Parallel Numerics '05, 223-231 M. Vajteršic, R. Trobec, P. Zinterhof, A. Uhl (Eds.)
Chapter 7: Systems and Simulation ISBN 961-6303-67-8

Large-Scale Molecular Dynamics Simulations on Parallel Clusters

Dušanka Janežič[1*], Urban Borštnik[1]

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

We have developed a parallel program for molecular dynamics (MD) simulation on clusters of personal computers (PCs). The program implements the Split Integration Symplectic Method (SISM) for MD integration that analytically treats high-frequency motions in molecules, allowing the simulation time step to be longer than in standard methods. The program is designed to run on parallel clusters of personal computers that we have built. Additionally, MD-GRAPE II processors, which are designed for fast MD simulations, can be used to calculate nonbonding interactions among atoms. Multiple such processors can be used in parallel, further reducing the computational time. We have compared the performance of the implemented program with several different numbers of processors using two different-sized molecular systems. The parallelization of the SISM and the use of the MD-GRAPE II is shown to be an effective combination.

1 Introduction

Molecular dynamics (MD) simulations are an important tool in studying biologically, physically, and chemically interesting systems. For many applications, a molecular mechanics description of the Hamiltonian is sufficient. In the molecular mechanics description, atomic interactions can be 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. The number of bonding interactions increase roughly linearly with the size of the system, while the number of

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

nonbonding interactions increase with the square of the size of the system. With a spherical cutoff scheme in which the interaction among distant atoms is ignored, the square dependence is reduced but nevertheless, nonbonding interactions remain the most computationally demanding part of the simulation. To obtain more accurate results, simulations must be run for long periods of time on large systems. To reduce the computational time required for a simulation, new integration methods must be developed to increase the simulation time step or to decrease the computational time required for each simulation time step.

Improved MD methods that analytically treat the bonding interactions allow longer simulation time steps to be used than with standard methods [1, 2]. Long time steps mean that fewer steps are needed for a simulation or that a longer simulation can be run for the same computational time as with standard methods.

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 the processors. Because the processing occurs in parallel, the program finishes faster than it would when running on a single processor. Since the processors must frequently exchange data and each simulation time step is dependent on the previous step, load balancing and low-latency fast communication among the processors is crucial to obtaining good computational performance [3, 4, 5].

Most recent parallel computers are clusters of computers, many of them Beowulf-type clusters of personal computers (PCs) [6]. They share a common distributed-memory Multiple Instruction Multiple Data (MIMD) architecture. Programs written for such a parallel architecture must use message passing for transferring all of the data among processors [4, 5].

Since the calculation of nonbonding interactions is the most computationally demanding part of the simulation, increasing its computational speed reflects on the speed of the whole program. The MD-GRAPE II processor is designed for the very fast calculation of nonbonding interactions [7, 8]. Since the calculation of these represents the bulk of the calculations in an MD simulation, the computational speed of the simulation is greatly increased.

We have developed a parallel program implementing the SISM that runs on distributed-memory PC clusters and uses the MD-GRAPE II processor for calculating nonbonding interactions. Bonding interactions, which account for high-frequency motion in a molecule and thus limit the time step, are treated analytically, while the nonbonding interactions are treated numerically. Both the calculation of bonding and nonbonding interactions is parallelized. With such an approach, it is possible to achieve significantly faster computation of

longer simulations.

2 The Split Integration Symplectic Method

The Split Integration Symplectic Method is an integration method for MD simulations. It splits the calculation of the Hamiltonian into two parts, the high-frequency bonding intramolecular part and the low-frequency nonbonding intermolecular part. It treats the high-frequency part completely analytically while the low-frequency remaining part is treated with standard numerical methods [9]. Because the highest-frequency motions that limit the simulation time step are treated analytically, the time step can be much longer than with standard methods that do not discriminate between high- and low-frequency motion [10]. The SISM's longer time step means that a longer simulation can be run for the same computational cost as with standard methods [11].

While the SISM calculates the bonding interactions analytically, they form only a small part of the total interactions in a molecular system. Their number is linearly dependent on the number of atoms in the system. The number of all the interactions, which must still be calculated, is dependent on the square of the number of atoms.

We have implemented a parallel algorithm for calculating nonbonding interactions that uses a cutoff radius with a neighbor list of atoms, meaning that only interactions among atoms within a certain cutoff distance are calculated. While using a cutoff radius when calculating nonbonding interactions means that the required calculations in each time step is linearly dependent on the number of atoms, calculating the neighbor list every few steps still requires a number of calculations that depends on the square of the number of atoms [12]. We have also implemented a parallel algorithm for calculating this list of neighboring atoms.

3 The CROW Clusters

We have build several CROW (Columns and Rows of Workstations) clusters targeted for running parallel MD programs [13]. They are comprised of PCs with standard Ethernet networking using various topologies. The newest cluster, CROW8, has a hierarchical hypercube topology. This topology takes advantage of the virtual hypercube topology of the MD program and its property that different amounts of data are transferred over different dimensions of the hypercube. The physical links of the topology are implemented as a mixture of the system bus of dual-processor PCs, a 1 Gb/s Ethernet switch, and point-to-point Gigabit Ethernet connections among pairs of PCs. A schematic

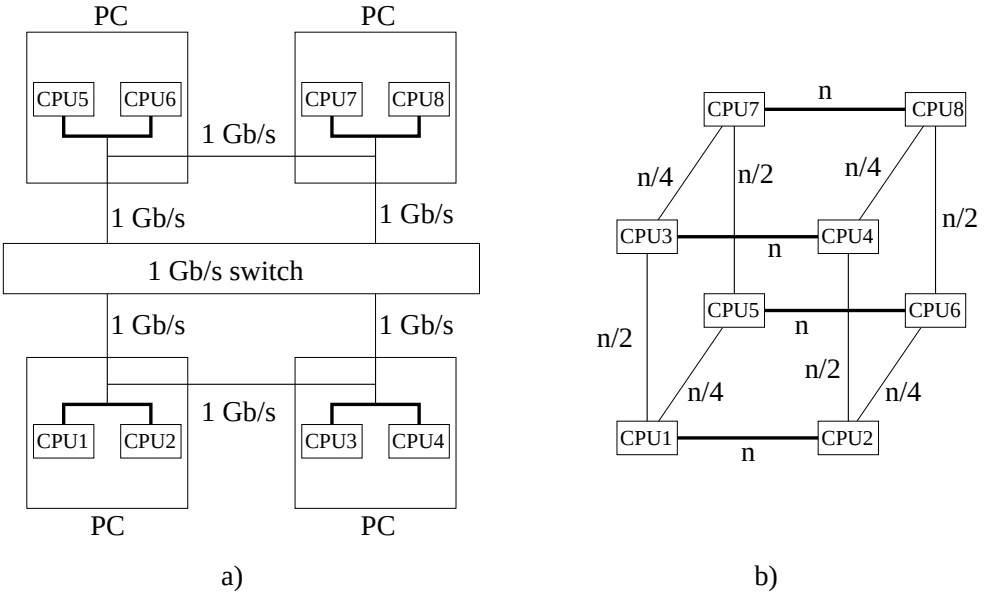

Figure 1: A schematic of the hierarchical hypercube topology of the CROW8 PC cluster showing the speeds of the different links for eight processors (a) and the amount of data transferred (n, $n/2$, $n/4$ for an arbitrary n) through the three corresponding dimensions (b).

of this topology is shown in Figure 1. Based on the premise that more data is transferred through the lower than higher dimensions, the links of lower dimensions have a higher speed than those of the higher dimensions. The first dimension has the fastest links, the system bus of an SMP PC. The second dimension is a 1 Gb/s link between two PCs. The third and higher dimensions are aggregated onto a 1 Gb/s switch [14].

4 Specialized Processors for MD Simulation

In addition to supporting clusters of PCS for parallel MD simulations, we have integrated support for MD-GRAPE II processors, which are specialized for calculating the nonbonding interactions in an MD simulation [7, 8]. Multiple MD-GRAPE II processors that run in parallel in different PCs are supported in the program for the parallel computation of nonbonding interactions. A representation of a cluster with two MD-GRAPE II processors is illustrated in Figure 2.

All the interactions among the atoms of a small sample system are shown

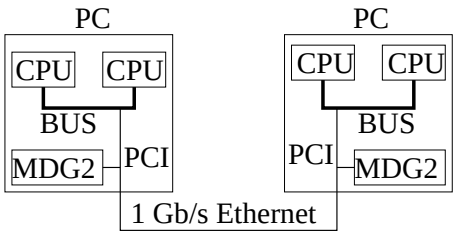

Figure 2: A schematic representation of the connections among two PCs and the connections among the two CPUs and the MD-GRAPE II processor (labeled MDG2) in each PC.

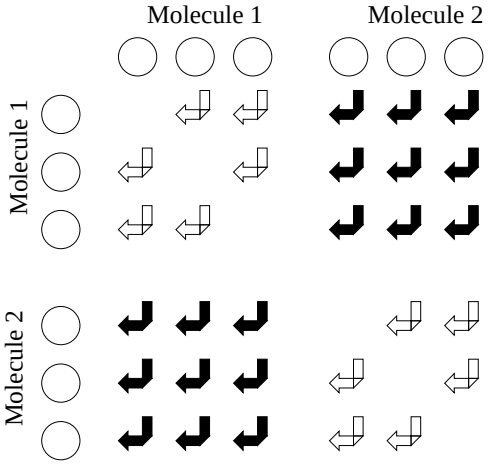

Figure 3: The interactions between the six atoms of a sample two-molecule system. The atoms in the top row and at the left are exactly the same atoms, the split being for illustration purposes. The arrows indicate the interactions (forces) calculated by the MD-GRAPE II processor. The atoms in the top row exert forces onto the atoms at the left. Black arrows indicate desired calculations, while the grayed arrows indicate intramolecular forces that are treated analytically and must therefore be subtracted.

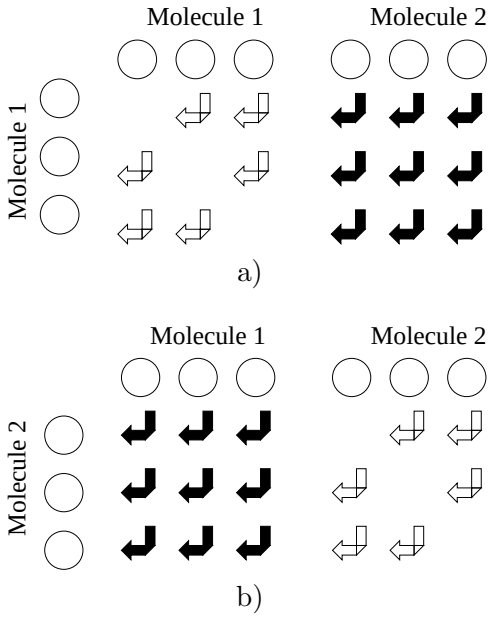

Figure 4: The division of the interaction (force) calculations from Figure 3 among the first (a) and second (b) MD-GRAPE II processor. While six atoms are still used to calculate the forces in both cases, there are only half as many atoms (three) onto which the forces are acting, thereby halving the number of computations performed by each of the MD-GRAPE II processors.

in Figure 3. The MD-GRAPE II processor calculates all of the interactions, indicated by arrows in the figure, among all the atoms in the system, regardless of whether they are bonding (white arrows) or nonbonding (black arrows). Since the bonding interactions are calculated by the SISM, the program must not use the results obtained by the MD-GRAPE II processor for these interactions. Their influence is cancelled by additionally calculating only these interactions and subtracting them from the results obtained by the MD-GRAPE II processor.

Two MD-GRAPE II processors in two interconnected personal computers (PCs) may be used for calculating nonbonding interactions. The two MD-GRAPE II processors each calculate half of the nonbonding interactions, after which the program shares the results with the other PC so they both have the same results. This division of interaction calculations from Figure 3 among two processors is shown in Figure 4.

Due to inherent memory limitations of the MD-GRAPE II processor, when a molecular system exceeds 200 000 atoms, the computation of interactions

Table 1: The elapsed time [minutes] for the MD simulation of systems of 87 808 (System 1) and 31 680 (System 2) water molecules using one MD-GRAPE II processor, two MD-GRAPE II processors in parallel, and one, two, and four standard PC CPUs with no MD-GRAPE II processors.

Processors	Elapsed Time [min]		Speedup	
	System 1	System 2	System 1	System 2
1 MD-GRAPE II	68.6	9.1	1.00	1.00
2 MD-GRAPE II	34.7	4.8	1.98	1.89
1 CPU	80.2	11.1	1.00	1.00
2 CPU	40.7	5.8	1.97	1.93
4 CPU	20.9	3.1	3.91	3.57
8 CPU	11.0	1.9	7.30	5.88

must be divided into smaller systems in a manner similar to the division used for parallelization.

5 Parallel Performance

To asses the parallel performance of the SISM implementation we have performed short simulations of two molecular systems. One was a system of 87 808 water molecules, while the other consisted of 31 680 water molecules. Simulations were performed on the CROW8 cluster both using the MD-GRAPE II processors and using only the host CPUs in the system. The elapsed time of the simulations and the speedup of using one and two MD-GRAPE II processors as well as using 1, 2, 4, and 8 CPUs of the cluster are presented in Table 1. The speedup indicates how much faster the program executes when running in parallel than when executing on a single processor.

Using two MD-GRAPE II processors in parallel yields a nearly double speedup for both systems. An MD simulation using both processors would therefore require virtually only half the time as when running on a single processor. The speedups of the parallel program when not using the MD-GRAPE II processors does not scale as well as when using the processors, especially for smaller systems.

The MD-GRAPE II processor outperforms the implementation not using the MD-GRAPE II, even though this implementation implements a cutoff when calculating nonbonding interactions and therefore does not calculate every pair of atomic interactions in each simulation time step.

6 Conclusions

An MD program implementing the SISM can be effectively parallelized. The SISM analytically treats bonding interactions and thus enables faster simulations since the simulation time step can be longer than with standard methods. In the parallel implementation of the SISM, the time required for each simulation step is decreased, further increasing the speed of the simulation. In addition, using MD-GRAPE II processors provides for very fast calculations of nonbonding interactions, increasing simulation speed even further. Parallelism between the MD-GRAPE II processor and the host CPUs remains to be implemented.

References

[1] Dušanka Janežič and Matej Praprotnik. Symplectic molecular dynamics integration using normal mode analysis. *Int. J. Quantum Chem. 84*, (2001), 2–12.

[2] Matej Praprotnik, Dušanka Janežič, and Janez Mavri. Temperature dependence of water vibrational spectrum: A molecular dynamics simulation study. *J. Phys. Chem. A 108*, (2004), 11056–11062.

[3] Dieter W. Heermann and Anthony N. Burkitt. *Parallel Algorithms in Computational Science*. Springer-Verlag, Berlin, 1991.

[4] Roman Trobec, Marjan Šterk, Matej Praprotnik, and Dušanka Janežič. Implementation and evaluation of MPI-based parallel MD program. *Int. J. Quant. Chem. 84* (2001), 23–31.

[5] Roman Trobec, Marjan Šterk, Matej Praprotnik, and Dušanka Janežič. Parallel programming library for molecular dynamics simulations. *Int. J. Quant. Chem. 95* (2004), 530–536.

[6] Thomas Sterling, Donald J. Becker, and Daniel Savarese. Beowulf: A parallel workstation for scientific computation. In *Proceedings, International Conference on Parallel Processing* (1995).

[7] Tetsu Narumi, Ryutaro Susukita, Toshikazu Ebisuzaki, Geoffrey McNiven, and Bruce Elmegreen. Molecular dynamics machine: Special-purpose computer for molecular dynamics simulations. *Mol. Sim. 21* (1999), 401–415.

[8] Tetsu Narumi. *Special-purpose computer for molecular dynamics simulations.* Doctor's thesis, University of Tokyo (1998).

[9] Dušanka Janežič, Matej Praprotnik, and Franci Merzel. Molecular dynamics integration and molecular vibrational theory. I. New symplectic integrators. *J. Chem. Phys. 122* (2005), 14 p.

[10] Matej Praprotnik and Dušanka Janežič. Molecular dynamics integration and molecular vibrational theory. II. Simulation of non-linear molecules. *J. Chem. Phys. 122* (2005), 9 p.

[11] Matej Praprotnik and Dušanka Janežič. Molecular dynamics integration and molecular vibrational theory. III. The IR spectrum of water. *J. Chem. Phys. 122* (2005), 10 p.

[12] Adrew R. Leach. *Molecular Modelling: Principles and Applications*. Addison Wesley Longman Limited, 1996.

[13] Milan Hodošček, Urban Borštnik, and Dušanka Janežič. CROW for large scale macromolecular simulations. *Cell. & Mol. Bio. Lett. 7* (2002), 118–119.

[14] Urban Borštnik, Milan Hodošček, and Dušanka Janežič. Improving the performance of molecular dynamics simulations on parallel clusters. *J. Chem. Inf. Comput. Sci. 44* (2004) 359–364.