

Parallel Molecular Dynamics Simulation on Elastic Properties of Solid Argon

Futoshi Shimizu, Hajime Kimizuka, Hideo Kaburaki
Center for Promotion of Computational Science and Engineering
Japan Atomic Energy Research Institute
Nakameguro 2-2-54, Japan 153-0061

shimizu@koma.jaeri.go.jp kimizuka@koma.jaeri.go.jp kaburaki@sugar.tokai.jaeri.go.jp

Ju Li, Sidney Yip
Department of Nuclear Engineering
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139
liju99@mit.edu syip@mit.edu

Abstract

Parallel Molecular Dynamics Stencil has been developed to execute effectively large-scale parallel molecular dynamics simulations. The Stencil is adapted to varieties of molecular dynamics simulations without special attention to parallelization techniques. As an example of large-scale simulation using this Stencil, the adiabatic elastic constants of solid argon in crystalline and amorphous states, have been evaluated over the temperature range of 15–75K. An MD model of solid argon, consisting of 500–1,000,188 atoms interacting via the Lennard-Jones potential, has been used throughout the simulations. It is found that the cutoff-length is critical in evaluating elastic constants and that the calculated values for the temperature dependence of crystalline argon agree well with the measured ones.

1 Introduction

Molecular dynamics (MD) method is widely employed to predict macroscopic properties from atomistic motions in diverse applications such as physics, chemistry, and material science. In particular, equilibrium molecular dynamics is used to predict physical properties of matter, such as macroscopic thermodynamics quantities, atomistic structural properties, and dynamical quantities of transport coefficients [1].

In today's computing environment, the distributed computing is the way to extend the system to a large scale. Parallelization is the basis towards this future distributed computing. Many parallel MD simulation programs, such as AL_CMD [2], IMD [3], Moldy [4], NAMD [5], and SPaSM [6], have been developed and published. Almost all of these programs introduce one specific parallel algorithm while there are many kinds of parallel algorithms for MD. In this paper, we take up two parallel algorithms, that is, particle decomposition and spatial decomposition methods, and make a comparison between them by our newly developed MD program, Parallel Molecular Dynamics Stencil [7]. We also report an MD simulation of solid argon as an application of the Stencil.

2 Parallel MD simulation

In the method of equilibrium molecular dynamics, the size effect would become more important when we derive the statistically averaged results, in particular, in the lower temperature region. The introduction of the massively parallel supercomputers leads us to the possibility of attacking larger systems, but also to the involvement in the complexities of parallel calculations along with choosing varieties of potentials and statistical ensemble methods for the MD method. In this section, we describe an empirical Lennard-Jones (LJ) potential, cutoff schemes, and parallel algorithms.

2.1 Lennard-Jones potential

As a model system for the method of MD simulation, a system interacting with the Lennard-Jones pair potential has long been used to elucidate the fundamental properties of many-body interacting particles. The standard Lennard-Jones 12-6 potential for argon below is suitable for measuring all the computational efficiency of the parallel MD simulation.

$$\phi(r) = 4\varepsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^6\right], \quad (1)$$

where parameters $\sigma (= 0.34[\text{nm}])$ and $\varepsilon (= 120[K] \times k_B)$ are the distance to the zero in $\phi(r)$ and the energy at the minimum in $\phi(r)$, respectively. Also, the prediction of elastic constants has been found to be a good benchmark to test the accuracy of the empirical potential and the numerical efficiency of the MD method [8, 9].

2.2 Cutoff schemes

In a system of N particles interacting via pair potential, there are $N(N-1)/2$ calculations in total. Thus, the cost of force calculation increases with the square of N , and almost all execution time of MD simulation is taken up in the calculation of force as the system size becomes larger. The cutoff distance r_c is introduced in the LJ system to save the computational cost. The shifted-force potential is used throughout this paper where pair interactions beyond r_c are neglected and the discontinuity involved in calculating the force is removed at the cutoff point.

$$\mathbf{f}(\mathbf{r}) = \frac{d\Phi}{dr}\hat{r} = \begin{cases} -\frac{d\phi}{dr}\hat{r} + \frac{d\phi}{dr}\Big|_{r_c}\hat{r} & (r < r_c) \\ 0 & (r \geq r_c), \end{cases} \quad (2)$$

$$\Phi(r) = \begin{cases} \phi(r) - \phi(r_c) - (r - r_c)\frac{d\phi}{dr}\Big|_{r_c} & (r < r_c) \\ 0 & (r \geq r_c). \end{cases} \quad (3)$$

The pair list is introduced for the effective calculation of the shifted-force potential system. The one method is to use the typical Verlet neighbor list, where each (i -th) particle in a system has a list of neighboring (j -th) particles whose positions are within a distance of r_{list} (Figure 1(a)). The distance r_{list} is taken as the sum of r_c and some buffer margin. The updating interval of the neighbor list is determined by this buffer margin. With this method, the computational cost is proportional to the number of total particles N and the numbers of particles in the list which is independent of N .

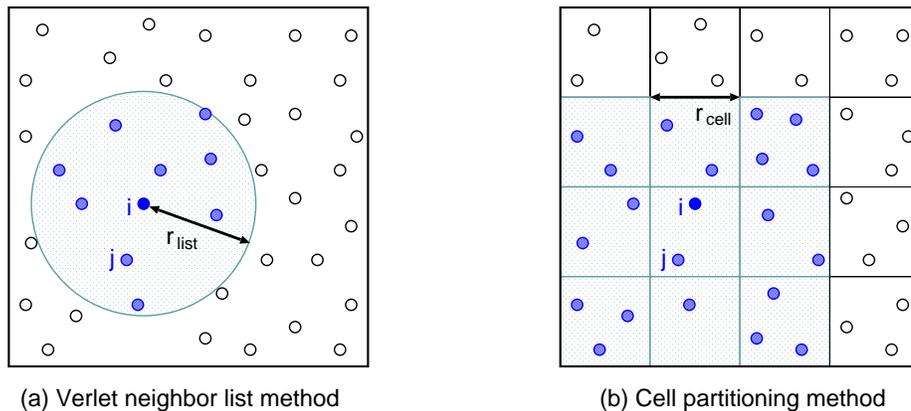


Figure 1: Cutoff schemes for short range interactions.

The other method is cell partitioning. The simulation space is divided into rectangular cells whose side length r_{cell} is taken larger than r_c (Figure 1(b)). In this method, the pair list for an i -th particle in a particular cell can be constructed by searching particles in the cell containing the i -th particle and the contacting neighboring cells. The computational cost for this method becomes also proportional to the number of total particles N and the numbers of particles in cells which is independent of N .

2.3 Parallel algorithms

Here we consider the parallelization of the MD simulation program, in which the data structures such as positions and velocities of particles and the computing procedures such as calculations of forces and updates of particle positions are partitioned and allotted to each processor. One of the key factor to be taken into account for the parallel computation is the communication between processors. The communication time, consisting of the data transfer time and the start-up time, always tends to reduce the parallel efficiency. Also, in the case that communications among processors are synchronous, load imbalance, in which one processor with the job done has to wait other processors to finish their jobs, deteriorates the parallel efficiency. Since the method of partitioning the data and computational procedures is closely related, a good parallel MD program should be made in such a way to lessen the communication data and frequencies and to realize a good load balance among processors. Next we describe the typical partitioning methods, particle and spatial decomposition method [10].

In the particle decomposition method, particles are globally numbered, partitioned, and allotted to processors, as shown in Fig.2(a). Basically, each processor takes care of the allotted particles from the beginning to the end of the calculation. Here, for simplicity of the program, the replicated data method is used where all processors store and share positions of particles for the whole system. It is predicted to be disadvantageous that scalability in this system deteriorates as the number of processors increases, since the synchronous global communication of broadcasting is required. On the other hand, load balance among processors can be easily adjusted to be uniform by allotting the same number of particles to each processor.

In the spatial decomposition method, communication cost can be reduced using the locality of particles, that is, by allotting particles belonging to the partitioned space to each processor (see Fig.2(b)). In this method, communication between neighboring processors is required to calculate the force, because particles near the boundary region interact with those in neigh-

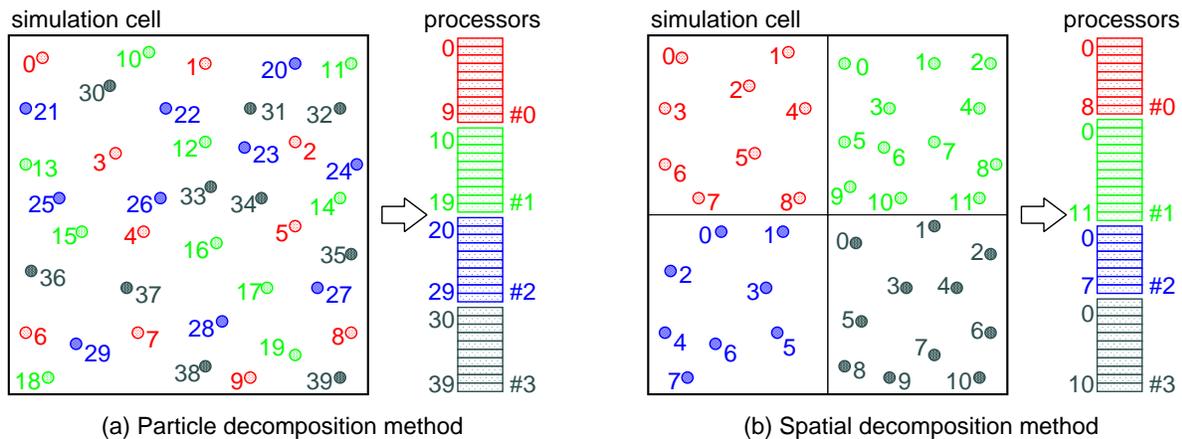


Figure 2: Two decomposition methods for parallel MD; each forty particles are allotted to 4 processors.

boring regions. Also, sending and receiving particles are necessary among processors when particles move across the boundary. Scalability for the program execution time as a function of the number of processors maintains well since the communication is restricted to local areas. Load balance among processors is generally good except that the distribution of particles is extremely imbalance.

2.4 Development of Parallel Molecular Dynamics Stencil

It has not been straightforward to implement the parallel algorithms methods, into the MD program, writing necessary procedures for data communication explicitly. Combination with the cutoff schemes for efficient computation of short range particle interactions makes the programming furthermore difficult and cumbersome. We have been developing Parallel Molecular Dynamics Stencil [7], which consists of parallel model programs for executing MD simulations by incorporating various statistical ensembles methods to realize constant temperature and constant pressure conditions, and the integration algorithms such as velocity Verlet, Beeman, and five-value Gear methods. The Stencil is designed in such a way to separate and hide parts of the programs for cutoff schemes and parallel algorithms, so that users can concentrate on the numerical simulation itself. Although the whole program is written in C using MPI [11], users do not need to know how to use each MPI function.

In the present version of the Stencil, the pair list for cutoff schemes is taken as the interface to calculate force. This part of the program can be revised according to other physical models. With the Stencil's framework, parallel programming can be done in the same way as sequential programming for a single processor. This stage of programming is designed to be independent of other programming stages of cutoff schemes, integration algorithms, and statistical ensembles. Any of these routines can be easily linked to construct a parallelized executable program.

2.5 Performance of Parallel MD Stencil

The rare-gas elements are good examples for the calculation of MD properties since they have comparatively simple structures, and weak (van der Waals) forces can be represented by well-defined short-range interactions. We have evaluated performance of Parallel MD Stencil on Intel Paragon XP/S75 with argon atoms interacting via the shifted-force LJ potential of

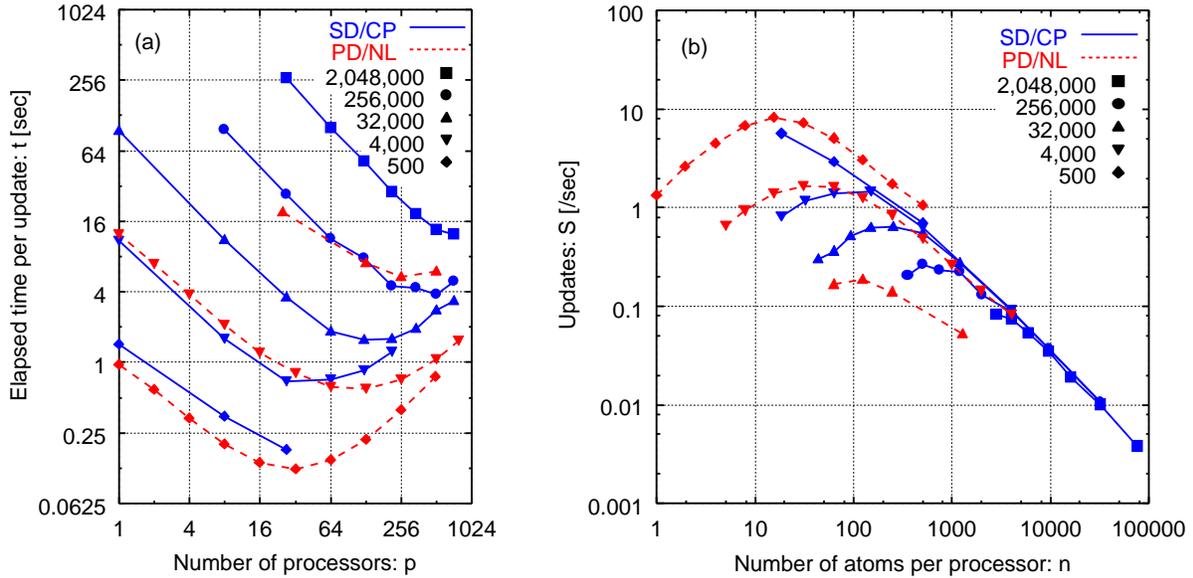


Figure 3: Performance of Parallel MD Stencil; (a) time versus number of processors and (b) speed versus number of atoms per processor are shown by two methods, spatial decomposition with cell partitioning method (solid line) and particle decomposition with Verlet neighbor list method (dashed line).

Eq.3. Here, Nosé-Hoover [12] thermostat and Parrinello-Rahman [13] constant-stress method are incorporated to realize the NTP ensemble. We chose the cutoff length to be $2.5\sigma(0.85[\text{nm}])$ and measured the performance over many cases of simulations using atoms ranging from 500 to 2,048,000.

Figure 3(a) shows elapsed time per update of parallel execution by two methods; spatial decomposition with cell partitioning method (solid line) and particle decomposition with Verlet neighbor list method (dashed line). Because of the limitation of memories of each processor, the maximum number of atoms that can be treated by the particle decomposition method is 32,000 in the present computing environment. As the total number of atoms decreases, there is an upper limit to the number of processors that can be used in the spatial decomposition method. This limitation comes from the condition of interaction range, in which the side length of processor cell has to be taken larger than r_c . On the other hand, in the particle decomposition method, it is noted that, in principle, even one atom can be assigned to each processor without regard to its efficiency. Figure 3(b) represents the speed S as a function of the number of atoms per processor n . In the spatial decomposition method, scalability is confirmed by the fact that data for different total number atoms collapse onto a single diagonal line in the region of $n > 1,000$. This is because the region for search to make the pair list based on cutoff algorithm is restricted to local areas. By ignoring inter-processor communications, the execution time per update is estimated to be $t_{SD} = C(An + Bn)$, where A, B, C are the constants. Here, An and Bn stand for costs for force calculation and pair list making, respectively. The logarithm of the number of updates per second $S_{SD} = 1/t_{SD}$ is written as

$$\log S_{SD} = -\log n + D, \quad (4)$$

where $D = -\log C - \log(A + B)$.

In the particle decomposition method, on the other hand, there are three different lines for cases of 32,000 atoms, 4,000 atoms, and 500 atoms. Since all atoms are objects of search for a

pair list construction, time per update is estimated to be $t_{PD} = C'(A'n + qB'n)$, where A', B', C' are the constants. A parameter $q = N/n$ is nearly equal to the number of processors p . $A'n$ and $qB'n$ stand for costs for force calculation and pair list making, respectively. In this case, data for different total number atoms do not conform to a single line, because the logarithm of the number of updates per second $S_{PD} = 1/t_{PD}$ depends on q :

$$\log S_{PD} = -\log n + D'(q). \quad (5)$$

Since $D'(q) = -\log C' - \log(A' + qB')$ is a function of q which depends on N , there appear three separate lines for each N . In addition, owing to the collective communications which involve all processors, the deviation from the diagonal line is observed for each N . In the case of 500 atoms, it is noted that the updates for the particle decomposition method are larger than those for the spatial decomposition method.

With all the performance described above, it becomes straightforward to extract the best performance from your computing environment when the parameters for simulation, such as number of atoms, cut-off length, etc., are fixed. The Parallel MD Stencil is designed in such a way to adapt easily to a wide range of parameter space.

3 Elastic Properties of Solid Argon

The spherically symmetric rare-gases atoms crystallize with face-centered cubic (fcc) structures, and their solids have been the subject of many theoretical investigations. Also, a quantitative agreement has been obtained for several of the experimentally determined properties. In this study, we focused on the elastic properties of crystalline and amorphous argon solid, and investigated these system-size effect to show the effectiveness of parallel computing.

3.1 Elastic property of crystalline argon

We have performed MD simulations of the crystalline argon using several techniques employed in the Parallel MD Stencil. We started from the model system in the fcc structure (500–1,000,188 atoms in a cubic cell with periodic boundary conditions, interacting via the shifted-force LJ potential of Eq.3 with the cutoff-length $r_c = 2.5\sigma$ or 4.1σ), and heated the system continuously to the desired temperature. The structures of the reference states at the desired temperature and stress were determined in the *NTP* ensemble using the Nosé-Hoover [12] thermostat and Parrinello-Rahman [13] constant-stress methods. The adiabatic elastic constants C_{ij} at each temperature were calculated within the constant-volume and constant-energy (*NVE*) ensemble. The fluctuation formula for the internal stress tensor [14] was used to obtain a complete set of elastic constants over the temperature range of 15–75K.

Figure 4 shows the temperature dependence of the C_{ij} 's of crystalline argon. The calculated C_{11} has a smaller temperature dependence than that of the experiment whereas the calculated values of C_{12} and C_{44} tend to have a larger temperature dependence [15]. However, it is confirmed that the overall temperature dependence is well reproduced. In particular, the model agrees with experiment in predicting that C_{44} decreases larger with temperature than C_{12} . It is generally known that the Cauchy relation $C_{12} = C_{44}$ holds in the cubic symmetry if the atoms interact with central forces such as the LJ potential. (Also, the Poisson ratio will be estimated to be 1/4.) In Fig. 4, the calculated C_{12} and C_{44} values are distinct from each other,

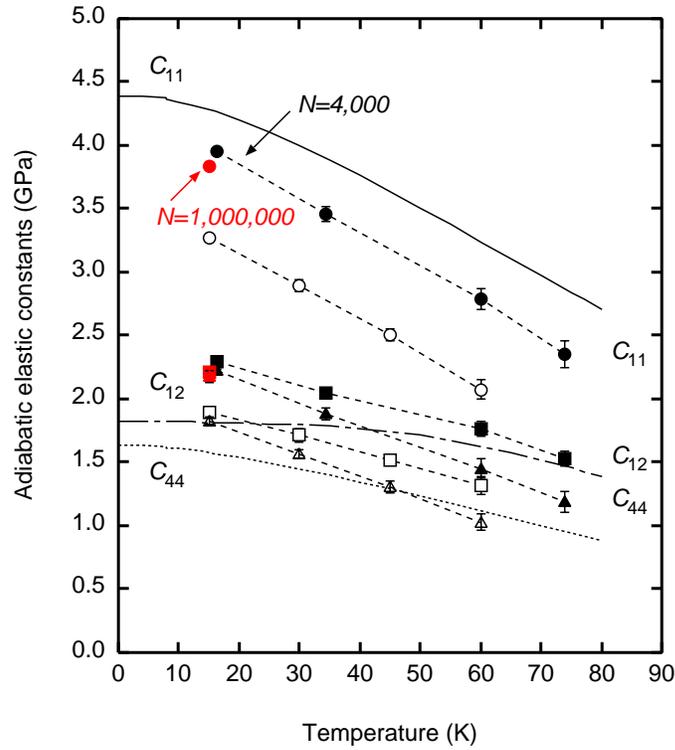


Figure 4: Temperature evolution of the adiabatic elastic stiffness constants of crystalline argon. The calculated elastic constants for $r_c = 4.1\sigma$ and 2.5σ are shown as solid and open plots, respectively; C_{11} (circle), C_{12} (square), and C_{44} (triangle). The experimental values for crystalline argon [15] are shown by curves.

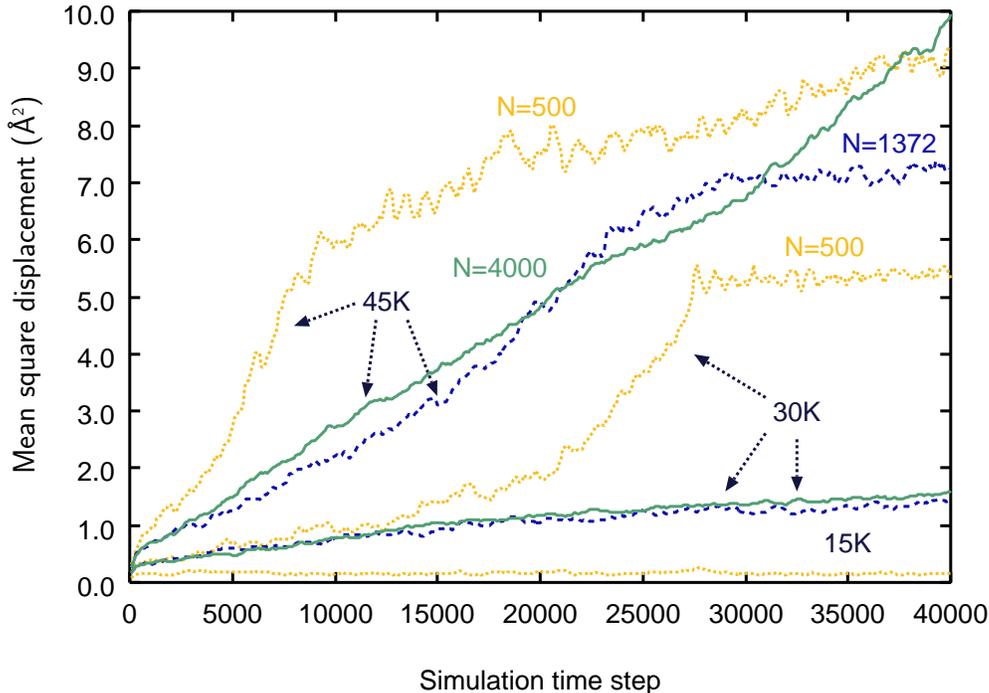


Figure 5: Time evolution of the mean-square displacements of argon atoms.

and this result agrees with experiment. It indicates that the present results reproduce the finite temperature effects in an exact manner. However, the Cauchy relation is shown to hold in the extremely low-temperature region, since the elastic constants are expressed by the fluctuation of a certain ensemble and this fluctuation approaches zero in the limit of $T=0\text{K}$.

Comparing the two cutoff-lengths, the C_{ij} values for $r_c=2.5\sigma$ are significantly smaller than those for $r_c=4.1\sigma$, and the temperature dependence is underestimated in this temperature range. It depends on the number of interactions taken into account to evaluate the elastic constants. The cutoff-length effect is important for the elastic properties and the relatively long-range interactions are recommended in this method.

3.2 Elastic property of amorphous argon

An MD model of amorphous argon has been obtained from a model of a supercooled liquid ($T=100\text{K}$) by relaxing in the NTP ensemble at constant zero pressure using the above-mentioned methods.

Figure 5 shows the time evolution of the mean-square displacements (MSD) of argon atoms. Overall, the MSD values increase linearly with time, and the slope of the MSD curve, which is equivalent to the diffusion coefficient, becomes steep as the temperature increases. It suggests that the argon atoms be loosely jointed, and drift gradually keeping the amorphous states. Rapid changes in the MSD curves indicate the transformation to the more stable phase due to crystallization. Moreover, it reveals that the duration (lifetime) of amorphous state depends on the system size.

Figure 6 shows the temperature dependence of the C_{ij} 's of amorphous argon. It is found that the C_{ij} values of the model of amorphous argon satisfy adequately the condition for an isotropic material: $C_{44} = \frac{1}{2}(C_{11} - C_{12})$. It is noted that C_{44} is significantly small compared with the

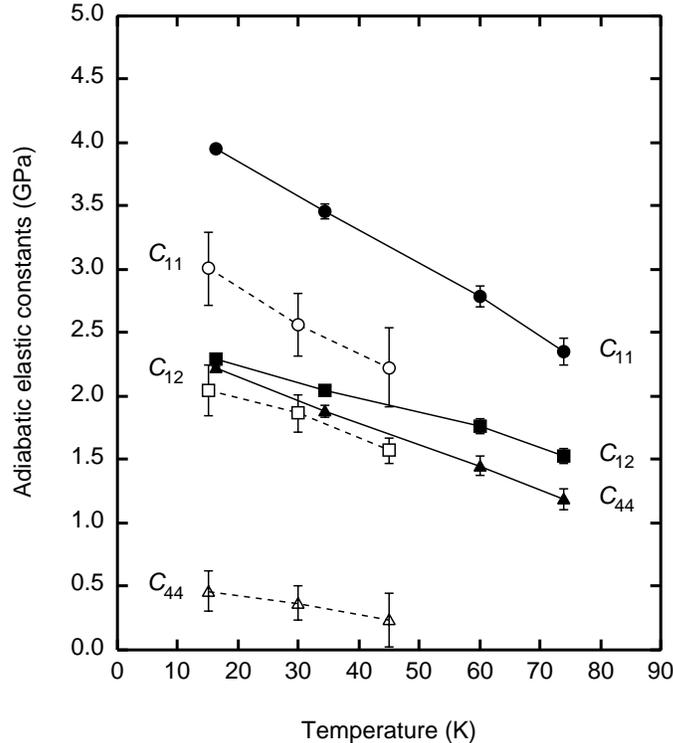


Figure 6: Temperature evolution of the adiabatic elastic stiffness constants of solid argon. The calculated elastic constants of crystalline and amorphous argon are shown as solid and open plots, respectively; C_{11} (circle), C_{12} (square), and C_{44} (triangle).

other components, and a Poisson ratio exhibits the value close to the isotropic upper limit of 0.5, as found in rubbery materials. The C_{ij} values tend to decrease with temperature, and C_{44} particularly disappears (approaches zero) at around 60K. It indicates that the obtained solid phase becomes mechanically unstable and transforms to a liquid phase at this temperature. It contrasts with the fact that argon solids remain in the fcc structure until about 84K (melting temperature).

4 Conclusions

We have implemented atom and spatial decomposition methods into Parallel Molecular Dynamics Stencil. By performing the evaluation analysis of the Stencil using argon atoms, a guiding principle in parallel MD simulation has been obtained; the spatial decomposition method achieves a good parallel efficiency for system with a large number of atoms, while the particle decomposition method is efficient for systems with a small number of atoms.

The Parallel MD Stencil was found to be useful in simulations for searching parameters in a wide range, in particular, changing the system size. It is found that the cutoff-length of at least $r_c = 4.1\sigma$ is needed in evaluating the elastic properties of argon solid. The simulation reproduces well the absolute values and the temperature dependence of adiabatic elastic constants of crystalline argon, and, with the same method, predicts elastic constants of amorphous argon.

References

- [1] J. M. Haile, *Molecular Dynamics Simulation: Elementary Methods*, A Wiley-Interscience Publication (1992).
- [2] Ames Lab Classical Molecular Dynamics Source Code, http://cmp.ameslab.gov/cmp/CMP_Theory/cmd/alcmd_source.html.
- [3] The ITAP Molecular Dynamics Package Homepage, <http://www.itap.physik.uni-stuttgart.de/~imd/>.
- [4] A GENERAL-PURPOSE MD CODE, <http://www.earth.ox.ac.uk/~keith/moldy.html>.
- [5] NAMD – Scalable Molecular Dynamics, <http://www.ks.uiuc.edu/Research/namd/>.
- [6] SPaSM's home, <http://bifrost.lanl.gov/MD/MD.html>.
- [7] F. Shimizu, H. Kimizuka and H. Kaburaki, *Proc. of the Conf. on Comp. Eng. and Sci.* 5, 361 (2000), (in Japanese).
- [8] M. Tang and S. Yip, *J. Appl. Phys.* 76, 2719(1994).
- [9] H. Kimizuka, H. Kaburaki, and Y. Kogure, *Phys. Rev. Lett.* 84, 5548 (2000).
- [10] D. M. Beazley, et.al, *Parallel Algorithm for Short-Range Molecular Dynamics*, *World Scientific's Annual Reviews in Computational Physics*, 3, 119 (1995).
- [11] MPI - The Message Passing Interface Standard, <http://www-unix.mcs.anl.gov/mpi/>.
- [12] S. Nosé, *Mol. Phys.* 52, 255 (1984); W. G. Hoover, *Phys. Rev. A* 31, 1695 (1985).
- [13] M. Parrinello and A. Rahman, *J. Appl. Phys.* 52, 7182 (1981).
- [14] J. R. Ray and A. Rahman, *J. Chem. Phys.* 80, 4423 (1984); J. R. Ray, *Comput. Phys. Rep.* 8, 109 (1988).
- [15] G. L. Keeler and D. N. Batchelder, *J. Phys. C: Solid St. Phys.* 3, 510 (1970).