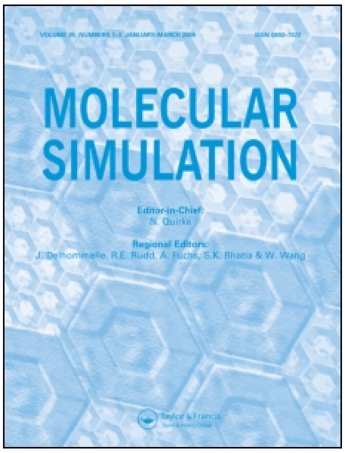

Molecular Simulation
Publication details, including instructions for authors and subscription information:
http://www.informaworld.com/smpp/title~content=t713644482

Molecular Dynamics Simulations of Proteins in Water Without the Truncation of Long-range Coulomb Interactions

Minoru Saito[a]
[a] Protein Engineering Research Institute, Suita, Osaka, Japan

To cite this Article Saito, Minoru(1992) 'Molecular Dynamics Simulations of Proteins in Water Without the Truncation of Long-range Coulomb Interactions', Molecular Simulation, 8: 6, 321 — 333

To link to this Article: DOI: 10.1080/08927029208022487

URL: http://dx.doi.org/10.1080/08927029208022487

Molecular Simulation, 1992, Vol. 8, pp. 321–333
Reprints available directly from the publisher
Photocopying permitted by license only

MOLECULAR DYNAMICS SIMULATIONS OF PROTEINS IN WATER WITHOUT THE TRUNCATION OF LONG-RANGE COULOMB INTERACTIONS

MINORU SAITO

Protein Engineering Research Institute, 6-2-3 Furuedai, Suita, Osaka, 565 Japan

(*Received May 1991, accepted September 1991*)

A new program package (COSMOS90) for molecular dynamics simulations was developed to simulate large molecular systems consisting of more than tens of thousands of atoms without the truncation of long-range coulomb interactions. This program package was based on a new approximation scheme (PPPC) for calculating efficiently the coulomb interactions without sacrificing accuracy. In this approximation scheme, the group of charges at a long distance from each atom was represented by a total charge and total dipole moment of the group. In order to assess the accuracy of PPPC and the ability of COSMOS90, molecular dynamics simulations were carried out for a large system consisting of 16108 atoms (human lysozyme in water) for 50 ps using this program package. The coulomb energy per solute atom was calculated with only five percent of the error found in the 10 Å cut-off approximation (about 0.9 kcal/mol versus 18 kcal/mol, respectively). The molecular dynamics simulations using COSMOS90 require no more CPU time than the simulations based on the 10 Å cut-off approximation of the conventional programs for macromolecular simulations.

KEY WORDS: Molecular dynamics, protein in water, truncation of coulomb interaction, COSMOS90, PPPC

1 INTRODUCTION

Molecular dynamics simulation is an important and powerful tool for the theoretical study in proteins. The first molecular dynamics simulation of proteins was carried out by McCammon *et al.* in 1977 [1]. Their simulation of the thermal motion of bovine pancreatic trypsin inhibitor (BPTI) consisting of 58 amino acid residues for 8.8 ps was in vacuum (absence of solvent molecules) because of the limited power of computers at that time. Molecular dynamics simulations for proteins in vacuum have serious shortcomings. For example, the deviation of protein structure from the known X-ray structure is unacceptably large [2]. In the early 1980s, with the growth in the power of computers, proteins partially solvated by a few hundred water molecules could be simulated [3]. Recently, super computers have made it possible to simulate systems such as proteins in water which are much closer to physiological environments [2,4].

In these calculations the evaluation of coulomb interactions was restricted to atom pairs occurring within a specified cut-off distance (typically 8–15 Å) [5] in order to lower the computational burden. However, the importance of coulomb interactions in proteins is well recognized. One reason is the important role played by coulomb interaction in the thermal stability and catalytic reaction of proteins.

The dipole axis moments of peptide units (about 3.5 D) are aligned nearly parallel to the helix axis in an α-helix. Thus, the α-helix has a large dipole moment of between 10 and 40 Debye [6]. Nicholson *et al.* found that the interaction between the α-helix dipole and the charged amino acid located at the edge of the helix stabilizes a protein [7]. Hol suggested that the electric field generated by the helix dipole facilitates enzymatic reactions [8]. Eden *et al.* observed that cytochrome *c*, one of the hemoproteins well known as an electron carrier, has different compressibility between the two oxidation states of heme (ferrous and ferric states) [9].

The detailed treatment of long-range coulomb interactions is probably necessary to obtain really accurate results in applications to various phenomena (e.g., thermal stability of proteins and charge transfer reaction in proteins, etc.), because the electrostatic screening for coulomb interactions is not expected in proteins which have a low dielectric constant in the range of 2–4 [10].

The calculation of coulomb interactions without truncation (i.e., the direct summation with the infinite cut-off range) requires a computational load proportional to N^2 for a system of N atoms, because all $\frac{1}{2}N(N - 1)$ forces between all pairs of N atoms are evaluated. Even a small protein surrounded by bulk water molecules constitutes a very large molecular system with tens of thousands of atoms ($N \sim 10^4$). Therefore, it is impossible to carry out molecular dynamics simulations for the realistic systems (proteins in water) without a cut-off, even if super computers are used. Since all of the conventional programs for macromolecular simulations are based on the cut-off approximations, we can not accomplish molecular dynamics simulations for the realistic systems without the truncation of coulomb interactions.

This paper presents a new approximation scheme, the PPPC (Particle–Particle and Particle–Cell) method, for calculating efficiently the long-range coulomb interactions of large molecular systems consisting of tens of thousands of atoms. This approximation scheme is incorporated into a new program package for molecular dynamics simulation, COSMOS90 (Computer Simulation program for Molecular Systems in the 1990s), that simulates large molecular systems while explicitly taking into account long-range coulomb interactions without truncation. In the PPPC method, a large number of long-range coulomb interactions are efficiently handled by grouping together larger groups of particles at larger distances. Coulomb potential from the group of charges is then approximated by that from the total charge and total dipole moment of the charge group. Thus, we can concentrate computational effort where most needed without sacrificing accuracy.

We chose human lysozyme to test the reliability of PPPC and ability of COSMOS90 because its precise structure was recently obtained by a high-resolution X-ray study in our institute [11]. Molecular dynamics simulations were carried out for human lysozyme in water for 50 ps by using COSMOS90. First, in order to test the reliability of the PPPC method, the coulomb energy of each atom evaluated by the PPPC method was compared with that attained by the most accurate method, i.e., the direct summation with the infinite cut-off range (hereafter referred to as the accurate method). Central processing unit (CPU) time required for one step of the simulation was compared between COSMOS90 and one of the conventional programs, PRESTO [17]. Deviation of the structure of human lysozyme from the known X-ray structure was then investigated in both vacuum and in water.

2 CALCULATION METHODS

Molecular dynamics simulations were carried out for human lysozyme in water (shown in Figure 1) by using COSMOS90. For preparation of human lysozyme in water, the high-resolution X-ray structure of human lysozyme with crystal waters [11] was used as the initial structure. The human lysozyme and 91 crystal waters were immersed in a bulk water sphere with the radius $r_0 = 34$ Å. The sphere radius of 34 Å was such that the minimum distance of any protein atom from the wall of the sphere was about 8 Å. In this process, the bulk water molecules colliding with the solute atoms and crystal waters were removed. The boundary force $k(r - r_0)(r \geqslant r_0$, $k = 100\,\mathrm{kcal/mol\,Å^2})$ acted on the water molecules to prevent them from vaporizing. The number of atoms in human lysozyme was 2041, in which 16 lone pair atoms were included. The number of water molecules was 4689. Thus, the total number of atoms in the present system was 16108.

The force field used in COSMOS90 has the following form,

$$
\begin{aligned}
E_{\text{total}} &= \sum_{\text{bonds}} K_d(d - d_0)^2 && (1)\\
&+ \sum_{\text{angles}} K_\theta(\theta - \theta_0)^2 && (2)\\
&+ \sum_{\text{dihedrals}} \frac{V_n}{2}[1 + \cos(n\phi - \gamma)] && (3)\\
&+ \sum_{i<j} \left[\frac{A_{ij}}{r_{ij}^{12}} - \frac{B_{ij}}{r_{ij}^{6}}\right] && (4)\\
&+ \frac{1}{2}\sum_{i,J} q_i \left[\frac{q_J}{r_{iJ}} + \frac{\boldsymbol{\mu}_J \cdot \mathbf{r}_{iJ}}{\mathbf{r}_{iJ}^{3}}\right] && (5)\\
&+ \sum_{\text{H-bonds}} \left[\frac{C_{ij}}{r_{ij}^{12}} - \frac{D_{ij}}{r_{ij}^{10}}\right] && (6)
\end{aligned}
$$

The terms of this equation represent (1) covalent bond stretching, (2) bond angle bending, (3) sinusoidal proper (and improper) dihedral torsion, (4) non-bonded van der Waals interactions, (5) electrostatic interactions of charge-charge and charge-dipole, and (6) correction for hydrogen bond, respectively. The all atom force field parameters of Weiner *et al.* [12] were used for the human lysozyme with the exception of the water molecules, which were described by the SPC model [13]. In this model, the following parameters are zero: (i) the 10-12 potential parameters of oxygen and hydrogen atoms. (ii) the van der Waals parameters of hydrogen atoms. The set of all bonds, angles, and torsions was prepared using AMBER [14].

As mentioned above, the PPPC method efficiently evaluates coulomb interactions without resort to truncation. This method is based on a representation of charge distribution as cubic cells with a point charge and point dipole by subdividing the system into tree-structured hierarchical cubic cells. An algorithm of such a hierarchical subdivision was proposed by Barnes and Hut [15] and vectorized by Makino [16] to simulate the dynamics of gravitational many-body systems (e.g., stellar systems). Their algorithm is as follows. First, an empty cubic cell large enough to contain the entire system (a root cell) is prepared. Second, the root cell is divided into cubic subcells. Third, each of subcells is recursively divided into subcells whenever more

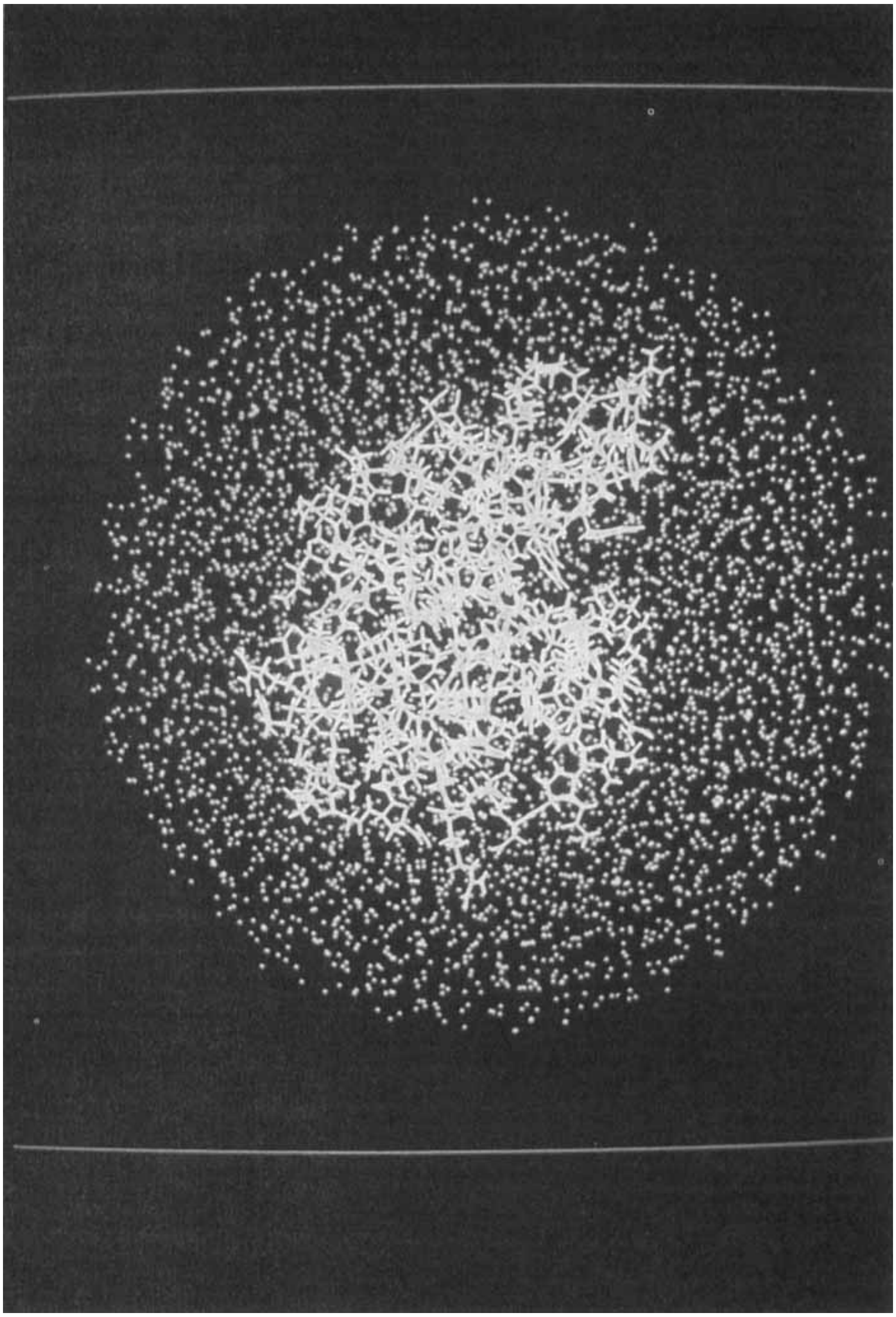

Figure 1 Human lysozyme in the 34 Å water sphere. Points distributed are water molecules. A cleft at right above the lysozyme is the active site cleft.

than one particle is found to occupy the same cell. Last, particles are grouped together into cubic cells, which are grouped together into larger cells. The particles correspond to the leaves of a tree, and the subcells to its branches.

In order to apply the algorithm to the present molecular system, each water molecule as well as each solute atom was treated as a single particle. Each subdivided cell, designated J, had a total charge q_J and total dipole moment $\boldsymbol{\mu}_J$ for its charges. The charge q_J and dipole $\boldsymbol{\mu}_J$ are located at the center of cell J. Then, a particle feels the forces from neighboring particles and from larger cells at larger distances. Sizes of cells interacting with each particle were determined by the following criterion;

$$l \leqslant \Theta(d)d, \tag{7}$$

$$\Theta(d) = \sqrt{\frac{\alpha d^2}{1 + \beta d^2}},$$

where l is the length of the cell and d the distance from the cell's center to the current particle, and $\alpha(=0.25 \times 10^{-3})$ and $\beta(=1.0 \times 10^{-3})$ are accuracy parameters. Since the right hand side of this inequality has a small value in the neighbor region of each particle compared with distances between particles, the neighbor region was divided down to the smallest cells containing a single particle. Each small cell containing a single particle was treated as the particle itself. The coulomb interactions in the neighbor region were then evaluated rigorously without any approximations.

Furthermore, in order to calculate efficiently coulomb interactions without sacrificing accuracy, computational efforts were concentrated on the neighbor region of each particle as follows. The space around each particle was divided into two regions (inner and outer regions) at the specific range $r_c = 6.0$ Å. The inner region contained only particles and no cells under the criterion given by inequality (7). On the other hand, the outer region contained a small number of particles and a large number of cells. Since the van der Waals force rapidly decreases as a function of distance as compared with the coulomb force, the van der Waals force was truncated at the boundary of the inner region. A water molecule in the outer region was represented by its point dipole.

The coulomb and van der Waals forces from the inner region were evaluated at every time step. Meanwhile, the coulomb forces from the outer regions were evaluated less frequently (every 10 time steps) and were kept fixed between updates. A hierarchical tree-structure for cells was reconstructed every 50 time steps. Errors caused by these approximations were as small as the thermal energy $kT \approx 0.6$ kcal/mol at $T = 300$ K, as shown in the next section.

The number of nonbonded forces acting on an atom is given by $C(\alpha, \beta) \log N$, where N is the number of atoms and the coefficient $C(\alpha, \beta)$ depends on the accuracy of subdivision. The CPU time required to simulate an N particle system by the PPPC method slowly grows as the function $[C(\alpha, \beta)/n] N \log N$ of N, where n is the number of time steps at which forces from the outer region are updated. Further details concerning the PPPC method appear in Appendices A, B, and C.

Translational motions of solute atoms and water molecules were described by the Newton equations for the Cartesian coordinates. Rotational motions of water molecules were described by the Euler equations for the quaternion parameters. Those equations were integrated using the leap-frog algorithm [19] for solute atoms and Fincham's algorithm [20] for water molecules, respectively. The constant temperature simulations were performed using slight modifications of the integration schemes [21]. Details for the integration schemes will be described in another paper.

Initially, the solute atoms (and water molecules) had the velocities (and angular velocities) with the absolute value in the thermal equilibrium at 300 K and random orientations. Molecular dynamics simulations were carried out for 50 ps at 300 K with the time step of 0.5 fs.

Calculations were carried out on the FACOM VP400E computer at the Protein Engineering Research Institute. The present calculations took about 50 hours of CPU time on the FACOM VP400E.

3 RESULTS AND DISCUSSION

Since the coulomb energy U_i of a particle i (solute atom or water molecule) is the most sensitive to approximations, error in the coulomb energy is regarded as a measure of the reliability of approximations. In the present paper, the error in coulomb energy was defined by the standard deviation ΔU.

$$\Delta U = \sqrt{\overline{(U_i' - U_i^0)^2}} \tag{8}$$

$$U_i^0 = \sum_{j \neq i} \frac{q_i q_j}{r_{ij}}$$

where U_i' is the approximate value and U_i^0 is the accurate value obtained by direct summation with the infinite cut-off range.

The coulomb energy of each solute atom and each water molecule was calculated for the configuration at 50 ps by three different methods (10 Å cut-off, PPPC, and accurate methods), where the cut-off approximation was implemented by the residue-based cut-off scheme commonly used in conventional programs. Coulomb energy values calculated by the two approximations (10 Å cut-off and PPPC) were compared with that by the accurate method in Figures 2a and 2b. In both Figures, the horizontal axis denotes the energy value U_i^0 calculated by the accurate method. The vertical axes denote the energy values U_i' calculated by the 10 Å cut-off approximation in Figure 2a and by the PPPC approximation in Figure 2b, respectively. Thus, the deviation of a cross sign from the diagonal line means the difference in the coulomb energy $(U_i' - U_i^0)$ for the respective particle i due to the 10 Å cut-off approximation (Figure 2a) and due to the PPPC approximation (Figure 2b). It was found that the error in the coulomb energy ΔU was substantial for the 10 Å cut-off approximation but very small for the PPPC approximation, as shown in Figures 2a and 2b.

The error in the coulomb energy due to the 10 Å cut-off approximation was significantly large (18 kcal/mol) for each solute atom and small (0.8 kcal/mol) for each water molecule, respectively. On the other hand, the error in the coulomb energy due to the PPPC approximation was noticeably small for each solute atom as well as each water molecule (0.9 and 0.2 kcal/mol, respectively). These results recommend the PPPC method as an excellent approximation for calculating coulomb interactions.

Since the water molecule is neutral and has a small coulomb energy, the error in coulomb energy is also small and insensitive to the approximation. The cross signs localized at the center in Figure 2a denote the coulomb energies of water molecules. On the other hand, solute atoms have net charges. Therefore, the error in the coulomb energy of each solute atom is large and sensitive to the approximation, as shown in Figure 2a. The present system is positively charged (+8.0e) due to charged amino acides. In order to check whether nonneutrality of the system changes the present

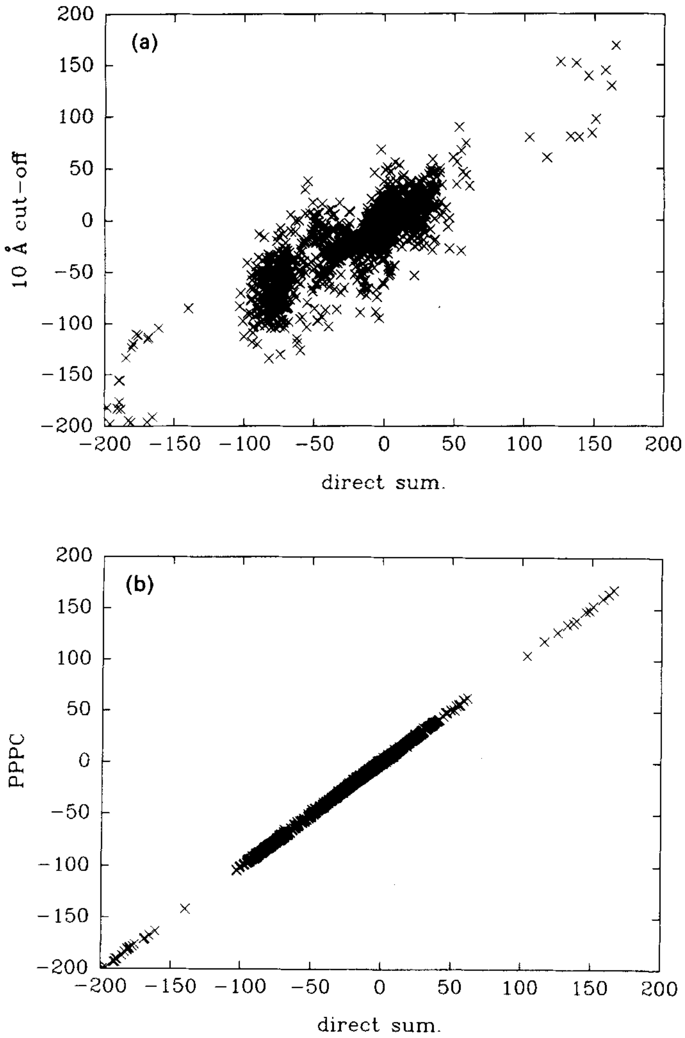

Figure 2 Coulomb energy (kcal/mol) of each atom of human lysozyme and each water molecule. In both Figures, the horizontal axes show the energy value calculated by the accurate method. The vertical axes show the energy values calculated by (a) the 10 Å cut-off approximation and (b) the PPPC approximation, respectively.

conclusion, a preliminary calculation was carried out for human lysozyme with counter ions which neutralize the system. It was found that the counter ions increase the error in coulomb energy significantly for the 10 Å cut-off method (about 29 kcal/mol) and negligibly for the PPPC method (about 1.2 kcal/mol), respectively. Details will be described in the next paper concerning more precise simulations of human lysozyme.

The CPU time required for one step of the simulation using the PPPC method was reduced as follows: (1) the hierarchical tree-structured cells were reconstructed every

Table 1 CPU time (sec.)[a] required for evaluating one step of molecular dynamics simulation.

Calculation method	*Human lysozyme*	
	In vacuum[b]	*In water*[c]
COSMOS90		
PPPC	0.32	1.8
PRESTO[d]		
Accurate method[e]	0.85	73.0
10 A cut-off[f]	0.38	1.6

[a] Measured using a FACOM VP400E
[b] Number of atoms = 2041.
[c] Number of atoms = 16108.
[d] Conventional program based on the cut-off method.
[e] Direct summation with the infinite cut-off range.
[f] Interaction table updated for every 20 steps.

50 time steps; (2) the coulomb force from the outer region was updated every 10 time steps; and (3) the van der Waals force was truncated at the boundary of the inner region. These approximations caused errors in the coulomb (van der Waals) energy per particle of 0.2 kcal/mol for (1), 0.5 kcal/mol for (2), and 0.3 kcal/mol for (3), respectively. These errors are small compared with the above mentioned error of the PPPC method (0.9 kcal/mol).

The performance speed of COSMOS90 was compared with that of the conventional program PRESTO [17] in Table 1. The performance speed was defined as the CPU time consumed for evaluating one step of the simulation. PRESTO is based on the cut-off approximation and has almost the same performance speed as the fully vectorized program ARGOS [18]. The CPU times required by COSMOS90 and by PRESTO for evaluating one step of a molecular dynamics simulation for the human lysozyme in vacuum and in water is shown in Table 1. For the human lysozyme in vacuum (2041 atoms), every calculation method requires short CPU time (less than 1 sec) to perform one step. Therefore, the molecular dynamics simulation of the human lysozyme in vacuum can easily be carried out for a few hundred picoseconds by using the accurate method with conventional programs.

For the human lysozyme in water (16108 atoms), however, the accurate method with PRESTO requires an extremely long CPU time (73 sec). This CPU time requirement makes impossible a molecular dynamics simulation of this large system for a few hundred pico-seconds, even with the aid of a super computer. The 10 Å cut-off approximation of PRESTO requires reasonable CPU tine (1.6 sec) but demands a sacrifice of accuracy. On the other hand, COSMOS90 requires almost the same CPU time (1.8 sec) as the cut-off approximation of PRESTO, even though the former explicitly takes into account the long-range coulomb interactions. Therefore, COSMOS90 has made it possible to carry out molecular dynamics simulations for large systems without sacrificing accuracy in the calculation of coulomb interactions.

The human lysozyme in water (16108 atoms) consists of about 8 times more atoms than the human lysozyme in vacuum (2041 atoms). The CPU time required to calculate interactions grows rapidly as N^2 for the accurate method. However, the CPU time required by the PPPC method grows slowly as $[C(\alpha, \beta)/n]N \log N$. Therefore, larger systems consisting of more than tens of thousands of atoms may be handled by COSMOS90, as the power of computers grows.

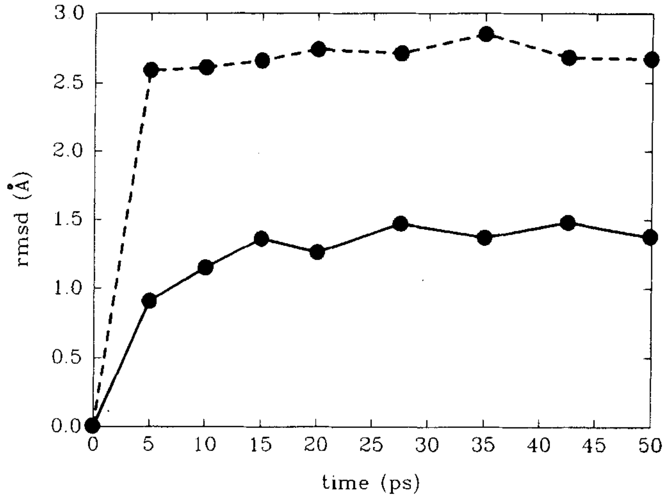

Figure 3 Time dependence of the root mean square deviation (rmsd) of all main chain atoms (C_α, C, and N) from X-ray structure. The dotted line denotes the rmsd of the human lysozyme in vacuum, and the solid line the human lysozyme in water.

In order to check whether the PPPC method was well incorporated into COSMOS90, molecular dynamics simulations were carried out for the human lysozyme in vacuum and in water for 50 ps by using COSMOS90. Deviation of the structure of human lysozyme from the initial X-ray structure was investigated to determine the reliability of calculations. The root mean square deviation (rmsd) of all main chain atoms (C_α, C, and N), shown in Figure 3, was calculated for the human lysozyme in water and in vacuum at 5, 10, 15, 20, 27.5, 35, 42.5, and 50 ps.

The rmsd in vacuum (dotted line in Figure 3) increased rapidly and reached a stable value of about 2.7 Å. On the other hand, the rmsd in water (solid line in Figure 3) increased gradually and reached a stable value of around 1.5 Å at about 20 ps. These results mean that the structure of human lysozyme deviates very quickly from the starting X-ray structure in vacuum, and remains closer to the initial X-ray structure in water. Significant structural changes were recognized on the active site cleft using a computer graphic display. The cleft was closed in vacuum. In contrast, the cleft was kept open in water and had almost the same shape as the cleft of the initial X-ray structure.

In general, a protein in crystal is surrounded by several hundred water molecules and has an environment similar to that in solution. This fact supports the present result; the human lysozyme has a structure closer to the X-ray structure in water than in vacuum. In order to discern fine structural changes, more precise simulations which take into account counter ions, abnormal pK_a of Glu 35, and adequate equilibration are in progress.

References

[1] J.A. McCammon, B.R. Gelin and M. Karplus, "Dynamics of folded proteins", *Nature*, **267**, 585 (1977).

[2] M. Levitt and R. Sharon, "Accurate simulation of protein dynamics in solution", *Proc. Natl. Acad. Sci. USA*, **85**, 7557 (1988).

[3] A.T. Brünger, C.L. Brooks III and M. Karplus, "Active site dynamics of ribonuclease", *Proc. Natl. Acad. Sci. USA*, **82**, 8458 (1985).

[4] J. de Vlieg, H.J.C. Berendsen and W.F. van Gunsteren, "An NMR-based molecular dynamics simulation of the interaction of the *lac* repressor headpiece and its operator in aqueous solution", *PROTEINS: Structure, Function, and Genetrics*, **6**, 104 (1989).

[5] R.J. Loncharich and B.R. Brooks, "The effects of truncating long-range forces on protein dynamics", *PROTEINS: Structure, Function, and Genetics*, **6**, 32 (1989).

[6] W.G.J. Hol, P.T. van Duijnen and H.J.C. Berendsen, "The α-helix dipole and the properties of proteins", *Nature*, **273**, 443 (1978).

[7] H. Nicholson, W.J. Becktel and B.W. Matthews, "Enhanced protein thermostability from designed mutations that interact with α-helix dipoles", *Nature*, **336**, 651 (1988).

[8] W.G.J. Hol, "The role of the α-helix dipole in protein function and structure", *Prog. Biophys. Molec. Biol.*, **45**, 149 (1985).

[9] D. Eden, J.B. Matthew, J.J. Rosa and F.M. Richards, "Increase in apparent compressibility of cytochrome *c* upon oxidation", *Proc. Natl. Acad. Sci. USA*, **79**, 815 (1982).

[10] S. Takashima and H.P. Schwan, "Dielectric dispersion of crystalline powders of amino acids, peptides, and proteins", *J. Phys. Chem.*, **69**, 4176 (1965).

[11] K. Inaka, Y. Taniyama, M. Kikuchi, K. Morikawa, and M. Matsushima, "The crystal structure of a mutant human lysozyme C77/95A with increased secretion efficiency in Yeast", *J. Biol. Chem.*, **266**, 12599 (1991).

[12] S.J. Weiner, P.A. Kollman, D.T. Nguyen and D.A. Case, *J. Comp. Chem.*, **7**, 230 (1986).

[13] H.J.C. Berendsen, J.P.M. Postma, W.F. von Gunsteren and J. Hermans, "Interaction models for water in relation to protein hydration", in *Intermolecular forces*, B. Pullman, ed., Reidel, Dordrecht, 1981, p. 331.

[14] U.C. Singh, P.K. Weiner, J.W. Caldwell and P.A. Kollman, AMBER(UCSF), version 3.0 (Department of Pharmaceutical Chemistry, University of California, San Francisco, 1986).

[15] J. Barnes and P. Hut, "A hierarchical $O(N \log N)$ force-calculation algorithm', *Nature*, **324**, 446 (1986).

[16] J. Makino, "Vectorization of a treecode", *J. Comput. Phys.*, **87**, 148 (1990).

[17] K. Morikami, T. Nakai, A. Kidera, M. Saito and H. Nakamura, "PRESTO: A vectorized molecular mechanics program", *Computers Chem.*, in press.

[18] T.P. Straatsma and J.A. McCammon, "ARGOS, a vectorized general molecular dynamics program", *J. Comp. Chem.*, **11**, 943 (1990).

[19] R.W. Hockney, "The potential calculation and some applications", *Methods Comput. Phys.*, **9**, 136 (1970).

[20] D. Fincham, "An algorithm for the rotational motion of rigid molecules", *Daresbury Laboratory Information Quarterly for MD and MC simulations*, **2**, 6 (1981).

[21] D. Brown and J.H.R. Clarke, "A comparison of constant energy, constant temperature and constant pressure ensembles in molecular dynamics simulations of atomic liquids", *Molec. Phys.*, **51**, 1243 (1984).

[22] L. Greengard and V. Rokhlin, "A fast algorithm for particle simulations", *J. Comput. Phys.*, **73**, 325 (1987).

[23] H. Grubmüller, H. Heller, A. Windemuth and K. Schulten, "Generalized verlet algorithm for efficient molecular dynamics simulations with long-range interactions", *Molecular Simulation*, **6**, 121 (1991).

APPENDICES

A. Location of the monopole and dipole in a cell.

For gravitational systems [15], the total mass of a cell is located at the center-of-mass position $\mathbf{R}_G$

$$\mathbf{R}_G = \frac{\sum_{i=1}^{N} m_i \mathbf{r}_i}{\sum_{i=1}^{N} m_i}, \tag{A1}$$

where N is the number of particles in a cell, m_i and $\mathbf{r}_i$ are the mass and position vector of a particle i, respectively. For coulomb systems, a center-of-charge position $\mathbf{R}_C$ also can be defined for a cell by a similar equation,

$$\mathbf{R}_C = \frac{\sum_{i=1}^{N} q_i \mathbf{r}_i}{\sum_{i=1}^{N} q_i}, \tag{A2}$$

where q_i is charge of a particle i. However, the center-of-charge is not useful because of its ambiguous meaning, as shown below.

Let us consider the following case. $N - 1$ particles have a total charge of zero and total dipole moment $\boldsymbol{\mu} \neq 0$,

$$\sum_{i=1}^{N-1} q_i = 0 \tag{A3}$$

$$\boldsymbol{\mu} = \sum_{i=1}^{N-1} q_i \mathbf{r}_i \tag{A4}$$

Using Equations (A3) and (A4), $\mathbf{R}_C$ is written as follows,

$$\mathbf{R}_C = \frac{\sum_{i=1}^{N-1} q_i \mathbf{r}_i + q_N \mathbf{r}_N}{\sum_{i=1}^{N-1} q_i + q_N} = \frac{\boldsymbol{\mu}}{q_N} + \mathbf{r}_N. \tag{A5}$$

If the N-th particle has the large charge $q_N \gg 1$, the center-of-charge $\mathbf{R}_C$ is nearly equal to the position of N-th particle $\mathbf{r}_N$, as expected from physical intuition. However, if the N-th particle has the small charge $q_N \ll 1$, the first term $\boldsymbol{\mu}/q_N$ becomes extremely large compared with $\mathbf{r}_N$. Thus, $\mathbf{R}_C$ is significantly separated from the geometrical center of a cell and loses its physical meaning. Therefore, we have abandoned the idea of center-of-charge for coulomb systems, but have adopted the geometrical center of a cell together with a multipole expansion around it [22]. Multipole expansion around the cell center was truncated at the dipole term in the present study; nevertheless it has adequate accuracy, as shown in the previous section. It is straightforward to take into account higher order terms.

B. Physical meaning of the criterion $l \leqslant \Theta(d)d$.

For gravitational systems, sizes of cells interacting with each particle were determined by

$$l \leqslant \Theta_0 d, \tag{A6}$$

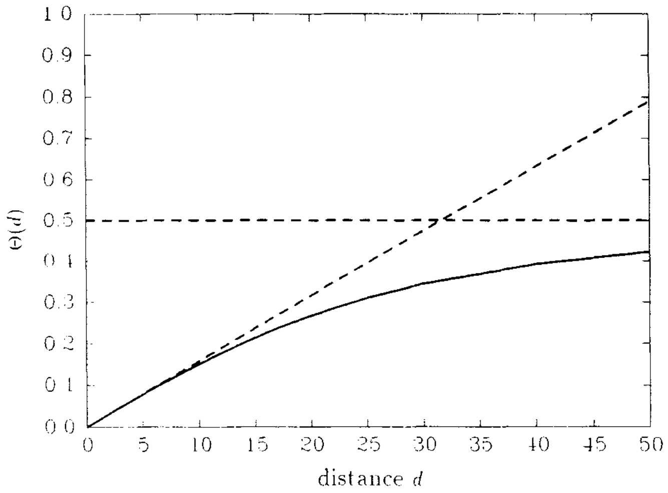

Figure 4 $\Theta(d)$ as a function of the distance d between a particle and cell (denoted by the solid curve), and its asymptotic forms $\Theta(d \to \infty)$ and $\Theta(d \to 0)$ (denoted by the broken lines).

where Θ_0 is constant and represents the opening angle for a cell [16]. In general, a smaller opening angle makes the error smaller but demands more CPU time. In coulomb systems, a certain shielding effect is expected for coulomb forces from charges of distant particles. In order to utilize this advantage, the opening angle was continuously varied from a small value for the neighbor region to a large value for the distant region by using the criterion given by inequality (7), i.e.,

$$l \leqslant \Theta(d)d,$$

$$\Theta(d) = \sqrt{\frac{\alpha d^2}{1 + \beta d^2}}. \tag{A7}$$

$\Theta(d)$ is a function of the particle-cell distance d and has the following asymptotic forms:

$$\Theta(d \to \infty) = \sqrt{\frac{\alpha}{\beta}} \tag{A8}$$

$$\Theta(d \to 0) = \sqrt{\alpha} d. \tag{A9}$$

$\Theta(d)$, $\Theta(d \to \infty)$, and $\Theta(d \to 0)$ are shown for $\alpha = 0.25 \times 10^{-3}$ and $\beta = 1.0 \times 10^{-3}$ in Figure 4 by the solid curve and broken lines, respectively. We can control the opening angle of neighbor and distant regions by choosing appropriate values of α and β. Inequality (A7) is equivalent with

$$l^2 \leqslant \frac{\alpha d^2}{1 + \beta d^2} d^2. \tag{A10}$$

This criterion is easily evaluated from positions of a particle and cell without using the FORTRAN intrinsic function, SQRT, and thus is advantageous for saving CPU time.

Table 2 CPU time and acceleration rate of four subprocesses of the PPPC method.

Subprocess[a]	*CPU time (sec.)*[b]		*Acceleration rate*[b]
	Vector	*Scalar*	
A	7.5	130	17
B	0.7	31	44
C	3.5	156	45
D	1.1	21	19

[a] Each subprocess is defined in the text.
[b] Vectorized and measured using a FACOM VP400E.

C. Process for computing coulomb interactions by the PPPC method.

In the PPPC method, coulomb interactions are computed according to the following subprocesses A, B, C, and D, which correspond to subprograms of COSMOS90. CPU time consumed by these subprocesses is shown in Table 2, together with the acceleration rate by vectorization.

Subprocess A (updated every 50 steps)
A-1 Subdividing a system into hierarchical cubic cells.
A-2 Making an interaction table for particle-particle and particle-cell.
Subprocess B (updated every 10 steps)
B-1 Dividing a system into the inner and outer regions.
B-2 Calculating charge and dipole moment of each cell.
Subprocess C (updated every 10 steps)
C-1 Calculating forces from particles and cells in the outer region.
Subprocess D (updated every step)
D-1 Calculating forces from particles in the inner region.

For the subprocesses A, B, and C, appropriate periods of update were chosen to decrease computational labor. Error caused by the every-50-steps update for subprocess A is small (about 0.2 kcal/mol) compared with kT, because solute atoms can not diffuse and water molecules negligibly diffuse for the update period ($50 \times 0.5 = 25$ fs). On the other hand, a water molecule rotates and changes its dipole moment quickly compared with the center-of-mass motion. Therefore, subprocesses B and C should be updated more frequently (every 10 steps) than the subprocess A to suppress the error below kT.

The basic idea described above (i.e., updating forces from the outer region every 10 steps) is based on the fact that Coulomb forces from distant particles are insensitive in the time development of those particles. This feature is also utilized by the distance-class method [23], in which a system is divided into several shells according to the distance from a current particle, and the forces from inner shells are computed more often than those from outer shells. Further, coulomb forces from distant particles have another feature: those forces are insensitive in the precise position of distant particles. This feature is not utilized in the distance-class method, because the interactions are calculated to update the forces for all pairs of particles without grouping distant particles. On the other hand, the PPPC method utilizes this feature as well as the previous feature by grouping distant particles as a cell and calculating forces from a monopole and dipole of the cell.