Symplectic Molecular Dynamics Simulations on Specially Designed Parallel Computers

Urban Borštnik and Dušanka Janežič*

National Institute of Chemistry, Hajdrihova 19, SI-1000 Ljubljana, Slovenia

Received May 26, 2005

We have developed a computer program for molecular dynamics (MD) simulation that implements the Split Integration Symplectic Method (SISM) and is designed to run on specialized parallel computers. The MD integration is performed by the SISM, which analytically treats high-frequency vibrational motion and thus enables the use of longer simulation time steps. The low-frequency motion is treated numerically on specially designed parallel computers, which decreases the computational time of each simulation time step. The combination of these approaches means that less time is required and fewer steps are needed and so enables fast MD simulations. We study the computational performance of MD simulation of molecular systems on specialized computers and provide a comparison to standard personal computers. The combination of the SISM with two specialized parallel computers is an effective way to increase the speed of MD simulations up to 16-fold over a single PC processor.

1. INTRODUCTION

Molecular dynamics (MD) simulation is an important tool in studying biologically, physically, and chemically interesting systems.[1−3] To describe molecular motion, interactions among atoms are divided into bonding and nonbonding interactions. Bonding interactions take into account chemical bond lengths, valence angles, and dihedral angles. Nonbonding interactions describe interaction energies among atoms of different molecules or distant atoms within the same molecule, usually including electrostatic and van der Waals interactions.[4] To obtain more accurate results, simulations must be run for long periods of time on large systems.[5]

The computational time of a simulation can be reduced either by increasing the length of the simulation time step or by decreasing the computational time required for the time step.[6] We have developed a new, symplectic MD integration method that enables longer time steps, and therefore longer MD simulations are possible in the same run time.[7−9]

The computational time required for each simulation time step can be decreased using parallel computation. In parallel algorithms for MD simulations, the computation of each simulation step is divided among all the processors. Because the processing occurs in parallel, the program finishes faster than it would when running on a single processor.[10] Since the processors must frequently exchange data and each simulation time step is dependent on the previous step, load balancing and fast low-latency communication is crucial for obtaining good computational performance.[11,12]

Most recent parallel computers are often built as computing clusters, many of them Beowulf-type clusters of personal computers (PCs)[13] connected into a separate network serving as the primary processor interconnection.[14] The processors cannot directly access the memory on other PCs, meaning that PC clusters have a distributed-memory Multiple Instruction Multiple Data (MIMD) architecture.[13] Programs written for such a parallel architecture must use message passing for transferring all of the data among processors and care must be taken to reduce the communication time.[11,12,15]

Figure 1. Simplified schematic of the data transferred between the host computer and the MDGRAPE-2 daughter board. The MDGRAPE-2 is used for calculating nonbonding forces among a set of atoms, while the general-purpose PC processor performs all of the other calculations. The PC processor prepares an array of atomic coordinates for the MDGRAPE-2, which returns an array of the forces acting on the atoms.

Besides parallelization, specialized processors may also be used to increase computational speed.[16,17] The MDGRAPE-2 (Molecular Dynamics Gravity Pipeline) processor is designed for the fast calculation of nonbonding interactions in MD simulations.[18,19] Since the calculation of nonbonding interactions represents the bulk of the calculations in an MD simulation, the computational speed of the simulation is greatly increased by using the MDGRAPE-2. The use of multiple MDGRAPE-2 processors in parallel has been shown to increase the speed of MD simulations with standard MD methods.[17]

The MDGRAPE-2 processor is implemented as a PCI daughter board that works in conjunction with a host PC. The MDGRAPE-2 processor uses 32-bit floating point numbers internally for the force calculations.[19] Using more than one MDGRAPE-2 daughter board on one PCI bus would be inefficient because the combined data transfer would oversaturate the bus.[18] The MDGRAPE-2 processor is used only as a coprocessor for the calculation of nonbonding interactions; the general-purpose CPUs in the host computer are used for all of the remaining calculations. A simplified scheme of the data flow between the host computer and the MDGRAPE-2 is illustrated in Figure 1. The CPU communicates with the MDGRAPE-2 processor

* Corresponding author e-mail: dusa@cmm.ki.si.

10.1021/ci050216q CCC: $30.25
Published on Web 10/12/2005

by sending it data arrays of atomic positions and scaling factors such as charges or van der Waals parameters. Also sent is a user-defined function that is used to evaluate atom positions and their scaling factors. The CPU receives the result of the MDGRAPE-2 calculations as an array of forces acting on the corresponding input atoms.[18] The time required for the force calculations given atom position arrays of lengths n and m is $O(nm)$.

In this paper we present a parallel program we have developed, implementing the SISM (Split Integration Symplectic Method)[7−9] that runs on distributed-memory PC clusters and uses the MDGRAPE-2 for calculating nonbonding interactions. Both the calculation of bonding and nonbonding interactions as well as the integration is parallelized. With such an approach, we have achieved substantially faster computation of longer simulations.

In the next section we briefly characterize the SISM method for MD simulation and how it is implemented on parallel MDGRAPE-2 processors. In the following section we describe the performance of the parallel MDGRAPE-2 processors in comparison to a standard parallel PC cluster for two model systems of liquid water.

2. METHODS

A parallel program was written implementing the SISM. A typical MD potential function,[4,20] which was used for the program, includes the following terms

$$H = \sum_i \frac{\vec{p}_i^{\,2}}{2m_i} + \sum_b k_b(b - b_0)^2 + \sum_\phi k_\phi(\phi - \phi_0)^2 + \sum_{i<j} \frac{e_i e_j}{r_{ij}} + \sum_{i<j} 4\epsilon_{ij}\left[\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12} + \left(\frac{\sigma_{ij}}{r_{ij}}\right)^{6}\right] \quad (1)$$

where i and j run over all the atoms, m_i is the mass and $\vec{p}_i$ is the linear momentum of atom i in the kinetic energy term, b_0 is the reference value for bond lengths and k_b is the force constant for bond b in the stretching term, ϕ_0 is the reference value for the angle and b_ϕ is the force constant for angle ϕ in the bending term, e_i is the charge on atom i in the Coulombic term, and ϵ_{ij} and σ_{ij} are the Lennard-Jones constants between atoms i and j in the van der Waals term.

A symplectic integration method for a Hamiltonian H means that for each fixed value of t the corresponding t-flow $\phi_{t,H}$ is a symplectic or canonical mapping, meaning that it preserves the differential form $D = d\vec{p} \wedge d\vec{q}$, which defines the symplectic structure in the phase space.[21,22] In the SISM the bonding interactions, which account for high-frequency motion in a molecule and thus limit the time step,[23−25] are treated analytically, while the nonbonding interactions are treated numerically. The SISM splits the Hamiltonian into two parts, $H = H_0 + H_r$, where H_0 is the pure harmonic part and H_r is the remaining part.[21,26] The SISM uses a generalized leapfrog Verlet scheme for MD integration.[20,27]

The nonbonding electrostatic and van der Waals interactions are modeled using the Coulomb and Lennard-Jones[4] potentials among all atoms in the molecular system and calculated with the MDGRAPE-2. We have also implemented the calculation of these potentials in software to facilitate comparison between the MDGRAPE-2 processor and the PC processors. In both cases, the forces among all pairs of atoms are calculated with no cutoff.

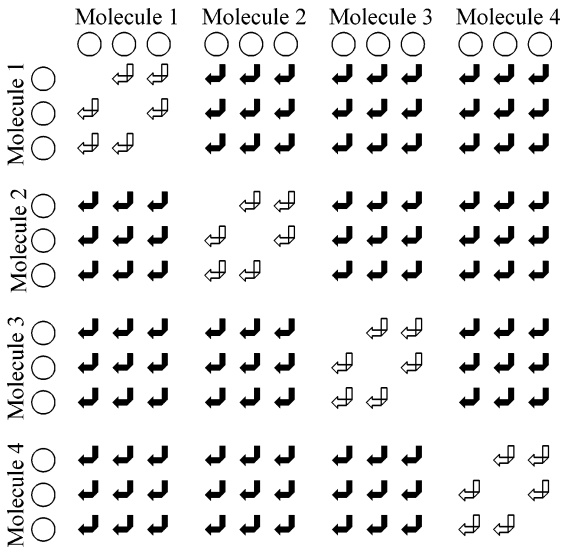


Figure 2. All of the interactions in a sample 12-atom molecular system. The diagram shows how each of the 12 atoms interacts with the remaining 11 atoms: every row represents the forces exerted by 11 atoms (at the top) onto the atom at the left of that row. A diagonal symmetry is implied, since the interaction between any two atoms is equal but opposite. The interactions drawn in white represent intramolecular interactions (analytical part), while the interactions drawn in black represent the numerous intermolecular interactions (numerical part).

For any molecular system the interactions among all of the atoms must be calculated. Explicitly calculating every one for a system of n atoms requires n^2 calculations.[4] All the interactions in a model system of four triatomic molecules are represented in Figure 2. The 12 atoms along the top and left are the same but drawn twice to illustrate the interactions: the atoms at the top exert forces onto the atoms along the left, as the arrows indicate. Black arrows represent interactions between atoms in different molecules; these interactions are treated as nonbonding interactions. White arrows represent intramolecular interactions treated as bonding interactions.

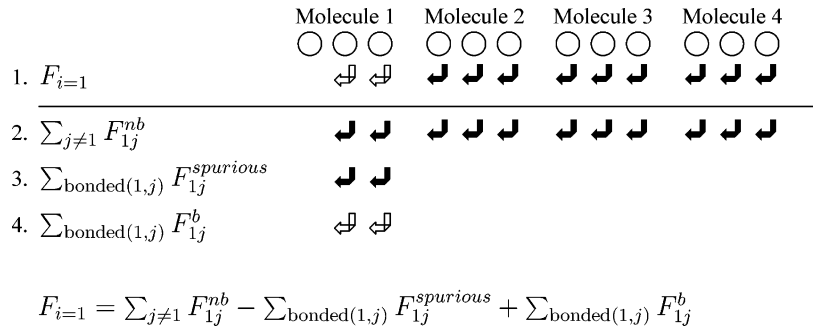
The force F_i on atom i is the sum of the individual forces F_{ij} of all other atoms $j \neq i$ on atom i:

$$F_i = \sum_{j \neq i} F_{ij} \quad (2)$$

In Figure 2, each of the arrows in any row i corresponds to one of the forces F_{ij}. The sum of these forces in row i is F_i. Each of the forces arises either from a bonding (F^b) or nonbonding (F^{nb}) interaction

$$F_i = \sum_{\text{bonded}(i,j)} F_{ij}^{b} + \sum_{\text{nonbonded}(i,j)} F_{ij}^{nb} \quad (3)$$

where the first sum over *bonded* (i, j) sums all pairs of atoms bonded to atom i and the second sum over *nonbonded* (i, j) sums all pairs of atoms not bonded to atom i. In the implemented program, forces F^b are calculated analytically from the stretching and bending potentials, while forces F^{nb} are calculated numerically from the Coulomb and Lennard-Jones potentials using the MDGRAPE-2.



$$F_{i=1} = \sum_{j\neq 1} F_{1j}^{nb} - \sum_{\mathrm{bonded}(1,j)} F_{1j}^{spurious} + \sum_{\mathrm{bonded}(1,j)} F_{1j}^{b}$$

Figure 3. The calculation of the total force $F_{i=1}$ acting on atom $i = 1$ (the first line from Figure 1). The force F_i (line 1) is composed of forces from bonding (white arrows) and nonbonding (black arrows) interactions. The true nonbonding forces are determined by calculating the nonbonding forces for all atom pairs with atom 1 (line 2) and then subtracting the spurious forces from bonding interactions calculated with nonbonding potentials (line 3). The bonding forces (line 4) are added to the true nonbonding forces to obtain the total $F_{i=1}$.

It should be noted that bonding interactions, which are calculated analytically with bonding potentials, should not also be recalculated with nonbonding potentials; however, to reduce the overhead of preparing atomic data when using the MDGRAPE-2, it is faster to calculate all interactions with nonbonding potentials and later subtract the spuriously calculated nonbonding potentials of bonded atom pairs.[19] The true force for an atom i arising from nonbonding interactions (the second term in the rhs of eq 3) is thus obtained in the following manner

$$\sum_{\mathrm{nonbonded}(i,j)} F_{ij}^{\mathrm{nb}} = \sum_{j\neq i} F_{ij}^{\mathrm{nb}} - \sum_{\mathrm{bonded}(i,\,j)} F_{ij}^{\mathrm{spurious}} \qquad (4)$$

where the lhs of eq 4 contains the desired true nonbonding forces, the first term of the rhs of eq 4 represents forces calculated from nonbonding potentials for all the atom pairs with atom i, and the second term of the rhs of eq 4 represents the forces due to the spuriously calculated nonbonding potentials of atom pairs bonded with i. The difference between the second term and the first is the desired nonbonding force: only those among atoms that are not bonded. Figure 3 shows an example of the calculations for obtaining the total force $F_{i=1}$ on atom 1 and corresponds to the first row in Figure 2.

The assumption when using the MDGRAPE-2 is that the calculation time of the rhs of eq 4 (both $\sum_{j\neq i} F_{ij}^{\mathrm{nb}}$ and $\sum_{\mathrm{bonded}(i,\,j)} F_{ij}^{\mathrm{spurious}}$) is less than the calculation time of the lhs of eq 4 ($\sum_{\mathrm{nonbonded}(i,\,j)} F_{ij}^{\mathrm{nb}}$). The calculation time for a system of n atoms is $an^2 + bn$ for the rhs where a and b are constants: calculating the force for all atom pairs takes an^2 time because the interaction is calculated for every pair of atoms, while the calculation of spurious forces takes bn time. For the lhs the time required is cn^2 for some constant c. Because the MDGRAPE-2 calculation time is longer if bonded interactions are explicitly excluded (corresponding to the lhs of eq 4) than if every atom pair is calculated with nonbonding potentials and the spurious nonbonding forces are later subtracted (corresponding to the rhs of eq 4), it follows that $c > a$. With an increasing system size the bn term becomes negligible compared to the n^2 terms ($O(n) < O(n^2)$) and so $an^2 + bn < cn^2$. For larger molecular systems it is therefore faster to calculate nonbonding interactions for every pair of atoms and later subtract such interactions for bonded atoms than it is to calculate nonbonding interactions only for the nonbonding atom pairs.

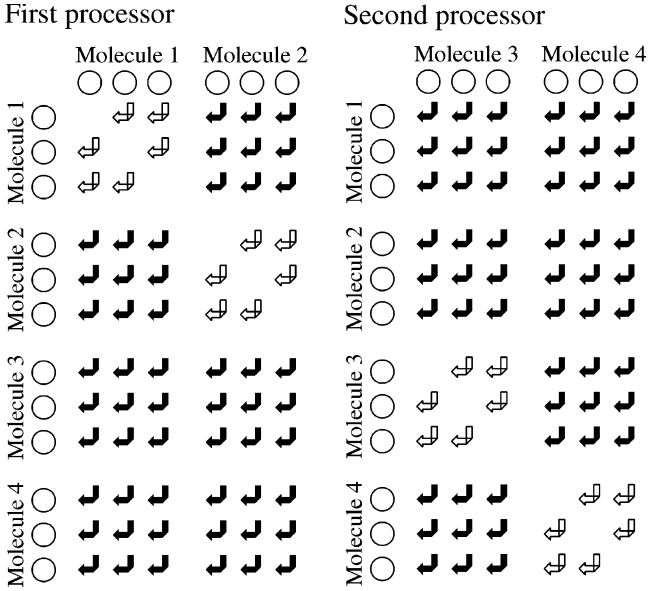


Figure 4. Atomic decomposition of a sample 12-atom molecular system from Figure 1. The calculation of the interactions is evenly distributed among the two processors. The forces exerted on each of the atoms, which are separately calculated on the two processors, are then combined to yield the total force exerted on the atom.

Atomic decomposition is used to parallelize the SISM.[10−12,28] In this parallelization scheme, all of the processors have the positions and other data for all of the atoms. The calculation of interactions and integration is divided among the processors by assigning each of the P processors n/P different atoms and n^2/P different atom pairs. Such an assignment of atom and atom pairs for the model system from Figure 2 is shown in Figure 4 for two parallel processors. In calculating the forces on atoms, each of the processors considers interactions from only n/P of the atoms on all n atoms in the molecular system, yielding n^2/P interaction calculations for each processor. The principle of dividing the computation of interactions among the processors remains the same when using multiple MDGRAPE-2s. The interactions that each of two MDGRAPE-2 calculates is depicted in Figure 4. The calculation of each force F_i for atom i (eq 2) is split into P parts (one for each processor), which are then summed at the end of the force calculations:

$$F_i = \sum_{(p=1)}^{P} \sum_{(n(p-1)/P<j\leq np/P)} F_{ij} \qquad (5)$$

Table 1. Simulation Computational Time, Parallel Computation Speedup (Labeled Parallel Speedup), and Speedup of the MDGRAPE-2 over a PC Processor (Labeled MDG2 Speedup), and Parallel Efficiency for Systems of 31 680 and 87 808 Liquid Water Molecules Using One and Two MDGRAPE-2 (Labeled MDG2) in Conjunction with One Standard PC CPU, as Well as 1, 2, 4, 8, and 16 Standard PC CPUs in Parallel on up to 8 Dual-Processor Computers with No MDGRAPE-2 Installed

number of processors		system of 31 680 water molecules				system of 87 808 water molecules			
CPU	MDG2	time [min]	parallel speedup	MDG speedup	efficiency [%]	time [min]	parallel speedup	MDG speedup	efficiency [%]
1	1	9.06		8.27		68.6		8.33	
2	2	4.80	1.89	15.6	94.3	34.7	1.98	16.47	98.9
1		74.96				571.31			
2		39.06	1.92		96.0	295.63	1.93		96.7
4		21.71	3.45		86.3	154.45	3.70		98.7
8		10.18	7.36		92.0	76.31	7.49		93.6
16		6.03	12.44		77.7	43.45	13.15		82.2

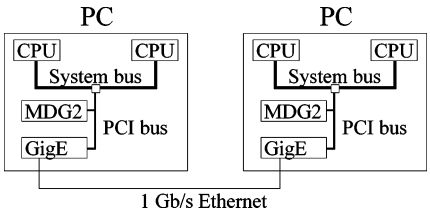


Figure 5. A diagram of a dual-PC cluster with two MDGRAPE-2 (labeled MDG2). Two processors share the system bus. The MDGRAPE-2s and the Ethernet interfaces (labeled GigE) share the PCI bus. Both PCs are connected with a dedicated 1 Gb/s Ethernet link.

The integration is also evenly divided among the *P* processors based on the assignment of atoms to processors described above.

To prevent load imbalance, the atoms and atom pairs should be evenly distributed among the *P* processors. Also, atoms belonging to the same molecule must be assigned to the same processor.

3. RESULTS AND DISCUSSION

The implemented parallel SISM MD program's computational performance was evaluated on two model systems of bulk liquid water,[9] one with 31 680 and the other with 87 808 water molecules, using a 2 fs integration time step and periodic boundary conditions with the minimum-image convention.[4] We compared the performance of the program using the MDGRAPE-2 to the performance of only the PC processors with no MDGRAPE-2. We also compared the program's performance when using multiple MDGRAPE-2 in parallel as well as using multiple PC processors. We also examined how the system size influences the efficiency of parallel computation.

All of the simulations were performed on dual-processor AMD MP-2200+ PCs. For simulations using the MDGRAPE-2, two such PCs, each with one MDGRAPE-2 daughter board, were connected with a 1 Gb/s Ethernet link, as shown in Figure 5. Only two MDGRAPE-2s were used due to the unavailability of more. Simulations using only PC processors with no MDGRAPE-2 were performed on a cluster of eight such PCs that had no MDGRAPE-2 installed. In this cluster the PCs are connected with gigabit Ethernet in a hierarchical hypercube topology.[29] Simulations with a single PC processor were performed on one computer, while simulations with 2, 4, 8, and 16 PC processors were performed on 1, 2, 4, and 8 dual-processor computers, respectively.

The computational times for the simulations of the two systems are listed in Table 1 for various numbers of MDGRAPE-2 (1 and 2) and PC processors (1, 2, 4, 8, and 16). Also shown are the speedups attained when using parallel computation on multiple processors as well as the efficiencies of the parallel computation. The speedup is defined as how much faster the program runs with *P* processors in parallel than on a single processor. For the simulations with the MDGRAPE-2, the speedup compared to a single PC processor is also listed. The efficiency is defined as the ratio between the theoretical minimum parallel computational time (which is equal to the computational time on one processor divided by *P*, the number of parallel processors used) and the actual parallel computational time.

Using one MDGRAPE-2, we achieved even an 8-fold speedup over the PC processor. Two MDGRAPE-2 resulted in a greater than 16-fold speedup compared to a single processor for the larger molecular system. The speedup with the MDGRAPE-2 compared to the PC processor increases with larger systems. This increase in speedup is attributed to a set up time needed for calculations with the MDGRAPE-2, which is a linear function of the size of the system size. Because the calculation time increases with the square of the system size, the setup time becomes shorter in comparison to the total simulation time of larger systems.

For both the MDGRAPE-2 and PC processors parallel execution is more efficient with the larger system. The efficiency is increased because the latency of the communication is constant for any size system and the communication time increases linearly with the system size. Compared to the computational time, which increases with the square of the system size, the communication time becomes a smaller fraction of the total time, leading to an increase in the parallel efficiency. This efficiency increase is especially noticeable with a greater numbers of PC processors. For the larger system, the efficiency also increases when four PC processors are used rather than two. While this may be attributed to a relatively shorter communication time compared to the computational time, the efficiency begins to decrease when a greater number of processors is used. It should be emphasized that a single MDGRAPE-2 remains approximately twice as fast even when compared to four PC processors.

The atomic decomposition method for parallel computation is well load-balanced.[10] Its drawback is the amount of communication required. Since every processor needs to have data for all of the atoms, the volume of data that must be transferred during each time step remains constant for any number of parallel processors. In comparison, the volume

of transferred data decreases with an increasing number of processors in two other MD parallelization techniques, spacial decomposition and force decomposition.[10,30,31]

4. CONCLUSIONS

We have demonstrated how the SISM for MD can be matched with a specialized parallel computer. In our approach the high-frequency vibrational motion corresponding to bonding interactions is treated analytically while the MDGRAPE-2 is used for the fast calculation of nonbonding interactions. We evaluate the parallel MDGRAPE-2 performance and compare it to the performance of standard PC processors. Combining the SISM, allowing for longer MD integration time steps, with specialized parallel computers, which reduce the computational time required for each simulation time step, substantially increases the speed of MD simulations. However, much work remains to be done in the development of approaches to effectively distribute the calculation of atomic interactions among the processors in parallel MD simulations.

ACKNOWLEDGMENT

The authors would like to thank Dr. T. Narumi for making available the MDGRAPE-2 hardware and his expertise to us as well as to Drs. M. Hodošček and M. Praprotnik for their helpful discussions. The financial support through grants P1-0002 and J1-6331 of the Ministry of Higher Education, Science, and Technology of Slovenia is acknowledged.

REFERENCES AND NOTES

(1) Allen, M. P.; Tildesly, D. J. *Computer Simulation of Liquids;* Oxford University Press: New York, 1987.
(2) Heermann, D. W.; Burkitt, A. N. *Parallel Algorithms in Computational Science;* Springer-Verlag: Berlin, 1991.
(3) Jorgensen, W. L. The many roles of computation in drug discovery. *Science* **2004**, *303*, 1812−1818.
(4) Leach, A. R. *Molecular Modelling: Principles and Applications;* Addison-Wesley Longman Limited: Essex, 1996.
(5) van Gunsteren, W. F.; Berendsen, H. J. C. Computer simulation of molecular dynamics: Methodology, applications, and perspectives in chemistry. *Angew. Chem., Int. Ed.* **1990**, *29*, 992−1023.
(6) Schlick, T.; Barth, E.; Mandziuk, M. Biomolecular dynamics at long timesteps: Bridging the timescale gap between simulation and experimentation. *Annu. Rev. Biophys. Biomol. Struct.* **1997**, *26*, 181−222.
(7) Janežič, D.; Praprotnik, M.; Merzel, F. Molecular dynamics integration and molecular vibrational theory. I. New symplectic integrators. *J. Chem. Phys.* **2005**, *122*, 174101.
(8) Praprotnik, M.; Janežič, D. Molecular dynamics integration and molecular vibrational theory. II. Simulation of non-linear molecules. *J. Chem. Phys.* **2005**, *122*, 174102.
(9) Praprotnik, M.; Janežič, D. Molecular dynamics integration and molecular vibrational theory. III. The IR spectrum of water. *J. Chem. Phys.* **2005**, *122*, 174103.
(10) Plimpton, S.; Hendrickson, B. Parallel Molecular Dynamics Algorithms for Simulation of Molecular Systems. In *Parallel Computing in Computational Chemistry*; Mattson, T. G., Ed.; Symposium Series 592 American Chemical Society: 1995.
(11) Trobec, R.; Šterk, M.; Praprotnik, M.; Janežič, D. Implementation and evaluation of MPI-based parallel MD program. *Int. J. Quantum Chem.* **2001**, *84*, 23−31.
(12) Trobec, R.; Šterk, M.; Praprotnik, M.; Janežič, D. Parallel programming library for molecular dynamics simulations. *Int. J. Quantum Chem.* **2004**, *95*, 530−536.
(13) Sterling, T.; Becker, D. J.; Savarese, D.; Dorband, J. E.; Ranawake, U. A.; Packer, C. V. Beowulf: A Parallel Workstation for Scientific Computation. In *Proceedings, International Conference on Parallel Processing*; Oconomowoc, WI, 1995.
(14) Hodošček, M.; Borštnik, U.; Janežič, D. CROW for large scale macromolecular simulations. *Cell. Mol. Biol. Lett.* **2002**, *7*, 118−119.
(15) Robič, B.; Vilfan, B. Improved schemes for mapping arbitrary algorithms onto processor meshes. *Parallel Computing* **1996**, *22*, 701−724.
(16) Narumi, T.; Kawai, A.; Koishi, T. An 8.61 Tflop/s Molecular Dynamics Simulation for NaCl with a Special-Purpose Computer: MDM. In *Proceedings of SuperComputing 2001*; Denver, 2001.
(17) Taiji, M.; Narumi, T.; Ohno, Y.; Futatsugi, N.; Suenaga, A.; Takada, N.; Konagaya, A. Protein Explorer: A Petaflops Special-Purpose Computer System for Molecular Dynamics Simulations. In *Proceedings of SuperComputing 2003*; Phoenix, 2003.
(18) Narumi, T.; Susukita, R.; Ebisuzaki, T.; McNiven, G.; Elmegreen, B. Molecular dynamics machine: Special-purpose computer for molecular dynamics simulations. *Mol. Simul.* **1999**, *21*, 401−415.
(19) Narumi, T. Special-purpose computer for molecular dynamics simulations, Doctor's thesis, University of Tokyo, 1998.
(20) Sanz-Serna, J. M.; Calvo, M. P. *Numerical Hamiltonian Problems;* Chapman & Hall: London, 1994.
(21) Janežič, D.; Merzel, F. An efficient symplectic integration algorithm for molecular dynamics simulations. *J. Chem. Inf. Comput. Sci.* **1995**, *35*, 321−326.
(22) Hardy, D. J.; Okunbor, D. I.; Skeel, R. D. Symplectic variable step size integration for N-body problems. *Appl. Numer. Math.* **1999**, *29*, 19−30.
(23) Brooks, B. R.; Janežič, D.; Karplus, M. Harmonic analysis of large systems. I. Methodology. *J. Comput. Chem.* **1995**, *16*, 1522−1542.
(24) Janežič, D.; Brooks, B. R. Harmonic analysis of large systems. II. Comparison of different protein models. *J. Comput. Chem.* **1995**, *16*, 1543−1553.
(25) Janežič, D.; Venable, R. M.; Brooks, B. R. Harmonic analysis of large systems. III. Comparison with molecular dynamics. *J. Comput. Chem.* **1995**, *16*, 1554−1566.
(26) Janežič, D.; Merzel, F. Split integration symplectic method for molecular dynamics integration. *J. Chem. Inf. Comput. Sci.* **1997**, *37*, 1048−1054.
(27) Verlet, L. Computer "experiments" on classical fluids. I. thermodynamical properties of Lennard-Jones molecules. *Phys. Rev.* **1967**, *159*, 98−103.
(28) Plimpton, S.; Hendrickson, B. A new parallel method for molecular dynamics simulation of macromolecular systems. *J. Comput. Chem.* **1998**, *17*, 326−337.
(29) Borštnik, U.; Hodošček, M.; Janežič, D. Improving the performance of molecular dynamics simulations on parallel clusters. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 359−364.
(30) Murty, R.; Okunbor, D. Efficient parallel algorithms for molecular dynamics simulations. *Parallel Computing* **1999**, *25*, 217−230.
(31) Snir, M. A note on n-body computations with cutoffs. *Theory Comput. Systems* **2004**, *37*, 295−318.

CI050216Q