

## A Special-purpose Computer for Molecular Dynamics Simulations: MDM

Tetsu Narumi

*High Performance Biocomputing Team, Computer/Informatics Facilities, RIKEN Genomic Sciences Center, Maeda 214, Totsuka, Yokohama, Kanagawa 244-0408, Japan*

**Abstract.** We developed a special-purpose computer for molecular dynamics simulations, which we call Molecular Dynamics Machine (MDM). MDM accelerates the Coulomb and van der Waals force calculation with the Ewald method. It is composed of three parts: MDGRAPE-2, WINE-2 and a host computer. MDGRAPE-2 accelerates the real-space part of the Coulomb and van der Waals forces. WINE-2 accelerates wavenumber-space part of the Coulomb force. The host computer performs other calculations. The peak performance is 78 Tflop/s. It can also be used for cosmological, Smoothed Particle Hydrodynamics, and vortex dynamics simulations.

### 1. Introduction

Molecular dynamics (MD) simulations have been widely used to study the physical properties of condensed matter at the atomic level. There are strong motivations to perform MD simulations with a large number of atoms to study large molecular systems, such as proteins or lipid bilayers. Even when the system contains simple molecules, one needs a very large number of atoms to get the equilibrium properties of a system with very high accuracy.

In MD simulations each atom is treated as a classical particle. An inter atomic interaction contains the Coulomb, van der Waals and bonding forces. Newton's equations of motion of atoms are solved by numerical integration. The most time consuming part is the calculation of Coulomb and Van der Waals forces. In particular, the calculation cost for the Coulomb force is prohibitively high for systems larger than 10,000 atoms, since the Coulomb force is a long-range force; the force from an atom cannot be ignored no matter the distance to other atoms.

There are many algorithms to reduce the calculation cost for the Coulomb force. The Ewald method (Ewald 1921) is often used under the periodic boundary condition to reduce the calculation cost for the Coulomb force to  $O(N^{3/2})$  instead of the native method's  $O(N^2)$ . Moreover, many other faster methods (Barnes & Hut 1986; Greengard & Rokhlin 1987; Essmann, Perera, & Berkowitz 1995; Strain 1992) that scale as  $O(N)$  or  $O(N \log N)$  have been developed. However, the accuracy of these methods has not been well discussed on systems with a large numbers of particles, such as 1,000,000 or larger.

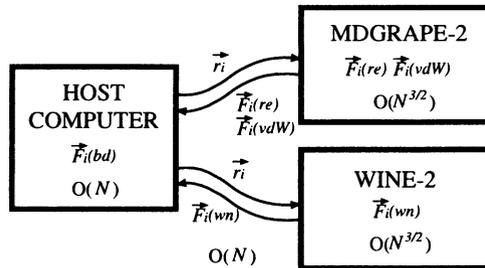


Figure 1. Basic structure of the Molecular Dynamics Machine.

In order to meet the huge computational requirements for the Coulomb force and to realize a large scale simulation with one million particles, we have developed a special-purpose computer, the Molecular Dynamics Machine (MDM), based on the Ewald method. The MDM is divided into three parts: MDGRAPE-2, WINE-2 and a host computer (figure 1). MDGRAPE-2 calculates the real-space part of the Coulomb and van der Waals forces. WINE-2 calculates the wavenumber-space part of the Coulomb force. The host computer performs the bonding force calculation and the other operations; for example, updating the positions and velocities of the particles. The peak speed of MDM is 78 Tflop/s.

Sugimoto et al. (1990) have developed a series of application specific processor designs called GRAPE, which is specially architected to accelerate the inter-particle force calculation. Their latest machine, GRAPE-6, will reach 100 Tflop/s (Makino 2001). Their group also developed a special-purpose hardware for MD simulations, such as MD-GRAPE (Taiji et al. 1994) and WINE-1 (Fukushige et al. 1993). MDM is the successor of these machines (Narumi et al. 1999; Narumi et al. 2001).

In this paper, we describe the MDM system and report preliminary performance of it. Section 2 presents the method of calculation adopted; section 3 gives an outline of the MDM hardware specifications while in section 4 is described the software for MDM. Performance of the MDM is reported in section 5. In section 6 we discuss future applications as well as which kind of astronomical simulations can be accelerated by the MDM.

## 2. Calculation of Forces in the Ewald Method

The force,  $\vec{F}_i$ , on a particle  $i$  is expressed as:

$$\vec{F}_i = \vec{F}_i(\text{Clb}) + \vec{F}_i(\text{vdW}) + \vec{F}_i(\text{bd}), \quad (1)$$

where  $\vec{F}_i(\text{Clb})$  is the Coulomb force,  $\vec{F}_i(\text{vdW})$  is the van der Waals force, and  $\vec{F}_i(\text{bd})$  is the bonding force by covalent and hydrogen bonds.

The Coulomb force,  $\vec{F}_i(\text{Clb})$ , on a particle  $i$  is divided into two parts: the real-space part,  $\vec{F}_i(\text{re})$ , and the wavenumber-space part,  $\vec{F}_i(\text{wn})$ , in the Ewald method:

$$\vec{F}_i(\text{Clb}) = \vec{F}_i(\text{re}) + \vec{F}_i(\text{wn}). \quad (2)$$

Here,  $\vec{F}_i(\text{re})$  and  $\vec{F}_i(\text{wn})$  are expressed as

$$\vec{F}_i(\text{re}) = \frac{q_i}{4\pi\epsilon_0} \sum_j^{\tau_{ij} < \tau_{cut}} q_j \left[ \frac{\text{erfc}(\alpha r_{ij}/L)}{r_{ij}} + \frac{2\alpha}{\pi^{1/2}L} \exp(-\alpha^2 r_{ij}^2/L^2) \right] \frac{\vec{r}_{ij}}{r_{ij}^2}, \quad (3)$$

$$\begin{aligned} \vec{F}_i(\text{wn}) &= \frac{q_i}{2\pi\epsilon_0 L^3} \sum_{\vec{k}}^{k < k_{cut}} \frac{\vec{k}}{k^2} \exp(-\pi^2 L^2 k^2 / \alpha^2) \\ &\times \left[ \sin(2\pi\vec{k} \cdot \vec{r}_i) \sum_j q_j \cos(2\pi\vec{k} \cdot \vec{r}_j) - \cos(2\pi\vec{k} \cdot \vec{r}_i) \sum_j q_j \sin(2\pi\vec{k} \cdot \vec{r}_j) \right], \quad (4) \end{aligned}$$

where  $\vec{r}_i$  is the position of particle  $i$ ,  $\vec{r}_{ij}$  is the relative vector from particle  $j$  to particle  $i$ ,  $r_{ij}$  is the distance between particles  $i$  and  $j$ ,  $q_i$  is the electrostatic charge of particle  $i$ ,  $\vec{k}$  is the wavenumber vector,  $k$  is the length of  $\vec{k}$ ,  $L$  is the length of a side of the computational box,  $r_{cut}$  and  $k_{cut}$  are the cut-off lengths of relative vector and wavenumber vector, respectively,  $\epsilon_0$  is the dielectric constant of vacuum,  $\alpha$  is a parameter to balance the computational cost for the real and wavenumber parts of the Coulomb force calculation, and  $\text{erfc}(x)$  is the complementary error function.

The van der Waals force,  $\vec{F}_i(\text{vdW})$ , on particle  $i$  is calculated as follows when the potential function is Lennard-Jones:

$$\vec{F}_i(\text{vdW}) = \sum_j^{\tau_{ij} < \tau_{cut}} \varepsilon(at_i, at_j) \left\{ 2 \left[ \frac{\sigma(at_i, at_j)}{r_{ij}} \right]^{14} - \left[ \frac{\sigma(at_i, at_j)}{r_{ij}} \right]^8 \right\} \vec{r}_{ij}, \quad (5)$$

where  $\varepsilon(at_i, at_j)$  and  $\sigma(at_i, at_j)$  are parameters which are determined by the particle types,  $at_i$  and  $at_j$ .  $at_i$  is the particle type of particle  $i$ .

MDGRAPE-2 uses the cell-index (link-cell) method (Hockney & Eastwood 1981), to search the neighboring particles instead of comparing the distance of each particle. A simulation box is divided into cells, and particles are allocated to them. We can get the neighboring particles by searching only neighboring cells. The calculation cost for searching neighboring cells is very small compared with that for directly searching neighboring particles. In the case of MDGRAPE-2, we set the size of a cell to a little larger than  $r_{cut}$ . So a particle in a cell interact with particles in neighboring 27 cells including itself; equations 3 and 5 can be rewritten as:

$$\vec{F}_i(\text{re}) = \frac{q_i}{4\pi\epsilon_0} \sum_{c=1}^{27} \sum_{j=j_{start_c}}^{j_{end_c}} \vec{f}_{i,j}(\text{re}), \quad (6)$$

$$\vec{F}_i(\text{vdW}) = \sum_{c=1}^{27} \sum_{j=j_{start_c}}^{j_{end_c}} \vec{f}_{i,j}(\text{vdW}). \quad (7)$$

Here  $c$  is the index of a neighboring cell,  $j_{start_c}$  is the first index of a particle in cell  $c$ ,  $j_{end_c}$  is the last index of a particle in cell  $c$ ,  $\vec{f}_{i,j}(\text{re})$  and  $\vec{f}_{i,j}(\text{vdW})$  are pairwise Coulomb and van der Waals forces between particles  $i$  and  $j$ , respectively. We assumed that the indices of particles in a cell are contiguous.

### 3. MDM System

Figure 1 shows the basic structure of the MDM. MDM is composed of two special-purpose computers (MDGRAPE-2 and WINE-2) and a host computer.

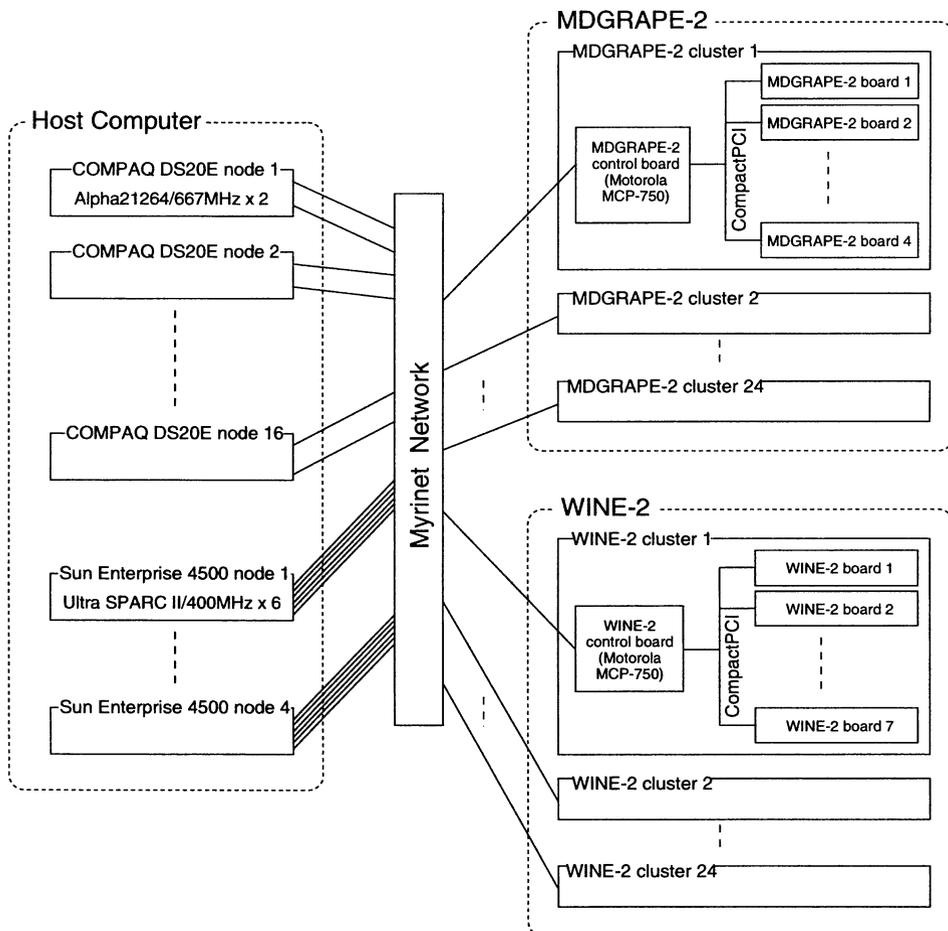


Figure 2. Block diagram of the Molecular Dynamics Machine (MDM).

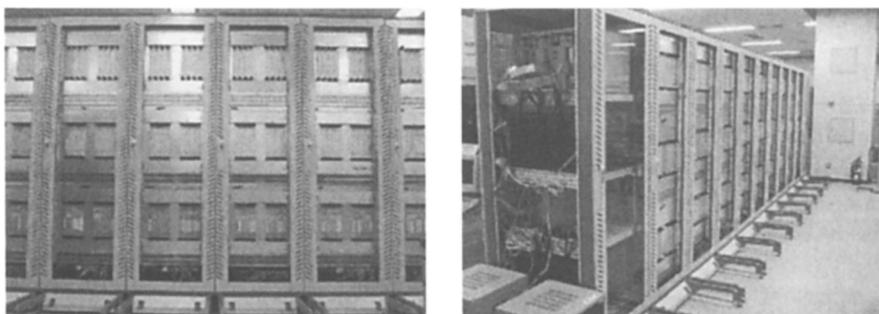


Figure 3. Front view of the MDM (left). Side view of the MDM (right).

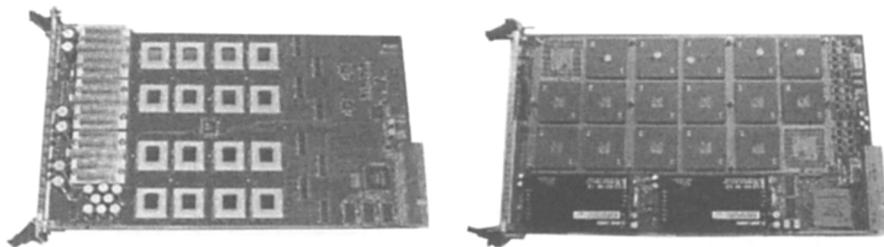


Figure 4. An MDGRAPE-2 board (left) and a WINE-2 board (right).

The host computer is a general-purpose computer, and works as a front-end processor.

The basic calculation flow in one time-step of MD simulation is as follows. First, the host computer sends the coordinates of particles to MDGRAPE-2 and WINE-2. Second, MDGRAPE-2 calculates the Coulomb force from real-space and van der Waals force, and WINE-2 calculates the Coulomb force from wavenumber-space. Third, the host computer receives the forces on particles from MDGRAPE-2 and WINE-2. Fourth, the host computer performs other operations; bonding force calculation, time integration of particles, file I/O, etc.

The calculation cost on two special-purpose computers scales as  $O(N^{3/2})$ , while that on the host computer and the communication between them scale as  $O(N)$ . Therefore, we can accelerate the MD simulation by accelerating only the long-range force calculation in the case of large  $N$ ; the host computer and the communication do not cause the bottleneck of the system.

Figure 2 shows the block diagram of the MDM, and figure 3 shows the photos of it. In the following sections, we describe the hardware in detail especially targeted on MDGRAPE-2.

### 3.1. Host computer

The host computer is composed of a 16-node Compaq DS20E and a 4-node Sun Microsystems Enterprise 4500. A DS20E has two Alpha 21264/667MHz CPUs and 2 Gbyte of memory. An Enterprise 4500 has 6 Ultra SPARC II/400MHz CPUs and 6 Gbyte of memory.

All the CPUs including the control boards are connected via a Myrinet network. Myrinet is a high speed network provided by Myricom, Inc. Each CPU has a dedicated Myrinet network card. We used twelve 16-port Myrinet switches.

### 3.2. MDGRAPE-2

MDGRAPE-2 consists of 24 MDGRAPE-2 clusters. Each MDGRAPE-2 cluster has 4 MDGRAPE-2 boards and a control board (Motorola MCP-750). An MDGRAPE-2 board [figure 4(left)] has 16 MDGRAPE-2 chips. The total number of MDGRAPE-2 chips is 1,536. The peak speed of MDGRAPE-2 corresponds to 24.6 Tflop/s.

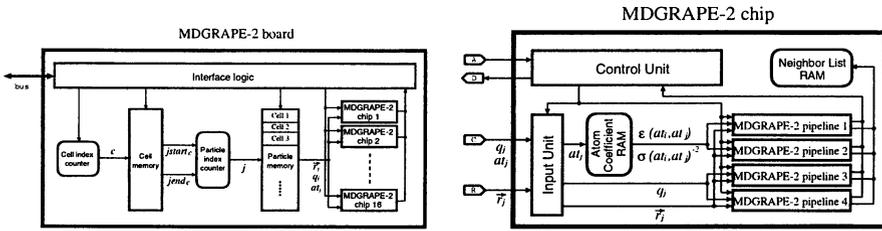


Figure 5. Block diagram of an MDGRAPE-2 board (left) and an MDGRAPE-2 chip (right).

**MDGRAPE-2 board** An MDGRAPE-2 board [figure 5(left)] is composed of 16 MDGRAPE-2 chips, interface logic, cell index counter, cell memory, particle index counter, and particle memory. A host computer communicates with MDGRAPE-2 chips, counters and memories on a board through interface logic. The index of a particle is determined by dual counters to support cell-index method (equations 6 and 7). Cell index counter specifies the neighboring cell index  $c$ , and cell memory outputs the range of indices in the cell  $c$ . Particle index counter indicates the particle index  $j$  to particle memory. The position, charge, and particle type of a particle  $j$  are supplied to all of the MDGRAPE-2 chips. An MDGRAPE-2 board has a 6U CompactPCI standard form factor, except that its depth is 40cm instead of 16cm. Interface logic, cell index counter, cell memory, particle index counter are integrated in an FPGA (EPF10K200SFC672-1 by Altera). 20 Mbyte of SDRAM is used for the particle memory.

**MDGRAPE-2 chip** An MDGRAPE-2 chip [figure 5(right)] has four identical MDGRAPE-2 pipelines, atom coefficient RAM, and neighbor list RAM. Atom coefficient RAM stores the coefficients  $\sigma(at_i, at_j)$  and  $\epsilon(at_i, at_j)$ . The maximum number of particle types is 32, which is enough for MD simulation with proteins. Neighbor list RAM, which was not used in our simulation, can be used to search neighboring particles. Peak performance of an MDGRAPE-2 chip corresponds to about 16 Gflop/s at a clock frequency of 100 MHz. It is made by IBM with SA-12 technology, whose design rule is  $0.25\mu\text{m}$  and operation voltage is 2.5V. The number of transistors of an MDGRAPE-2 chip is 9 million.

**MDGRAPE-2 pipeline** An MDGRAPE-2 pipeline calculates the pairwise force  $\vec{f}_{i,j}$  as:

$$\vec{f}_{i,j} = b_{ij}g(a_{ij}r_{ij}^2)\vec{r}_{ij}, \quad (8)$$

where  $g(x)$  is an arbitrary central force, and  $a_{ij}$  and  $b_{ij}$  are coefficients determined by particle types of particles  $i$  and  $j$ .

Figure 6(left) shows the structure of the pipeline. The pipeline calculates  $\vec{r}_{ij}$  and  $a_{ij}r_{ij}^2$ , and then evaluate  $g(x)$  in the function evaluator. Function evaluator performs fourth-order interpolation segmented by 1,024 region. The coefficients of the interpolation function are stored in the RAM in function evaluator. Therefore, we can use any arbitrary central force by changing the contents of the RAM. After the function evaluator, the pipeline multiplies  $b_{ij}$

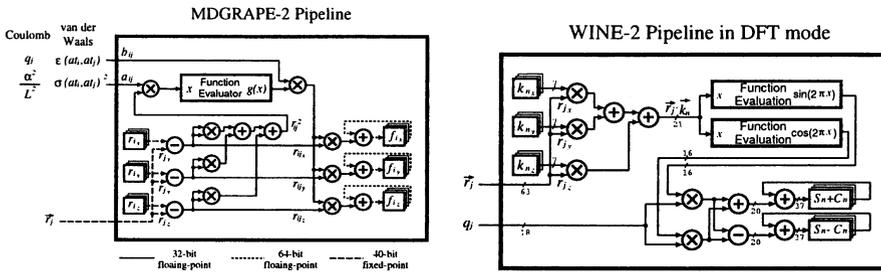


Figure 6. Block diagram of a pipeline of MDGRAPE-2 (left) and WINE-2 chips (right).

and  $\vec{r}_{ij}$ , and then accumulates them. The relative accuracy of a pairwise force is about  $10^{-7}$ , since most of the arithmetic units in the pipeline use IEEE/754 single floating point format. The double floating point format is used for accumulating the force in order to prevent the underflow when large number of particles are used.

In the case of real-space part of the Coulomb force,  $\vec{f}_{i,j}(re)$  in equation 6 is calculated as:

$$g(x) = \frac{2 \exp(-x)}{\pi^{1/2} x} + \frac{\operatorname{erfc}(x^{1/2})}{x^{3/2}},$$

$$a_{ij} = \alpha^2 L^{-2},$$

$$b_{ij} = q_j.$$

In the case of van der Waals force,  $\vec{f}_{i,j}(vdW)$  in equation 7 is calculated as:

$$g(x) = 2x^{-7} - x^{-4},$$

$$a_{ij} = \sigma(at_i, at_j)^{-2},$$

$$b_{ij} = \epsilon(at_i, at_j).$$

### 3.3. WINE-2

WINE-2 is composed of 24 WINE-2 clusters. Each WINE-2 cluster has 7 WINE-2 boards and a control board (Motorola MCP-750). A WINE-2 board [figure 4(right)] has 16 WINE-2 chips. The total number of WINE-2 chips is 2,688. The peak speed of WINE-2 corresponds to 53.6 Tflop/s.

A WINE-2 chip has eight identical WINE-2 pipelines. A WINE-2 pipeline performs discrete Fourier transform (DFT) or inverse DFT (IDFT) to calculate the wavenumber-space part of the Coulomb force. Figure 6(right) shows the structure of the pipeline in DFT mode. Detailed description of WINE-2 is presented by Narumi et al. 2000.

### 4. Software

The difference of the program when we use MDM or not is that we call library routines to calculate real-space and wavenumber-space forces instead of calling

internal force subroutines. Message Passing Interface (MPI) is used for parallel processing. The library routines call device drivers for each operating system. Tables 1 and 2 show library routines which are often used from the MD program.

For real-space part, communication between processes must be done by user. The function table for  $g(x)$  is generated beforehand by a separate utility program, and loaded to MDGRAPE-2 chips at the beginning of the simulation by calling `m2_allocate_unit`.

For wavenumber-space part, the library routine for force calculation is already parallelized with MPI, and users do not care any communication between processes.

Table 1. Library routine for MDGRAPE-2

Category	Name	Function
Initialization	<code>m2_allocate_unit</code>	set the function table, and acquire MDGRAPE-2 boards
Setting parameters for atoms	<code>m2_set_positions</code> <code>m2_set_charges</code> <code>m2_set_types</code> <code>m2_set_charge_matrix</code> <code>m2_set_rscale_matrix</code>	set positions of atoms set charges of atoms set atom types of atoms set $\varepsilon(at_i, at_j)$ for each atom type set $\sigma(at_i, at_j)$ for each atom type
Force calculation	<code>m2_set_cells</code> <code>m2_set_pipeline_types</code> <code>m2_calculate_forces</code>	set the cell memory set atom types of atoms calculate forces on atoms
Finalization	<code>m2_free_unit</code>	release MDGRAPE-2 boards

Table 2. Library routine for WINE-2

Category	Name	Function
Initialization	<code>wine2_set_MPI_community</code>	set the MPI community for wavenumber-space part
	<code>wine2_allocate_board</code>	set the number of WINE-2 boards to acquire
	<code>wine2_initialize_board</code>	acquire WINE-2 boards
Force calculation	<code>wine2_set_nn</code>	set the number of particles for each process
	<code>calculate_force_and_pot_wavepart_nooffset</code>	calculate the wavenumber-space part of force
Finalization	<code>wine2_free_board</code>	release WINE-2 boards

## 5. Performance

We performed an MD simulation of 33,461,708 NaCl pairs (66,923,416 particles) for 5,000 time-steps (10 ps). The elapsed time was 78.7 hours (283,233 seconds). We obtained an effective calculated speed of 8.61 Tflop/s.

We adopted the Tosi-Fumi potential (Tosi & Fumi 1964) as a force field between ions:

$$\phi(r_{ij}) = \frac{q_i q_j}{r_{ij}} + A_{ij} b \exp\left(\frac{\sigma_i + \sigma_j - r_{ij}}{\rho}\right) - \frac{c_{ij}}{r_{ij}^6} - \frac{d_{ij}}{r_{ij}^8}, \quad (9)$$

where  $A_{ij}$ ,  $b$ ,  $\sigma_i$ ,  $\sigma_j$ ,  $\rho$ ,  $c_{ij}$  and  $d_{ij}$  are parameters. The length,  $L$ , of a side of the simulation box is 1297 Å. The cut-off length,  $r_{cut}$ , of the real-space part of the Coulomb and other forces is 54.0 Å. The non-dimensional cut-off length in the wavenumber-space part of the Coulomb force is set to be  $Lk_{cut} = 55.0$ . We used a time-step of 2 fs and a temperature of 1200 K. We performed a canonical ensemble ( $NVT$  constant) MD simulation in the initial 4,000 steps by scaling velocities, and a micro-canonical ( $NVE$  constant) one in the next 1,000 steps. The total energy is well conserved; relative error of the total energy is less than  $1.7 \times 10^{-5}$  percent.

The fluctuation of the temperature during the  $NVE$  constant simulation is less than 0.1 Kelvin, which confirms the necessity of using a very large number of particles in order to characterize with good accuracy the physical-chemical property of a system of interacting ions.

We estimated the effective performance as follows. First, we count the number of floating operations needed for real- and wavenumber-space parts of Ewald method with a general-purpose computer. Then we convert the calculation speed to flop/s. MDGRAPE-2 performs many times larger number of floating point operations than a general-purpose computer because of hardware simplicity. Therefore the effective speed becomes 8.61 Tflop/s even though the MDM itself calculates at a speed of 35.4 Tflop/s. The detailed description is presented in Narumi et al. 2001.

## 6. Discussion

The size of the cell of the MD simulation is larger than 0.1  $\mu\text{m}$ . Many targets will be included if we can perform 0.1  $\mu\text{m}$  size of a simulation. For example, the order of the size of a grain is 0.1–1  $\mu\text{m}$  in typical polycrystals. Polycrystals is useful in industry products, and we can study with NaCl system. The size of a transistor of the latest LSI is about 0.1  $\mu\text{m}$ , which may also be the target application of the MDM.

In the biological field, the size of a cold virus is smaller than 0.1  $\mu\text{m}$ . Most biological molecules are much smaller than 0.1  $\mu\text{m}$ , but they often forms complexes with special structures, such as a lipid bilayer and a molecular motor. We need 0.1  $\mu\text{m}$  size of a simulation to treat even the basic structure of them.

One of the big target of the MDM is to simulate the folding of a protein from a straight string. The folding process of a protein is not well studied mainly because of the huge computational cost. We believe our MDM will shorten the way to realize the folding simulation of a protein.

In the astronomical field, MDM can be used for such as cosmological simulation (Fukushige et al. 1993), Smoothed Particle Hydrodynamics (SPH) (Umemura 1993; Steinmetz 1996), and vortex dynamics simulation (Hachisu et al. 1995). We have started large scale cosmological simulation with the MDM. One time-step takes less than 50 seconds for  $N = 3 \times 10^7$  with only 16 MDGRAPE-2 boards using tree based method.

**Acknowledgments.** This work is supported by the fund for “Heterogeneous Computer System” of the Ministry of Education, Culture, Sports, Science and Technology of Japan. Parts of this work were done during the Special Postdoctoral Researchers Program at RIKEN.

## References

- Barnes, J. & Hut, P. 1986, *Nature*, 324, 446
- Essmann, U., Perera. L., & Berkowitz, L. 1995, *J. Chem. Phys.*, 103, 8577
- Ewald, P.P. 1921, *Ann. Phys.*, 64, 253
- Fukushige, T., Makino, J., Ito, T., Okumura, S., Ebisuzaki, T., & Sugimoto, D. 1993, *PASJ*, 45, 361
- Greengard, L. & Rokhlin, V. 1987, *J. Comput. Phys.*, 73, 325
- Hachisu, I., Makino, J., Ebisuzaki, T., & Sugimoto, D. 1995, *Parallel Computational Fluid Dynamics: New Algorithms and Applications*, ed. N. Satofuka, J. Periaux, & A. Ecer (Elsevier Sciences), 155
- Hockney, R. W. & Eastwood, J. W. 1981, *Computer Simulation Using Particles* (New York: McGraw-Hill)
- Makino, J. 2001, in *IAU Symp. 208, Astrophysical Supercomputing using Particle Simulations*, ed. P. Hut & J. Makino
- Narumi, T., Susukita, R., Ebisuzaki T., McNiven G., & Elmegreen B. 1999, *Mol. Sim.*, 21, 401
- Narumi, T., Susukita, R., Furusawa, H., & Ebisuzaki, T. 2000, in *Proceedings of the 5th International Conference on Signal Processing*
- Narumi, T., Kawai, A., & Koishi, T. 2001, in *Proceedings of SC2001, High Performance Computing and Networking*
- Steinmetz, M. 1996, *MNRAS*, 278, 1005
- Strain, J. 1992, *J. Comput. Phys.*, 99, 251
- Sugimoto, D., Chikada, Y., Makino, J., Ito, T., Ebisuzaki, T., & Umemura, M. 1990, *Nature*, 345, 33
- Taiji, M., Fukushige, T., Makino, J., Ebisuzaki, T., & Sugimoto, D. 1994, in *Proceedings of the 6th Joint EPS-APS international conference on Physics Computing*, 200
- Tosi, M. P. & Fumi, F. G. 1964, *J. Phys. Chem. Solids*, 25, 45
- Umemura, M. 1993, *ApJ*, 406, 361