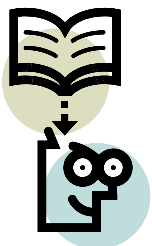
The scaling index x depends on microscopic dynamics.

Simulations Results (1/2)

Number of Bubbles | Average Volume of Bubbles



Power-law behavior in late-stage of time evolution.
Sharing the value of an exponent as predicted by the theory.



The LSW theory works fine.

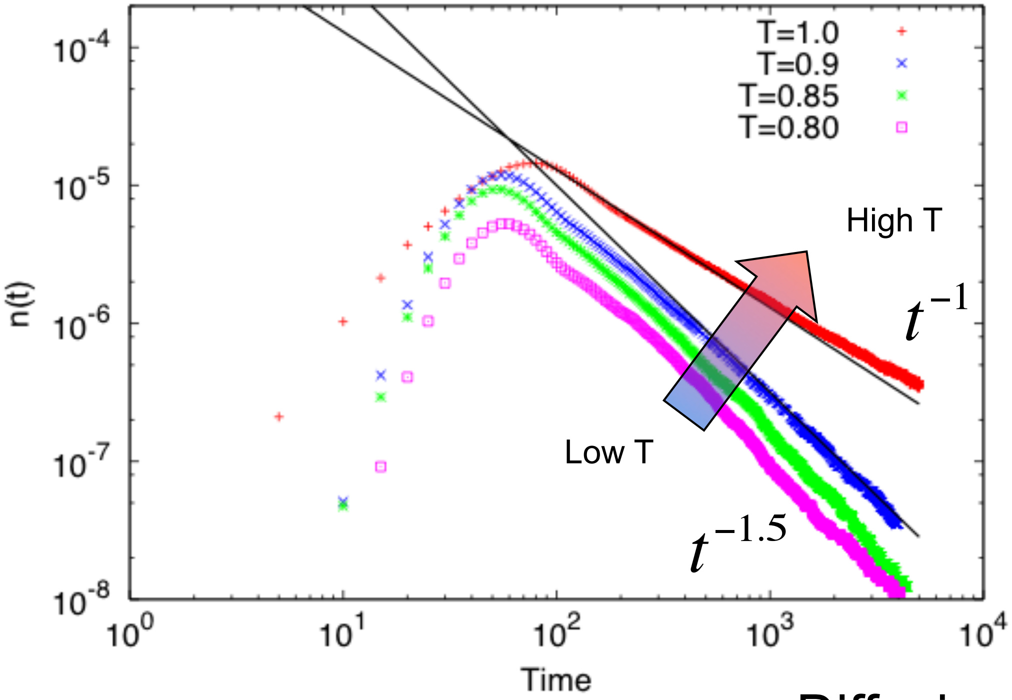
➡ Assumptions are valid.

We cannot obtain the scaling region using small systems.

➡ Huge-scale simulation is necessary.

Simulations Results (2/2)

Scaling exponent changed from 1.5 to 1



Reaction-Limit

$$\dot{v} \propto R^2 \Delta\mu$$

→ x=3/2

Diffusion-Limit

$$\dot{v} \propto R^2 J = DR^2 \left.\frac{d\rho}{dr}\right|_{r=R}$$

→ x=1

Microscopic dynamics can be investigated from macroscopic behavior.

University of Tokyo

Summary and Discussion

Summary

We did physics using a peta-scale computer.

Simulations involving billions of particles allows us to investigate multi-scale and multi-physics problems directly.

Toward Exa-scale Computing

Please give us decent programming language!

MPI (library), OpenMP (directive), SIMD (intrinsic functions)

References

MDACP (Molecular Dynamics code for Avogadro Challenge Project)
Source codes are available online: http://mdacp.sourceforge.net/

MDACP SEARCH

Acknowledgements

Computational Resources:
"Large-scale HPC Challenge" Project, ITC, Univ. of Tokyo
Institute of Solid State Physics, Univ. of Tokyo
HPCI project (hp130047) on the K computer, AICS, RIKEN

Financial Supports:
CMSI
KAUST GRP (KUK-I1-005-04)
KAKENHI (23740287)

University of Tokyo