Partial Rigid-Body Dynamics in NPT, NPAT and NPγT Ensembles for Proteins and Membranes

MITSUNORI IKEGUCHI
Graduate School of Integrated Science, Yokohama City University, 1-7-29, Suehirocho, Tsurumi-ku, Yokohama 230-0045, Japan

Received 15 July 2003; Accepted 29 September 2003

Abstract: A partial rigid-body method of molecular dynamics simulations for proteins and membranes is presented. In this method, the symplectic integrator for rigid bodies is combined with the equations of motion for the NPT ensemble. The standard NPT ensemble is extended to the membrane-specific ensembles, the NPAT (constant normal pressure and lateral surface area of membranes and constant temperature) and NPγT (constant normal pressure and lateral surface tension of membranes and constant temperature) ensembles. By more than 30-ns simulations of aqueous proteins and hydrated lipid bilayers, the results of the partial rigid-body method demonstrated excellent conservation of total energy and consistent behavior with the traditional constraint method in terms of structural distribution and fluctuation of proteins and lipids. The efficient implementation of the partial rigid-body method in parallel computation is presented, which is shown to work well in large-scale molecular dynamics simulations.

 J Comput Chem 25: 529–541, 2004

Key words: molecular dynamics simulation; protein; membrane; NPT ensemble; NPAT ensemble; NPγT ensemble; rigid body; symplectic integrator; MARBLE

Introduction

Molecular dynamics simulations have become an indispensable tool to study the structure, dynamics, and functions of biomolecules such as proteins, nucleic acids, and lipid bilayers.[1–3] The increase of computational power and the improvement of algorithms enable us to apply molecular dynamics simulations to larger molecular systems in size in a longer time scale. However, to study biologically important molecular functions, spatial and temporal scales of current molecular dynamics simulations is not enough yet. Thus, more efforts on the improvement of the methodology of molecular dynamics simulations are required.

One of the important innovations in molecular dynamics algorithms is the improvement of algorithms for integrating equations of motion, the so-called "integrators." When the integrators are symplectic, reversible, and/or preserve phase space volume, stable simulations in a long time scale are possible. Recently, the simulation time scale has rapidly been increasing, and several tens of nanosecond simulations with explicit solvents are not rare. In such long time scale simulations, the stability of the integrator is very important.

The fast degrees of freedom, such as the lengths of bonds involving hydrogen, limit the short time step of integrators. However, by applying holonomic constraints to such bonds, the fast degrees of freedom can be removed, and a longer time step can be utilized. Therefore, in molecular dynamics simulations of biomolecules, the holonomic constraints are often introduced by using, for example, the SHAKE/RATTLE method.[4] However, the SHAKE/RATTLE method requires iterations, and is neither symplectic, reversible, nor phase space volume preserving, because it has a finite tolerance. Another way to remove the fast degrees of freedom is rigid-body dynamics. By treating small molecules (e.g., water molecules) and small atomic groups as rigid bodies, the fast degrees of freedom can be removed. In rigid-body dynamics, the equations of rotational motion of rigid bodies must be solved in addition to the equations of translational motion. Several methods to integrate equations of rotational motion have been proposed so far,[4] and recently, two sophisticated integrators for rigid bodies were proposed: one is reversible and preserves phase space volume.[5] The other is not only reversible and phase space volume preserving but also symplectic.[6] These integrators were shown to provide very stable trajectories of simulations.

Algorithms that generate the isothermal or isothermal–isobaric ensemble are also important to obtain equilibrium states in biomolecular simulations. Several methods were developed to gener-

Correspondence to: M. Ikeguchi; e-mail: ike@tsurumi.yokohama-cu.ac.jp

Contract/grant sponsor: Grant-in-Aid for Scientific Research on Priority Areas ("Genome Information Science," Ministry of Education, Culture, Sports, Science and Technology of Japan

ate the ensemble previously.[4] Rigorous algorithms to generate the isothermal or isothermal–isobaric ensemble have also been proposed,[4,7–9] among which the equations of motion proposed by Martyna et al.[7] for the isothermal–isobaric ensemble satisfy both the pressure and kinetic virial theorems. A reversible integrator for the equations of motion was also proposed.[10]

To perform molecular dynamics simulations for membranes and membrane proteins, in addition to the standard NPT ensemble, specific ensembles have been utilized due to the anisotropic nature of membranes, that is, thermodynamic differences between the lateral and normal directions of membranes.[11] Therefore, as special ensembles for membranes, the NPAT ensemble, in which the normal pressure and the lateral surface area of the membrane, are kept constant, and the NPγT ensemble, in which the normal pressure and the lateral surface tension are kept constant, were proposed.[11] However, the previous equations of motion for these ensembles did not consider the pressure and kinetic virial theorems.

This article proposes another set of equations of motion and an integrator of rigid-body dynamics in the NPT, NPAT, and NPγT ensembles for biomolecules such as proteins and membranes. In this method, small molecules and atomic groups involving hydrogen (—CH_x, —NH_x, —OH) are treated as rigid bodies, and the symplectic rigid-body integrator is combined with the equations of motion that rigorously generate the NPT ensemble. Bonds between heavy atoms are treated as flexible bonds. The SHAKE/RATTLE method, which is usually used in current biomolecular simulations, constrains only the length of bonds involving hydrogen. Because the partial rigid-body method proposed in this article treats small atomic groups as rigid bodies, the bond angles of H—X—H (H: hydrogen, X: a heavy atom) are also kept constant, as well as the bond lengths. Thus, the molecular model used in the partial rigid-body method is slightly different from that of the usually used SHAKE/RATTLE method. If constraints are applied to the lengths of virtual bonds between hydrogens of H—X—H, the SHAKE/RATTLE method can constrain the bond angles of H—X—H. The bond angles of water molecules are often fixed in such a way. However, the bond angles of H—X—H in larger molecules, such as proteins and lipid molecules, are not usually constrained. Therefore, an examination of the influence of the bond angles of H—X—H on the simulation results of proteins and lipid molecules is required.

Recently, Shinoda and Mikami reported a method that combines a reversible rigid-body integrator[5] with the NPT ensemble.[12] However, the rigid-body integrator of their method does not satisfy its symplectic nature, and it is limited to the NPT ensemble. Furthermore, they did not examine the influences of the constraint of H—X—H bond angles on the structural distribution and fluctuation of biomolecules such as proteins and lipids. The method proposed in this article combines the symplectic rigid-body integrator with the NPT, NPAT, and NPγT ensembles. More than 30 ns simulations of aqueous proteins and hydrated lipid bilayers were used to estimate the effect of the rigid-body method on the structural distribution and fluctuation of aqueous proteins and lipid bilayers.

To put the method to practical use, a combination with state-of-the-art molecular dynamics algorithms is crucial. The partial rigid-body method was implemented in the molecular dynamics program package, MARBLE. The MARBLE was built from scratch by the author of this article, which was mainly intended to perform molecular simulations of biomolecules. The MARBLE was written using the C language and used the Message Passing Interface (MPI) library for communications between processes. Because the only standard programming environment was used, the MARBLE can be installed in various computer platforms including PC clusters. By using the topology and parameter files of CHARMM[13,14] as input, a molecular data file, including the topology and force field data of target molecules, was generated from a Protein Data Bank (PDB) file. The AMBER[15] topology file is also available to generate the molecular data file. In the program, any part of the target molecules can be defined as a rigid body. (In this article, only simulations, in which small groups involving hydrogens (—CH_x, —NH_x, —OH, H_2O) were treated as rigid bodies, were shown.) The MARBLE incorporates several state-of-the-art molecular dynamics algorithms. As one of such algorithms, the MARBLE includes accurate and fast treatments of electrostatic interactions such as the particle mesh Ewald method[16] and the fast multipole method.[17] Another key to fast computation is parallel computation. In the MARBLE, the spatial-decomposition approach was adopted for parallelization. The spatial-decomposition approach involves the lowest communication cost among various parallelization algorithms of molecular dynamics simulations.[18–20] However, the spatial-decomposition approach suffers from the difficulty of load balancing among processors.[18–20] In the MARBLE, the dynamic load-balancing technique[21] was used to overcome the difficulty. In this article, it is demonstrated that the partial rigid-body method can be efficiently implemented in spatial-decomposition parallelization with dynamic load balancing.

Methods

To generate the NPT ensemble, the extended-system approach is used in this article. In the extended-system approach, additional dynamical variables are introduced to control the temperature and pressure of the system. The trajectories in the extended phase space were shown to generate the rigorous NPT equilibrium distribution.[7] In this section, the equations of motion proposed by Martyna et al.[7] for the NPT ensemble are extended to the equations of rigid-body motion. Next, the standard NPT ensemble is extended to the specific ensembles for membranes, that is, the NPAT (constant normal pressure, surface area and temperature) and NPγT (constant normal pressure, surface tension, and temperature) ensembles. Then, the numerical integrations of the equations of motion for the ensembles are described. The parallelization of the method and computational details are presented.

Equations of Rigid-Body Motion for the NPT Ensemble

The equations of atomic motion in the NPT ensemble proposed by Martyna et al.[7] are extended to the equations of rigid-body motion. Consider a system which consists of N_r rigid bodies and N_p particles that do not belong to rigid bodies ($N = N_r + N_p$). The pressure of the system is controlled by scaling the center of mass positions of rigid bodies and particles. The equations of rigid-body motion for the NPT ensemble are

$$\dot{\mathbf{r}}_i = \frac{\mathbf{p}_i}{m_i} + \frac{\overleftrightarrow{\mathbf{p}}_g}{W_g}\mathbf{r}_i, \qquad \dot{\mathbf{p}}_i = \mathbf{F}_i - \frac{\overleftrightarrow{\mathbf{p}}_g}{W_g}\mathbf{p}_i - \frac{1}{3N}\frac{\mathrm{Tr}[\overleftrightarrow{\mathbf{p}}_g]}{W_g}\mathbf{p}_i - \frac{p_{\eta_1}}{Q_1}\mathbf{p}_i, \qquad \dot{\mathbf{q}}_i = \frac{1}{2}\overleftrightarrow{\mathbf{S}}\boldsymbol{\omega}_i^{(4)}, \qquad \dot{\mathbf{L}}_i = \mathbf{T}_i - \boldsymbol{\omega}_i \times \mathbf{L}_i - \frac{p_{\eta_1}}{Q_1}\mathbf{L}_i, \qquad \dot{\overleftrightarrow{\mathbf{h}}} = \frac{\overleftrightarrow{\mathbf{p}}_g}{W_g}\overleftrightarrow{\mathbf{h}},$$

$$\dot{\eta}_k = \frac{p_{\eta_k}}{Q_k}, \quad k = 1, \ldots, M, \qquad \dot{p}_{\eta_k} = G_k - \frac{p_{\eta_{k+1}}}{Q_{k+1}}p_{\eta_k}, \quad k = 1, \ldots, M-1, \qquad \dot{p}_{\eta_k} = G_k, \quad k = M,$$

$$G_1 = \sum_{i=1}^{N}\frac{\mathbf{p}_i^2}{m_i} + \sum_{i=1}^{N_r}\boldsymbol{\omega}_i \cdot \mathbf{L}_i + \frac{\mathrm{Tr}[\overleftrightarrow{\mathbf{p}}_g'\overleftrightarrow{\mathbf{p}}_g]}{W_g} - (N_f + N_b)kT, \qquad G_k = \frac{p_{\eta_k}^2}{Q_k} - kT, \quad k = 2, \ldots, M, \tag{1}$$

and

$$\dot{\overleftrightarrow{\mathbf{p}}}_g = V(\overleftrightarrow{\mathbf{P}}_{\mathrm{int}} - P_{\mathrm{ext}}\overleftrightarrow{\mathbf{E}}) + \left[\frac{1}{3N}\sum_{i=1}^{N}\frac{\mathbf{p}_i^2}{m_i}\right]\overleftrightarrow{\mathbf{E}} - \frac{p_{\eta_1}}{Q_1}\overleftrightarrow{\mathbf{p}}_g, \tag{2}$$

where $\mathbf{r}_i$ and $\mathbf{p}_i$ are the center-of-mass position and the translational momentum, respectively, of the ith body ($i = 1, \ldots, N_r$) or the ith particle ($i = N_r + 1, \ldots, N$). m_i is the mass of the ith rigid body or the ith particle. $\mathbf{q}_i$ is the quaternion representing the orientation of the ith rigid body. $\mathbf{L}_i$ is the angular momentum in the body-fixed frame. $\mathbf{F}_i$ and $\mathbf{T}_i$ are the force and torque, respectively. $\overleftrightarrow{\mathbf{h}}$ is the matrix of cell parameters. The volume V is related to $\overleftrightarrow{\mathbf{h}}$ as $V = \det(\overleftrightarrow{\mathbf{h}})$. $\overleftrightarrow{\mathbf{p}}_g$ is the matrix of barostat momenta, and W_g is their associated mass parameter. η_k and p_{η_k} are the kth thermostat variable and its conjugate momentum, respectively. Q_k is its associated mass parameter and G_k is the force of the thermostat. M is the number of the Nóse–Hoover chain. N_f is the number of degrees of freedom in the system. N_b is the number of variables of the cell parameters, which is defined in each ensemble as described below. $\overleftrightarrow{\mathbf{P}}_{\mathrm{int}}$ is the instantaneous pressure tensor of the system, which is defined as,

$$(\overleftrightarrow{\mathbf{P}}_{\mathrm{int}})_{\alpha\beta} = \frac{1}{V}\left[\sum_{i=1}^{N}\frac{(\mathbf{p}_i)_\alpha(\mathbf{p}_i)_\beta}{m_i} + \sum_{i=1}^{N}(\mathbf{F}_i)_\alpha(\mathbf{r}_i)_\beta - \left(\frac{\partial U}{\partial \overleftrightarrow{\mathbf{h}}}\overleftrightarrow{\mathbf{h}}'\right)_{\alpha\beta}\right], \qquad \alpha = x, y, z, \quad \beta = x, y, z. \tag{3}$$

P_{ext} is the external pressure. $\overleftrightarrow{\mathbf{E}}$ is the identity matrix. $\boldsymbol{\omega}_i$ is the angular velocity in the body-fixed frame. $\boldsymbol{\omega}_i^{(4)}$ is the four-dimensional vector that consists of the angular velocity and an auxiliary term, and is defined as,

$$\boldsymbol{\omega}_i^{(4)} = (0, \omega_{xi}, \omega_{yi}, \omega_{zi}). \tag{4}$$

The angular momentum $\mathbf{L}_i$ is related to the angular velocity $\boldsymbol{\omega}_i$ in the following way:

$$\mathbf{L}_i = \overleftrightarrow{\mathbf{I}}_i\boldsymbol{\omega}_i, \tag{5}$$

where $\overleftrightarrow{\mathbf{I}}_i$ is the matrix of inertia momenta. The matrix $\overleftrightarrow{\mathbf{S}}$ is defined as,

$$\overleftrightarrow{\mathbf{S}} = \begin{pmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & -q_3 & q_2 \\ q_2 & q_3 & q_0 & -q_1 \\ q_3 & -q_2 & q_1 & q_0 \end{pmatrix}. \tag{6}$$

The equations of motion [eqs. (1), (2)] conserve the total energy of the extended system H' which is,

$$H' = U + K + P_{\mathrm{ext}}V + \frac{\mathrm{Tr}[\overleftrightarrow{\mathbf{p}}_g'\overleftrightarrow{\mathbf{p}}_g]}{2W_g} + kT\left[(N_f + N_b)\eta_1 + \sum_{k=2}^{M}\eta_k\right] + \sum_{k=1}^{M}\frac{p_{\eta_k}^2}{2Q_k}, \qquad K = \sum_{i=1}^{N}\frac{\mathbf{p}_i^2}{2m_i} + \sum_{i=1}^{N_r}\frac{\boldsymbol{\omega}_i \cdot \mathbf{L}_i}{2}, \tag{7}$$

where K is the kinetic energy of the system. Note that the total energy H' is not Hamiltonian. In the case of the fully flexible cell, N_b is 9 and the Jacobian J is

$$J = (\det \overleftrightarrow{\mathbf{h}})^{-2}\exp\left[(N_f + N_b)\eta_1 + \sum_{k=2}^{M}\eta_k\right]. \tag{8}$$

If there is an external force ($\sum_{i=0}^{N}\mathbf{F}_i \neq 0$), the total energy H' is the only conserved quantity and, thus, the partition function Δ takes the form,

$$\Delta \propto \int d\overleftrightarrow{\mathbf{h}} d^N\mathbf{r} d^N\mathbf{p} d^{N_r}\mathbf{q} d^{N_r}\mathbf{L} d\overleftrightarrow{\mathbf{p}}_g d^M\eta d^M p_\eta (\det \overleftrightarrow{\mathbf{h}})^{-2} \times \exp\left[(N_f + N_b)\eta_1 + \sum_{k=2}^{M}\eta_k\right]\eta(H' - E), \quad (9)$$

where E is a constant. The partition function Δ is integrated in terms of $\overleftrightarrow{\mathbf{p}}_g$, η and p_η, and $\overleftrightarrow{\mathbf{h}}$ is replaced by $\overleftrightarrow{\mathbf{h}}_0$ using $\overleftrightarrow{\mathbf{h}} = V^{1/3}\overleftrightarrow{\mathbf{h}}_0$. The partition function Δ thus becomes

$$\Delta \propto \int dV d\overleftrightarrow{\mathbf{h}}_0 d^N\mathbf{r} d^N\mathbf{p} d^{N_r}\mathbf{q} d^{N_r}\mathbf{L}\delta(\det \overleftrightarrow{\mathbf{h}}_0 - 1) \times \exp[-\beta(U + K + P_{\text{ext}}V)], \quad (10)$$

and it ensures that the equations of motion [eqs. (1), (2)] can generate the proper NPT ensemble. This derivation is an extension of that of the atomic systems presented by Martyna et al.[7]

However, in the case of the absence of an external force ($\Sigma_{i=0}^{N}\mathbf{F}_i = 0$), the following additional conservation law is also satisfied:

$$e^{\eta_1}V^{1/(3N)}\overleftrightarrow{\mathbf{h}}^t\mathbf{P} = \mathbf{C}, \quad (11)$$

where $\mathbf{P}$ is the translational momentum of the whole system and $\mathbf{C}$ is the constant vector. In the derivation of the conservation law [eq. (11)], the relation $\overleftrightarrow{\mathbf{p}}_g = \overleftrightarrow{\mathbf{p}}'_g$ was used. This relation comes from the elimination of the cell rotation.[10] Due to the conservation law, the partition function no longer satisfies eqs. (9) and (10). Thus, generally, the equations of motion do not generate the proper NPT ensembles except the following cases. If the momentum of the whole system is absent ($\mathbf{P} = 0$), the conservation law [eq. (11)] is always satisfied regardless of other variables and, thus, the equations of motion can generate the proper NPT ensemble. In this case, the number of degrees of freedom is decreased by three. If one divides the system into pieces coupled with different thermostats (e.g., protein and solvent molecules are coupled with different thermostats), the conservation law [eq. (11)] breaks. In this case, the proper NPT ensemble is also generated. The problem due to the additional conservation law was already reported for the NVT ensemble and the isotropic NPT ensemble.[8,9] Here, they are extended to the fully flexible cell.

The equations of motion satisfy the tensorial (or pressure) and kinetic virial theorems,

$$\langle(\overleftrightarrow{\mathbf{P}}_{\text{int}})_{\alpha\beta} - P_{\text{ext}}\delta_{\alpha\beta}\rangle = 0,$$

$$\langle V[(\overleftrightarrow{\mathbf{P}}_{\text{int}})_{\alpha\beta} - P_{\text{ext}}\delta_{\alpha\beta}]\rangle = kT\delta_{\alpha\beta}. \quad (12)$$

The NPT Ensemble for the Orthorhombic Cell

Sometimes, special scaling in the NPT simulation is required to vary the lengths of the axes of the simulation cell independently, keeping the angles between the axes at 90° (the NPT ensemble for the orthorhombic cell). In this case, the equation of

$$(\dot{\overleftrightarrow{\mathbf{p}}}_g)_{\alpha\beta} = \begin{cases} V[(\overleftrightarrow{\mathbf{P}}_{\text{int}})_{\alpha\alpha} - P_{\text{ext}}] + \dfrac{1}{3N}\displaystyle\sum_{i=1}^{N}\frac{\mathbf{p}_i^2}{m_i} - \frac{p_{\eta_1}}{Q_1}(\overleftrightarrow{\mathbf{p}}_g)_{\alpha\alpha} & (\alpha = \beta) \\ 0 & (\alpha \neq \beta) \end{cases} \quad (13)$$

is used instead of eq. (2), and the initial values of the off-diagonal terms of $\overleftrightarrow{\mathbf{p}}_g$ are set at zero. N_b is 3, and the Jacobian J is

$$J = \exp\left[(N_f + 3)\eta_1 + \sum_{k=2}^{M}\eta_k\right]. \quad (14)$$

If there is an external force ($\Sigma_{i=0}^{N}\mathbf{F}_i \neq 0$), the partition function Δ is expressed as,

$$\Delta \propto \int dh_x dh_y dh_z d^N\mathbf{r} d^N\mathbf{p} d^{N_r}\mathbf{q} d^{N_r}\mathbf{L} \exp[-\beta(U + K + P_{\text{ext}}h_x h_y h_z)], \quad (15)$$

and that partition function ensures that the equations of motion, eqs. (1) and (13), can generate the proper NPT ensemble for the orthorhombic cell.

If there is no external force ($\Sigma_{i=0}^{N}\mathbf{F}_i = 0$), the following additional conservation law is also satisfied:

$$e^{\eta_1}V^{1/(3N)}h_\alpha P_\alpha = C_\alpha, \quad \alpha = x, y, z. \quad (16)$$

Thus, the proper NPT ensemble can be generated in the following cases: when the momentum of the whole system is zero ($\mathbf{P} = 0$), or when multiple thermostats are applied to parts of the system. The equations of motion satisfy the diagonal parts of the tensorial and kinetic virial theorems [eq. (12)]. However, the off-diagonal parts of the tensorial and kinetic virial theorems are not satisfied.

When one needs to keep the condition $h_x = h_y$, the equation of

$$(\dot{\overleftrightarrow{\mathbf{p}}}_g)_{\alpha\alpha} = V\left\{\frac{1}{2}\left[(\overleftrightarrow{\mathbf{P}}_{\text{int}})_{xx} + (\overleftrightarrow{\mathbf{P}}_{\text{int}})_{yy}\right] - P_{\text{ext}}\right\} + \frac{1}{3N}\sum_{i=1}^{N}\frac{\mathbf{p}_i^2}{m_i} - \frac{p_{\eta_1}}{Q_1}(\overleftrightarrow{\mathbf{p}}_g)_{\alpha\alpha} \quad (\alpha = x \text{ or } y) \quad (17)$$

is used instead of eq. (13), and the initial velocities are set so that $(\overleftrightarrow{\mathbf{p}}_g)_{xx} = (\overleftrightarrow{\mathbf{p}}_g)_{yy}$. In this case, N_b is 2.

Specific Ensembles for Membrane Systems

For membranes, the thermodynamic properties of the pressure of the system are different between the lateral direction and normal direction. Thus, molecular dynamics simulations of membranes

sometimes use a different way to control the pressure between those directions.[11]

In this article, for membrane simulations, the equations of motion for the NPAT (constant normal pressure and lateral surface area of membranes and constant temperature) and the NPγT (constant normal pressure and lateral surface tension of membranes and constant temperature) ensembles are proposed by extending the method of the standard NPT ensemble described in the previous section. In the following sections, the lateral direction is defined as x or y, and the normal direction is defined as z.

The NPAT Ensemble

The equations of motion for the NPAT ensemble are eq. (1), and the equation of

$$(\dot{\overleftrightarrow{\mathbf{p}}}_g)_{\alpha\beta} = \begin{cases} V[(\overleftrightarrow{\mathbf{P}}_{\mathrm{int}})_{zz} - P_{\mathrm{ext}}] + \dfrac{1}{3N}\sum_{i=1}^{N}\dfrac{\mathbf{p}_i^2}{m_i} - \dfrac{p_{\eta_1}}{Q_1}(\overleftrightarrow{\mathbf{p}}_g)_{zz} & (\alpha = \beta = z) \\ 0 & (\alpha \neq z \text{ or } \beta \neq z) \end{cases}, \tag{18}$$

instead of eq. (2). The initial values of the elements of $\overleftrightarrow{\mathbf{p}}_g$, except $(\overleftrightarrow{\mathbf{p}}_g)_{zz}$, must be set at zero.

The different terms from the original equations of motion for the NPAT ensemble[11] are the third term on the right-hand side of the equation of $\dot{\mathbf{p}}_i$ in eq. (1) and the second term on the right-hand side of eq. (18). The difference enables us to generate the proper NPAT ensemble. In the NPAT ensemble, N_b is 1, and the Jacobian J takes the form,

$$J = \exp\left[(N_f + 1)\eta_1 + \sum_{k=2}^{M}\eta_k\right]. \tag{19}$$

If there is an external force ($\sum_{i=0}^{N}\mathbf{F}_i \neq 0$), the partition function Δ satisfies

$$\Delta \propto \int dh_z d^N\mathbf{r} d^N\mathbf{p} d^{N_r}\mathbf{q} d^{N_r}\mathbf{L} \exp[-\beta(U + K + P_{\mathrm{ext}}A_{xy}h_z)], \tag{20}$$

where A_{xy} represents the area of the xy plane of the cell. Equation (20) ensures generation of the proper NPAT ensemble.

If there is no external force ($\sum_{i=0}^{N}\mathbf{F}_i = 0$), the following additional conservation laws are also satisfied,

$$e^{\eta_1}h_z^{1/(3N)}P_x = C_x, \quad e^{\eta_1}h_z^{1/(3N)}P_y = C_y, \quad e^{\eta_1}h_z^{1/(3N)+1}P_z = C_z, \tag{21}$$

where P_x, P_y, and P_z are elements of the total momentum $\mathbf{P}$, and C_x, C_y, and C_z are constants. When the total momentum $\mathbf{P}$ is zero, eq. (21) is always true regardless of other variables, and the equations of motion can generate the proper NPAT ensemble.

If one divides the whole system into pieces coupled with different thermostats, the conservation laws [eq. (21)] break, and the equations of motion generate the proper NPAT ensemble.

The equations of motion for the NPAT ensemble satisfy only the $\alpha = \beta = z$ part of the tensorial and kinetic virial theorems [eq. (12)]. Other parts of the theorems are not satisfied.

The NPγT Ensemble

In regard to the equations of motion for the NPγT ensemble, instead of eq. (2), the following equation of $\dot{\overleftrightarrow{\mathbf{p}}}_g$ is used:

$$(\dot{\overleftrightarrow{\mathbf{p}}}_g)_{\alpha\beta} = \begin{cases} -A_{xy}(\gamma_\alpha - \gamma_{\mathrm{ext}}) + \dfrac{1}{3N}\sum_{i=1}^{N}\dfrac{\mathbf{p}_i^2}{m_i} - \dfrac{p_{\eta_1}}{Q_1}(\overleftrightarrow{\mathbf{p}}_g)_{\alpha\alpha} & (\alpha = \beta = x \text{ or } y) \\ V[(\overleftrightarrow{\mathbf{P}}_{\mathrm{int}})_{zz} - P_{\mathrm{ext}}] + \dfrac{1}{3N}\sum_{i=1}^{N}\dfrac{\mathbf{p}_i^2}{m_i} - \dfrac{p_{\eta_1}}{Q_1}(\overleftrightarrow{\mathbf{p}}_g)_{zz} & (\alpha = \beta = z) \\ 0 & (\alpha \neq \beta) \end{cases}, \tag{22}$$

where γ_{ext} is the external surface tension, and the instantaneous surface tension γ_α is defined as,

$$\gamma_\alpha = -h_z[(\overleftrightarrow{\mathbf{P}}_{\mathrm{int}})_{\alpha\alpha} - P_{\mathrm{ext}}]. \tag{23}$$

The initial values of the off-diagonal terms of $\overleftrightarrow{\mathbf{p}}_g$ are set at zero. Inserting eq. (23) into eq. (22),

$$(\dot{\overleftrightarrow{\mathbf{p}}}_g)_{\alpha\alpha} = V[(\overleftrightarrow{\mathbf{P}}_{\mathrm{int}})_{\alpha\alpha} - P_{\mathrm{ext}}] + A_{xy}\gamma_{\mathrm{ext}} + \frac{1}{3N}\sum_{i=1}^{N}\frac{\mathbf{p}_i^2}{m_i} - \frac{p_{\eta_1}}{Q_1}(\overleftrightarrow{\mathbf{p}}_g)_{\alpha\alpha}, \quad \alpha = x, y \tag{24}$$

is obtained. If $\gamma_{\mathrm{ext}} = 0$, eq. (22) or (24) becomes equivalent to eq. (13) in the NPT ensemble for the orthorhombic cell. In this ensemble, N_b is 3. The Jacobian J is expressed as,

$$J = \exp\left[(N_f + 3)\eta_1 + \sum_{k=2}^{M}\eta_k\right]. \tag{25}$$

If there is an external force ($\sum_{i=0}^{N}\mathbf{F}_i \neq 0$), the partition function is expressed as

$$\Delta \propto \int dh_x dh_y dh_z d^N\mathbf{r} d^N\mathbf{p} d^{N_r}\mathbf{q} d^{N_r}\mathbf{L} \times \exp[-\beta(U + K + P_{\mathrm{ext}}h_x h_z h_z - \gamma_{\mathrm{ext}}h_x h_y)]. \tag{26}$$

Equation (26) ensures generation of the proper NPγT ensemble.

If there is no external force ($\sum_{i=0}^{N}\mathbf{F}_i = 0$), the following additional conservation laws are also satisfied:

$$e^{\eta_1} V^{1/(3N)} h_\alpha P_\alpha = C_\alpha, \quad \alpha = x, y, z. \tag{27}$$

Thus, the proper NPγT ensemble can be generated as it is in the NPAT ensemble, when the momentum of the whole system is zero ($\mathbf{P} = 0$), or when multiple thermostats are applied to pieces of the systems.

The equations of motion satisfy the $\alpha = \beta = z$ part of the tensorial and kinetic virial theorems [eq. (12)], and the following related equations:

$$\langle \gamma_\alpha - \gamma_{\text{ext}} \rangle = 0, \quad \langle A_{xy}[\gamma_\alpha - \gamma_{\text{ext}}] \rangle = kT. \tag{28}$$

If one needs to keep $h_x = h_y$, instead of eq. (24), the equation

$$(\dot{\overleftrightarrow{\mathbf{p}}}_g)_{\alpha\alpha} = V\left\{\frac{1}{2}[(\overleftrightarrow{\mathbf{P}}_{\text{int}})_{xx} + (\overleftrightarrow{\mathbf{P}}_{\text{int}})_{yy}] - P_{\text{ext}}\right\} + A_{xy}\gamma_{\text{ext}} + \frac{1}{3N}\sum_{i=1}^{N}\frac{\mathbf{p}_i^2}{m_i} - \frac{p_{\eta_1}}{Q_1}(\overleftrightarrow{\mathbf{p}}_g)_{\alpha\alpha} \quad (\alpha = x \text{ or } y) \tag{29}$$

is used. The initial values of $\overleftrightarrow{\mathbf{p}}_g$ must be set at $(\overleftrightarrow{\mathbf{p}}_g)_{xx} = (\overleftrightarrow{\mathbf{p}}_g)_{yy}$. In this case, N_b is 2.

Numerical Integration for NPT, NPAT, and NPγT Ensembles

The equations of motion for the NPT, NPAT, and NPγT ensembles can be integrated by the following approximate evolution operator:

$$\exp(i\mathcal{L}\Delta t) = \exp\left(i\mathcal{L}_b\frac{\Delta t}{2}\right)\exp(i\mathcal{L}_s\Delta t)\exp\left(i\mathcal{L}_b\frac{\Delta t}{2}\right) + O(\Delta t^3), \tag{30}$$

where $i\mathcal{L}_s$ is the Liouville operator for equations of rigid-body motion and is described in the next section. $i\mathcal{L}_b$ is the Liouville operator of the thermostats and the barostats for the NPT, NPAT, and NPγT ensembles, and takes the form,

$$\begin{aligned}
\exp\left(i\mathcal{L}_b\frac{\Delta t}{2}\right) &= \exp\left(i\mathcal{L}^1_{T_M}\frac{\Delta t}{4}\right)\prod_{k=M-1}^{1}\left[\exp\left(i\mathcal{L}^2_{T_k}\frac{\Delta t}{8}\right)\exp\left(i\mathcal{L}^1_{T_k}\frac{\Delta t}{4}\right)\exp\left(i\mathcal{L}^2_{T_k}\frac{\Delta t}{8}\right)\right]\exp\left(i\mathcal{L}^2_P\frac{\Delta t}{8}\right)\exp\left(i\mathcal{L}^1_P\frac{\Delta t}{4}\right)\exp\left(i\mathcal{L}^2_P\frac{\Delta t}{8}\right)\\
&\times \exp\left(i\mathcal{L}^1_b\frac{\Delta t}{2}\right)\exp\left(i\mathcal{L}^2_P\frac{\Delta t}{8}\right)\exp\left(i\mathcal{L}^1_P\frac{\Delta t}{4}\right)\exp\left(i\mathcal{L}^2_P\frac{\Delta t}{8}\right)\prod_{k=1}^{M-1}\left[\exp\left(i\mathcal{L}^2_{T_k}\frac{\Delta t}{8}\right)\exp\left(i\mathcal{L}^1_{T_k}\frac{\Delta t}{4}\right)\exp\left(i\mathcal{L}^2_{T_k}\frac{\Delta t}{8}\right)\right]\exp\left(i\mathcal{L}^1_{T_M}\frac{\Delta t}{4}\right),\\
i\mathcal{L}^1_{T_k} &= G_k\frac{\partial}{\partial p_{\eta_k}}, \qquad i\mathcal{L}^2_{T_k} = \frac{p_{\eta_{k+1}}}{Q_{k+1}}p_{\eta_k}\frac{\partial}{\partial p_{\eta_k}}, \qquad i\mathcal{L}^1_P = \sum_{\alpha\beta}(\overleftrightarrow{\mathbf{G}}_g)_{\alpha\beta}\frac{\partial}{\partial(\overleftrightarrow{\mathbf{p}}_g)_{\alpha\beta}}, \qquad i\mathcal{L}^2_P = -\sum_{\alpha\beta}\frac{p_{\eta_1}}{Q_1}(\overleftrightarrow{\mathbf{p}}_g)_{\alpha\beta}\frac{\partial}{\partial(\overleftrightarrow{\mathbf{p}}_g)_{\alpha\beta}},\\
i\mathcal{L}^1_b &= -\sum_i^N\left(\frac{\overleftrightarrow{\mathbf{p}}_g}{W_g} + \frac{1}{3N}\frac{\mathrm{Tr}[\overleftrightarrow{\mathbf{p}}_g]}{W_g} + \frac{p_{\eta_1}}{Q_1}\right)\mathbf{p}_i\cdot\nabla_{\mathbf{p}_i} - \sum_i^{N_r}\frac{p_{\eta_1}}{Q_1}\mathbf{L}_i\cdot\nabla_{\mathbf{L}_i} + \sum_k^M\frac{p_{\eta_k}}{Q_k}\frac{\partial}{\partial\eta_k}.
\end{aligned} \tag{31}$$

Only by changing the definition of $\overleftrightarrow{\mathbf{G}}_g$ can various ensembles be generated. In the NPT ensemble for the fully flexible cell, $\overleftrightarrow{\mathbf{G}}_g$ takes the form,

$$(\overleftrightarrow{\mathbf{G}}_g)_{\alpha\beta} = V[(\overleftrightarrow{\mathbf{P}}_{\text{int}})_{\alpha\beta} - P_{\text{ext}}\delta_{\alpha\beta}] + \frac{1}{3N}\sum_{i=1}^{N}\frac{\mathbf{p}_i^2}{m_i}\delta_{\alpha\beta}. \tag{32}$$

In the NPT ensemble for the orthorhombic cell, the off-diagonal parts in eq. (32) are set at zero, and the initial values of the off-diagonal terms of $\overleftrightarrow{\mathbf{p}}_g$ are also set at zero.

In the NPAT ensemble, $(\overleftrightarrow{\mathbf{G}}_g)_{xx}$ and $(\overleftrightarrow{\mathbf{G}}_g)_{yy}$ are also set at zero, in addition to the off-diagonal terms. The initial values of $\overleftrightarrow{\mathbf{p}}_g$ are also set at zero, except $(\overleftrightarrow{\mathbf{p}}_g)_{zz}$.

In the NPγT ensemble, instead of eq. (32), $\overleftrightarrow{\mathbf{G}}_g$ takes the form,

$$(\overleftrightarrow{\mathbf{G}}_g)_{\alpha\beta} = \begin{cases} V[(\overleftrightarrow{\mathbf{P}}_{\text{int}})_{\alpha\alpha} - P_{\text{ext}}] + A_{xy}\gamma_{\text{ext}} + \dfrac{1}{3N}\displaystyle\sum_{i=1}^{N}\frac{\mathbf{p}_i^2}{m_i} & (\alpha = \beta = x \text{ or } y)\\ V[(\overleftrightarrow{\mathbf{P}}_{\text{int}})_{zz} - P_{\text{ext}}] + \dfrac{1}{3N}\displaystyle\sum_{i=1}^{N}\frac{\mathbf{p}_i^2}{m_i} & (\alpha = \beta = z)\\ 0 & (\alpha \neq \beta) \end{cases}. \tag{33}$$

Numerical Integration for Rigid-Body Motion

The equations of rigid-body motion can be integrated by the following approximate evolution operator:

$$\exp(i\mathcal{L}\Delta t) = \exp\left(i\mathcal{L}^4\frac{\Delta t}{2}\right)\exp\left(i\mathcal{L}^3\frac{\Delta t}{2}\right)\exp\left(i\mathcal{L}^2\frac{\Delta t}{2}\right)\exp(i\mathcal{L}^1\Delta t),$$

$$\times \exp\left(i\mathcal{L}^2\frac{\Delta t}{2}\right)\exp\left(i\mathcal{L}^3\frac{\Delta t}{2}\right)\exp\left(i\mathcal{L}^4\frac{\Delta t}{2}\right), i\mathcal{L}^1 = i\mathcal{L}_t^1 + i\mathcal{L}_r^1,$$

$$i\mathcal{L}^2 = i\mathcal{L}_r^2, \quad i\mathcal{L}^3 = i\mathcal{L}_r^3, \quad i\mathcal{L}^3 = i\mathcal{L}_r^3, \quad i\mathcal{L}^4 = i\mathcal{L}_t^4 + i\mathcal{L}_r^4, \quad (34)$$

where $i\mathcal{L}_t^k$ is the Liouville operator of translational motion and deformation of the cell, and $i\mathcal{L}_r^k$ is the Liouville operator of rotational motion. $i\mathcal{L}_t^k$ is expressed as,

$$i\mathcal{L}_t^1 = \sum_i^N \left[\frac{\mathbf{p}_i}{m_i} + \frac{\overleftrightarrow{\mathbf{p}}_g}{W_g}\mathbf{r}_i\right]\cdot\nabla_{\mathbf{r}_i} + \sum_{\alpha\beta}\frac{(\overleftrightarrow{\mathbf{p}}_g\overleftrightarrow{\mathbf{h}})_{\alpha\beta}}{W_g}\frac{\partial}{\partial(\overleftrightarrow{\mathbf{h}})_{\alpha\beta}},$$

$$i\mathcal{L}_t^4 = \sum_i^N \mathbf{F}_i\cdot\nabla_{\mathbf{p}_i}. \quad (35)$$

To numerically solve the equations of rotational motion, this article used the symplectic integrator for rigid bodies, which was recently proposed by Miller et al.[6] The integrator divides the rotational Liouville operator into four pieces. $\mathcal{L}_1$, $\mathcal{L}_2$, and $\mathcal{L}_3$ take the form,

$$\exp(i\mathcal{L}_r^k\Delta t)\mathbf{q} = \cos(\zeta_k\Delta t)\mathbf{q} + \sin(\zeta_k\Delta t)\mathbf{P}_k\mathbf{q},$$

$$\exp(i\mathcal{L}_r^k\Delta t)\mathbf{p}_r = \cos(\zeta_k\Delta t)\mathbf{p}_r + \sin(\zeta_k\Delta t)\mathbf{P}_k\mathbf{p}_r,$$

$$\zeta_k = \frac{1}{4I_k}\mathbf{p}_r^T\mathbf{P}_k\mathbf{q}, \quad (k = 1, 2, 3), \quad \mathbf{P}_1\mathbf{q} = (-q_1, q_0, q_3, -q_2),$$
$$\mathbf{P}_2\mathbf{q} = (-q_2, -q_3, q_0, q_1), \quad \mathbf{P}_3\mathbf{q} = (-q_3, q_2, -q_1, q_0), \quad (36)$$

where the suffix i representing the molecular number and its summation is omitted for simplicity, and $I_1 = I_{xx}$, $I_2 = I_{yy}$, $I_3 = I_{zz}$. The rotational momentum $\mathbf{p}_r$ is related to the angular momentum $\mathbf{L}$ in the following way:

$$\mathbf{p}_r = 2\overleftrightarrow{\mathbf{S}}\mathbf{L}^{(4)}, \quad \mathbf{L}^{(4)} = (0, L_x, L_y, L_z). \quad (37)$$

$\mathcal{L}_4$ takes the form,

$$e^{i\mathcal{L}_4\Delta t}\mathbf{p}_r = \mathbf{p}_r + \Delta t\cdot 2\overleftrightarrow{\mathbf{S}}\mathbf{T}^{(4)}, \quad \mathbf{T}^{(4)} = (0, T_x, T_y, T_z). \quad (38)$$

This integrator is not only reversible but also symplectic. In the microcanonical ensemble, it was shown that this integrator provides excellent conservation of the total energy.[6] However, when the integrator is combined with the equations of motion for the NPT ensemble, the whole integrator is no longer symplectic (but still reversible). The loss of the symplectic property is inevitable in the extended systems.[22]

For comparison, the other rigid-body integrator proposed by Matubayasi and Nakahara[5] is also used in this article. This integrator was recently combined with the NPT ensemble.[12] In that integrator, the rotational Liouville operator is divided in the following way:

$$i\mathcal{L}_r^1 = \frac{1}{2}[\overleftrightarrow{\mathbf{S}}\boldsymbol{\omega}^{(4)}]\cdot\nabla_{\mathbf{q}},$$

$$i\mathcal{L}_r^2 = \left[\frac{(I_{yy} - I_{zz})}{I_{xx}}\omega_y\omega_z\right]\frac{\partial}{\partial\omega_x} + \left[\frac{(I_{zz} - I_{yy})}{I_{zz}}\omega_y\omega_x\right]\frac{\partial}{\partial\omega_z},$$

$$i\mathcal{L}_r^3 = \left[\frac{(I_{zz} - I_{xx})}{I_{yy}}\omega_z\omega_x\right]\frac{\partial}{\partial\omega_y} + \left[\frac{(I_{xx} - I_{zz})}{I_{zz}}\omega_y\omega_x\right]\frac{\partial}{\partial\omega_z},$$

$$i\mathcal{L}_r^4 = \frac{T_x}{I_{xx}}\frac{\partial}{\partial\omega_x} + \frac{T_y}{I_{yy}}\frac{\partial}{\partial\omega_y} + \frac{T_z}{I_{zz}}\frac{\partial}{\partial\omega_z} \quad (39)$$

In this article, the two rigid-body integrators in the NPT ensemble are compared.

Parallelization

A number of parallelization methods for molecular dynamics simulations have been proposed so far.[18–20] Among them, the spatial-decomposition approach has the lowest communication cost between processors. In the spatial-decomposition approach, the space of the system is divided into subdomains.[19,21] Each processor computes the forces of atoms in one or a few subdomains. For the computation, each processor needs data of N/P atoms. Thus, the communication cost of the spatial decomposition is proportional to N/P.

In parallel computation, due to the communication overhead, there is a limitation in the parallel efficiency to solve small problems in size. However, if the system size is large enough, highly efficient parallel computation becomes possible. To solve large problems with a large number of processors, the parallel program should be *scalable.* A parallel program is said to be *scalable,* if one can find the size of a system that can be solved in the same parallel efficiency as increasing the number of processors.[21] For *scalability,* the communication-to-computation ratio must remain the same while increasing the number of processors. Among the parallelization methods for molecular dynamics simulations, the spatial-decomposition approach is *scalable,*[21] because both the computation cost and the communication cost are proportional to N/P. Therefore, in the molecular dynamics program MARBLE, the spatial-decomposition approach is used to parallelize the program.

If there are rigid bodies in the system, the way of spatial decomposition is not straightforward. It is useful that all atoms of a rigid body belong to the same processor because the integration of the equations of motion of the rigid body can be performed by the processor. However, because atoms are partitioned by their coordinates in the usual spatial-decomposition approach, all atoms of a rigid body do not always belong to the same processor. To solve this problem, the following two approaches can be considered. The first solution is that atoms are partitioned by coordinates of representative points of rigid bodies.[21] In this method, the size of the subdomain must be increased by the radii of the largest rigid bodies to avoid the drop of nonbond atom pairs for the calculation of interactions. In the case of rigid bodies of hydrogen groups, 2 to

2.5 Å must be added to the size of the subdomain. This may lead to an increase in the communication and computational cost of calculating nonbond interactions. Due to the decrease of the number of subdomains, the load balancing may become more difficult.

In the MARBLE, an alternative method is developed, by which atoms are partitioned by definition of subdomains, and an atom list for transferring across processors is prepared. When atoms belong to a processor that is different from that of rigid bodies, the list is modified so that the processor transferring data is corrected to be the actual processor to which the rigid body belongs. In this method, the size of the subdomain remains the same as that of fully flexible atomic molecular dynamics. Thus, the increase in the communication cost and inefficiency of load balancing due to rigid-body dynamics can be minimized.

In spatial decomposition, load balancing across processors is difficult, and this leads to severe parallel inefficiency. Recently, by introducing the dynamic load-balancing technique, in which computation tasks are balanced during execution, the problems were overcome and the efficiency of parallelization was greatly improved.[21] MARBLE also adopts dynamic load balancing in the same way as the NAMD2.[21]

Computational Details

To test the efficiency of the rigid-body dynamics for biological systems, molecular dynamics simulations were performed for an aqueous protein and for a fully hydrated lipid bilayer. For the aqueous protein simulation, Ubiquitin (PDB id: 1UBQ), which has 76 residues, was used; 7333 water molecules were placed around the protein, and the final system consisted of 23,230 atoms. For the force field of protein, AMBER99[15] was used. The TIP3P model[23] was used for water. The electrostatic interactions were calculated using the particle mesh Ewald method, and the Lennard–Jones interactions were switched to zero over a range of 10–12 Å. The temperature was set at 298.15 K, the pressure was 1 atm, and isotropic scaling was used for the NPT simulation. After the 500-ps equilibration, three 1-ns product runs were performed using the two partial rigid-body methods[5,6] and the SHAKE/RATTLE method. In the SHAKE/RATTLE method, atomic group scaling[24] was used. The tolerance of the SHAKE/RATTLE method was set at 10^{-8}. The time step of the integrators was set at 2 fs, which is the standard value in biomolecular simulations. In addition, to assess the accuracy of the integrators with longer time steps, additional simulations with time steps set at 3 and 4 fs were performed using the symplectic rigid-body method and the SHAKE/RATTLE method.

Although the SHAKE/RATTLE method constrains only the length of bonds involving hydrogen, the partial rigid-body method constrains the bond angles of hydrogen-heavy atom–hydrogen, in addition to the bond length. Thus, the model of the partial rigid-body method is slightly different from the model of the SHAKE/RATTLE method. To estimate the effect of the bond angle constraints on simulation results, the results of the partial rigid-body dynamics are compared with those of the SHAKE/RATTLE method in terms of the dynamics of protein. For this purpose, two additional sets of four 1-ns simulations of the same Ubiquitin system using the partial rigid-body method and the SHAKE/RATTLE method were performed, in which the time step was set at 2 fs.

In this article, the equations of motion in the NPT ensemble are extended to the membrane-specific ensembles, and the equations of motion in these ensembles are combined with the rigid-body integrators. To check the validity of the partial rigid-body dynamics in the membrane simulations, the most frequently theoretically studied dipalmitoyl phosphatidylcholine (DPPC) bilayer was chosen for the test calculation. This lipid bilayer was also used in the development of the force field for lipids.[13] In the simulation here, the system consisted of 72 DPPC molecules and 2094 water molecules. For the initial structure, the coordinates constructed by Feller et al.[25] were used. For the force field of DPPC, CHARMM27[13] was used, and the modified TIP3P model[23] was used for water. In the same way as the development of CHARMM27,[13] the electrostatic interactions were calculated using the particle mesh Ewald method[16] and the Lennard–Jones interactions were switched to zero over a range of 8–10 Å. The temperature was set at 363 K, at which DPPC is in the liquid crystal phase, and the NPAT ensemble is used in the same way as the development of CHARMM27.[13] The time step was set at 2 fs. In the lipid bilayer simulation, to estimate the effect of the constraint of H—X—H, a 10-ns simulation was performed using each of the partial rigid-body method and the SHAKE/RATTLE method, and results of the partial rigid-body method were compared with those of the SHAKE/RATTLE method. In total, by more than 30-ns molecular dynamics simulations, the validity of the partial rigid-body method was tested.

To test the computation efficiency of the partial rigid-body method, a very large system was prepared, and calculation speeds were compared. An aqueous F1-ATPase (PDB id: 1BMF; about 3000 residues) was chosen for the test. About 80,000 water molecules and 32 K^+ ions were added around the F1-ATPase, and finally, a system of 287,281 atoms was constructed. Then, the calculation speeds of the partial rigid-body method and the SHAKE/RATTLE method were compared. The force field of protein was CHARMM22.[14] The electrostatic interactions were calculated using the particle mesh Ewald method. Two conditions in which the Lennard–Jones interactions were truncated at 10 and 12 Å were also tested. Other conditions were the same in the simulations, including the time step of 2 fs.

Results and Discussion

Total Energy Conservation

Total energy conservation is a good indicator for measuring the stability of molecular dynamics simulations. Theoretically, keeping the total energy at the same value is required to generate the proper ensembles. Figure 1 demonstrates that the total energies of aqueous protein simulations range within ± 15 kcal/mol, when the time step is 2 fs. Compared with the amount of the potential energy, $-7.3 \times 10^4 \pm 130$ kcal/mol, the deviations of the total energies was small (Table 1). However, the total energy of the method proposed previously,[12] in which the NPT integrator was combined with the rigid-body integrator proposed by Matubayasi and Nakahara,[5] showed a gradual drift during 1 ns, although the

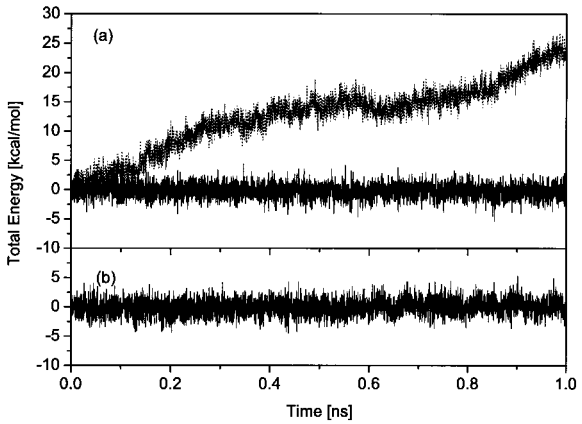

Figure 1. The total energies of molecular dynamics simulations for the aqueous Ubiquitin. (a) the simulation using the partial rigid-body method. For the rigid-body integrator, the methods reported by Miller et al.[6] (solid line) and by Matubayasi and Nakahara[5,12] (dotted line) were compared. (b) the simulation using the SHAKE/RATTLE method.

fluctuations of the total energies in the short time scale are almost the same between the two partial rigid-body methods. In contrast, the method proposed here, in which the NPT integrator was combined with the rigid-body integrator proposed by Miller et al.,[6] yields little drift during 1 ns. The trend is almost the same in the microcanonical ensemble,[6] suggesting that the integrator of the rigid-body rotation determines the stability of the whole integrator. Therefore, in the following analyses, only the rigid-body integrator proposed by Miller et al.[6] is used.

The SHAKE/RATTLE method also showed good total energy conservation (Fig. 1). However, the SHAKE/RATTLE method is not reversible nor symplectic because it has a finite tolerance. Although 1-ns simulations cannot distinguish the difference between the partial rigid-body method proposed here, and the SHAKE/RATTLE method (Fig. 1), in 10-ns simulations of the lipid bilayer, the SHAKE/RATTLE method showed a small gradual drift of the total energy (Fig. 2). However, the amounts of the drifts by both methods are very small, compared with the potential energy, $-2.1 \times 10^4 \pm 110$ kcal/mol (Table 2). This good conservation of the total energy by the SHAKE/RATTLE method may be due to the severe tolerance of 10^{-8}. It was reported that, while a tolerance of 10^{-6} was enough for total energy conservation in the NVT ensemble, a tolerance of 10^{-8} was required in the NPT ensemble.[12] When the tolerance is set at a smaller value, the computation cost becomes larger due to the increase in the number of iterations. Therefore, the tolerance should be determined, by considering the tradeoff between accuracy and computation cost.

To assess the accuracy of the integrators with longer time steps, additional simulations with time steps set at 3 and 4 fs were performed (Table 1). In the partial rigid-body method, the total energy conservation using the time step of 3 fs was almost same as that of 2 fs. However, the SHAKE/RATTLE method with 3 fs clearly showed the drift of the total energy. Both methods could not keep the total energy when the time step was 4 fs, although the partial rigid-body method was better than the SHAKE/RATTLE method.

Average pressures and densities of molecular dynamics simulations for the aqueous protein Ubiquitin using the partial rigid-body methods and the SHAKE/RATTLE method are compared in Table 1. In all methods, the pressure of the system was well controlled to the target pressure of 1 atm, despite the large fluctuation ranging over ± 158 atm. The average densities were within the range of the standard deviation in all simulations when the time step was 2 fs or 3 fs. When the time step was 4 fs, the average densities slightly decreased. In the simulation of the fully hydrated lipid bilayer, pressure normal to the membrane surface was well controlled to the target pressure of 1 atm (Table 2). In the lipid bilayer simulation, the NPAT ensemble was used, in which only the normal pressure was controlled during the simulation. Densities were also almost the same between the partial rigid-body method and the SHAKE/RATTLE method. Consequently, the

Table 1. Average Pressure, Density, Drift, and Noise of the Total Energy in the Aqueous Ubiquitin Simulation.

Method	Time step (fs)	Pressure (atm)	Density (g/cm^3)	Drift (kcal/mol/ns)	Noise (kcal/mol)
Partial rigid body[a]	2	1.00 (158.78)	0.9975 (0.0032)	−0.35	1.21
Partial rigid body[b]	2	1.00 (157.88)	0.9976 (0.0031)	19.06	2.07
SHAKE/RATTLE	2	1.00 (158.84)	0.9974 (0.0031)	0.58	1.35
Partial rigid body[a]	3	1.00 (158.94)	0.9961 (0.0031)	0.39	2.86
SHAKE/RATTLE	3	1.00 (158.22)	0.9955 (0.0030)	6.93	3.20
Partial rigid body[a]	4	1.00 (160.02)	0.9937 (0.0032)	265.2	6.51
SHAKE/RATTLE	4	1.00 (158.82)	0.9932 (0.0031)	878.1	12.12

The drift means the slope of the linear fit of the total energy against the time, and the noise means the standard deviation of residuals of the linear fit. Values in parentheses show the standard deviation during simulation.

[a]Partial rigid-body method in which the symplectic rigid-body integrator proposed by Miller et al.[6] is combined with the integrator for the NPT ensemble.

[b]Partial rigid-body method in which the reversible rigid-body integrator proposed by Matubayasi and Nakahara[5] is combined with the integrator for the NPT ensemble.

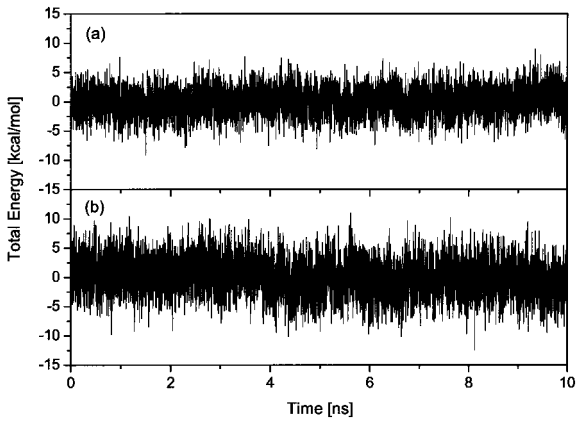

Figure 2. The total energies of molecular dynamics simulations for the fully hydrated DPPC lipid bilayer. (a) The simulation using the partial rigid-body method. For the rigid-body integrator, the method reported by Miller et al.[6] was used. (b) the simulation using the SHAKE/RATTLE method.

partial rigid-body method can provide almost the same average pressures and densities as the SHAKE/RATTLE method.

Dynamics of Proteins

The partial rigid-body method constrains both the bond lengths involving hydrogen and the bond angles of H—X—H (H: hydrogen, X: heavy atom), while the SHAKE/RATTLE method constrains only the bond lengths involving hydrogen. Thus, the model of the partial rigid-body method is slightly different from the model of the SHAKE/RATTLE method. Here, the effect of the constraint of the bond angles of H—X—H on the dynamics of proteins is examined.

The fluctuation of proteins was evaluated using the root-mean-square fluctuation (RMSF) of the C_α of Ubiquitin. Because the RMSF does not converge so fast, four 1-ns simulations were additionally performed using the partial rigid-body method and also four 1-ns simulations using the SHAKE/RATTLE method (eight in total). In Figure 3, the RMSF values of five 1-ns runs using the partial rigid-body method were compared with those of the SHAKE/RATTLE method. The overall trends of the RMSF are identical between the two methods, although the deviations of individual runs are large. The partial rigid-body method seems to slightly diminish the fluctuation. The slight decrease of the fluctuation may be the effect of the constraint on the bond angle of H—X—H.

Figure 4 shows the matrices of the correlated motion of Ubiquitin by the two methods. The element of the correlation matrix C_{ij} was calculated by,

$$C_{ij} = \frac{\langle \Delta \mathbf{r}_i \cdot \Delta \mathbf{r}_j \rangle}{\sqrt{\langle \Delta \mathbf{r}_i^2 \rangle \langle \Delta \mathbf{r}_j^2 \rangle}}. \tag{40}$$

In the calculation of the correlation matrix, only C_α atoms were considered. In Figure 4, the average values of five runs are shown. The patterns of the correlated motion of the protein are essentially identical between the two methods, as well as the fluctuations of residues.

Dynamics of Membranes

Next, the dynamics of the DPPC lipid bilayer is examined. The experimental quantities, which are often compared with simulations in validating the structures of lipid bilayers, are the order parameters S_{CD}, which can be obtained from the NMR experiment. From the simulations, the order parameters can be calculated using,

$$S_{CD} = \left\langle \frac{1}{2} (3 \cos^2\theta - 1) \right\rangle, \tag{41}$$

where θ is the angle between the C—H bond vector of the fatty acids and the bilayer normal. In Figure 5, the partial rigid-body simulation and the SHAKE/RATTLE simulation are compared with the experimental order parameters. The agreement of both simulation results with the experiments is excellent, and discrepancies with the experiments are comparable to the differences in experimental order parameters reported independently.

In Figure 6, the electron density profiles along the membrane normal are compared among two simulations and experiments. The latter half of 10-ns simulations was divided into five 1-ns

Table 2. Average Normal Pressure, Density, Drift, and Noise of the Total Energy in the Fully Hydrated DPPC Bilayer Simulation.

Method	P_{zz} (atm)	Density (g/cm^3)	Drift (kcal/mol/ns)	Noise (kcal/mol)
Partial rigid body[a]	1.00 (549.87)	0.9936 (0.0042)	0.105	2.25
SHAKE/RATTLE	1.00 (551.62)	0.9943 (0.0043)	−0.194	3.03

The drift means the slope of the linear fit of the total energy against the time, and the noise means the standard deviation of residuals of the linear fit. Values in parentheses show the standard deviation during simulation. All simulations were performed using the time step of 2 fs.
[a]Partial rigid-body method in which the symplectic rigid-body integrator proposed by Miller et al.[6] is combined with the integrator for the NPT ensemble.

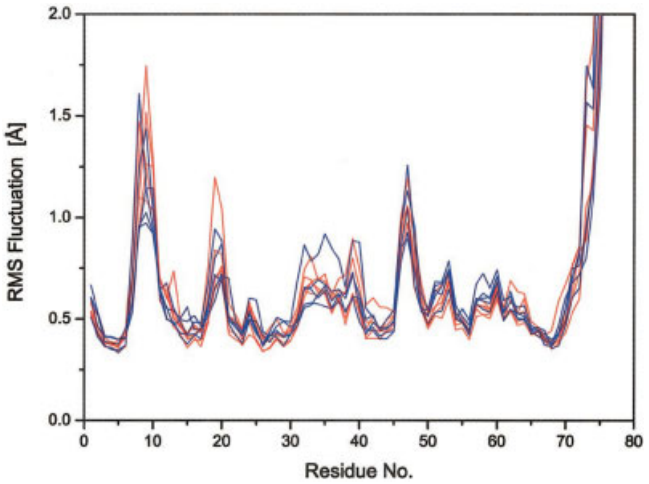

Figure 3. The root mean fluctuation of Ubiquitin. Results of five runs of the partial rigid-body method (red) are compared with those of five runs of the SHAKE/RATTLE method (blue).

blocks. For each block, the electron density was calculated and is shown in Figure 6. Two experimental results, estimated using a different interpretation of the same X-ray diffraction data,[26] are also shown in Figure 6. The simulation results agree well with the experimental results, and the discrepancies between the two simulation results are small again.

Consequently, in the molecular dynamics simulation of the lipid bilayer, the partial rigid-body method provides essentially the same results as the SHAKE/RATTLE method as in the case of the dynamics of proteins.

Efficiency of Parallelization

In the large-scale parallel computation, it is important that the program is *scalable,* as described in the Methods section. There is a parallel computation efficiency limitation for small-size problems. However, highly efficient calculation for large-size problems is possible. Here, a large molecular system that consists of 287,281 atoms was constructed, and then the computation efficiency of the partial rigid-body method was estimated. In this test calculation, the particle mesh Ewald method was employed for evaluation of the electrostatic interactions. Treatment of long-range electrostatic interaction is indispensable in current molecular dynamics simulations of biomolecules.

In Figure 7, the computational speed of the partial rigid-body method is compared with that of the SHAKE/RATTLE method. Two conditions in which the Lennard–Jones interactions were truncated at 10 and 12 Å were also tested. Other conditions were the same in the simulations, including the time step of 2 fs. The partial rigid-body method does not require the iterations that are necessary in the SHAKE/RATTLE method. Thus, the partial rigid-body method was faster than the SHAKE/RATTLE method over the whole range of the number of processors. In particular, even the partial rigid-body method with a 12 Å truncation was faster than the SHAKE/RATTLE method with a 10 Å truncation when more than 16 processors were used in the calculation. The computational time of the SHAKE/RATTLE method depends on the tolerance of the bond lengths. In these calculations, the tolerance of 10^{-8} was used in the same way as aqueous Ubiquitin simulations and DPPC simulations. As shown in Figure 7, the SHAKE/RATTLE method takes 5.3 days for 1 ns using 128 processors in Alpha Server SC. When the tolerance was set at 10^{-6}, the com-

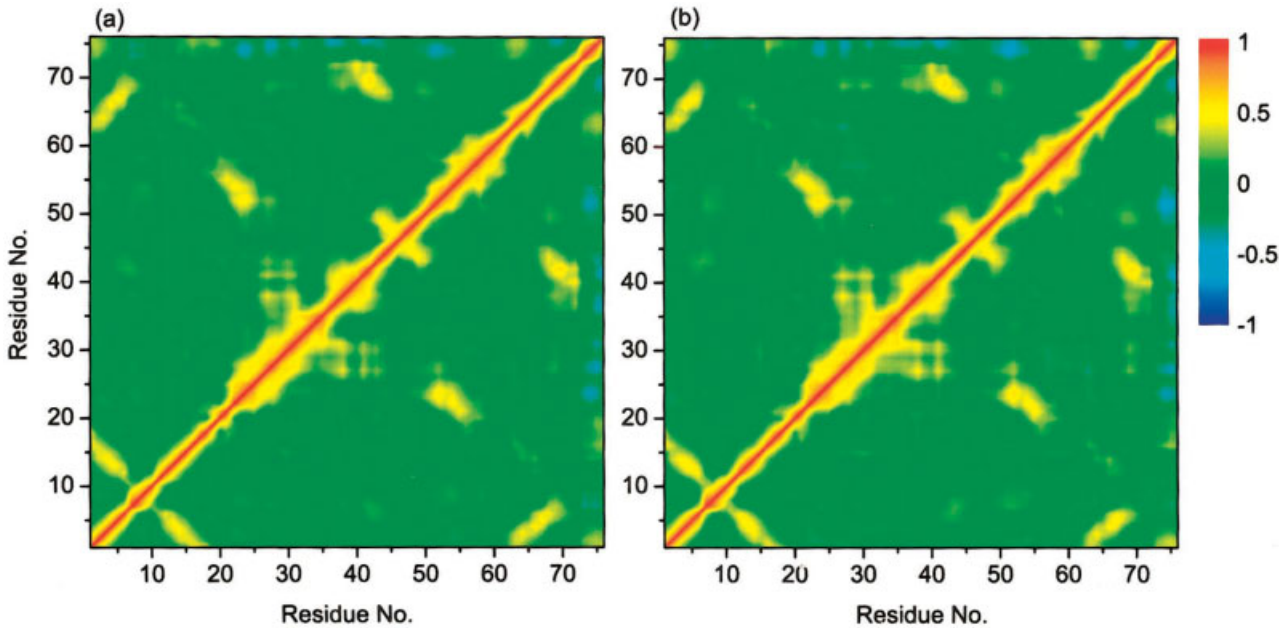

Figure 4. Matrices of the correlated motion of Ubiquitin. (a) The simulation using the partial rigid-body method. (b) The simulation using the SHAKE/RATTLE method.

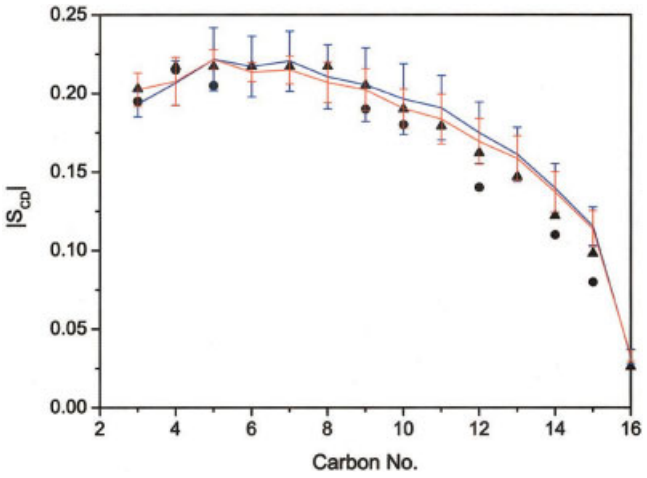

Figure 5. The order parameters of the DPPC lipid bilayer. The results of the partial rigid-body method (red) and those of the SHAKE/RATTLE method (blue) are compared. The error bar shows the standard deviation of blocks for every 1 ns. The filled circles and the filled triangles represent experimental values reported by Seelig and Seelig[27] and by Douliez et al.,[28] respectively.

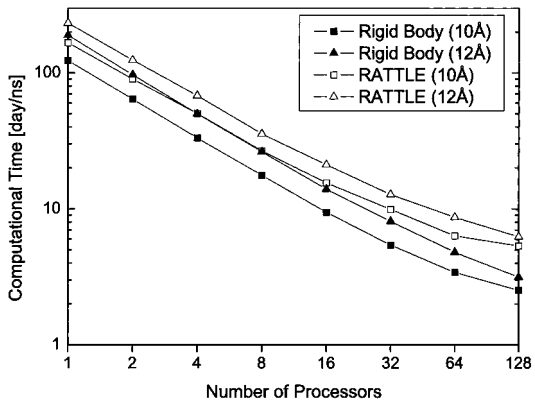

Figure 7. Comparison of computational times for aqueous F1-ATPase (287,281 atoms). The filled squares and triangles show the computational time of the rigid-body method using 10 and 12 Å cutoff distances, respectively, for the Lennard–Jones interactions. The open squares and triangles show the time of the SHAKE/RATTLE method using 10 and 12 Å cutoff distances. The electrostatic interactions were calculated using the particle mesh Ewald method.[16]

putation time decreased to 4.5 days for 1 ns. However, when the partial rigid-body method was used, the simulation of a 287,281-atom system took 2.5 days for 1 ns using 128 processors.

Conclusion

In this article, the symplectic integrator of rigid-body dynamics is combined with the NPT ensemble. The equations of motion for the NPT ensemble are extended to the membrane-specific ensembles, that is, the NPAT and NPγT ensembles. More than 30-ns simulations for aqueous proteins and lipid bilayers validate the partial rigid-body method in terms of total energy conservation, the protein dynamics and the lipid dynamics. In the partial rigid-body method, the bond angles of H—X—H (X: heavy atom) are constrained, as is the bond length. The simulation results indicate that the effect of the angle constraint is small in the overall trend of molecular dynamics. Furthermore, the computation of the partial rigid-body method is shown to be efficient in the environment of parallel computation.

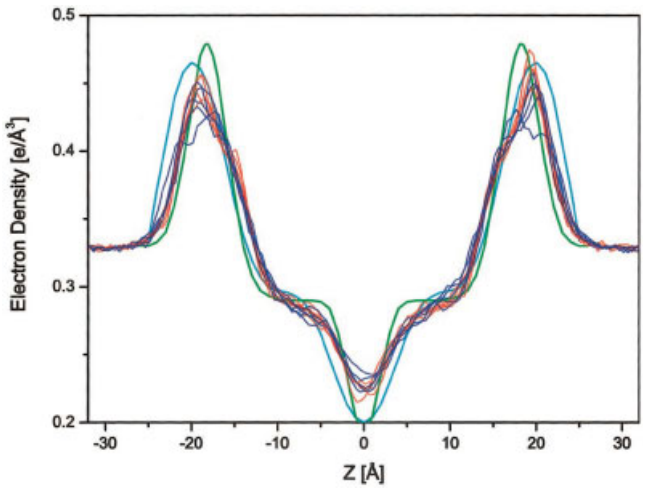

Figure 6. Electron density profile along the membrane normal. The red line and the blue line give the results of five runs of the partial rigid-body simulation and five runs of the SHAKE/RATTLE method, respectively. The green and cyan lines are two interpretations of the X-ray diffraction data.[26]

Acknowledgments

I thank Dr. A. Kidera and Mr. M. Hashido at Yokohama City University, Dr. Y. Sugita at the University of Tokyo, and Dr. J. G. Kim at the Japan Biological Information Research Center (JBIRC) for their useful discussions. The computations were done at the SBSS of Yokohama City University.

References

1. Karplus, M.; McCammon, J. A. Nat Struct Biol 2002, 9, 646.
2. Fersht, A. R.; Daggett, V. Cell 2002, 108, 573.
3. Berendsen, H. J. C.; Hayward, S. Curr Opin Struct Biol 2000, 10, 165.
4. Allen, M. P.; Tildesley, D. J. Computer Simulation of Liquids; Oxford: New York, 1987.
5. Matubayasi, N.; Nakahara, M. J Chem Phys 1999, 110, 3291.
6. Miller, T. F., III; Eleftheriou, M.; Pattnaik, P.; Ndirango, A.; Newns, D.; Martyna, G. J. J Chem Phys 2002, 116, 8649.
7. Martyna, G. J.; Tobias, D. J.; Klein, M. L. J Chem Phys 1994, 101, 4177.
8. Tuckerman, M. E.; Martyna, G. J. J Phys Chem B 2000, 104, 159.

9. Tuckerman, M. E.; Liu, Y.; Ciccotti, G.; Martyna, G. J. J Chem Phys 2001, 115, 1678.
10. Martyna, G. J.; Tuckerman, M. E.; Tobias, D. J.; Klein, M. L. Mol Phys 1996, 87, 1117.
11. Zhang, Y.; Feller, S. E.; Brooks, B. R.; Pastor, R. W. J Chem Phys 1995, 103, 10252.
12. Shinoda, W.; Mikami, M. J Comput Chem 2003, 24, 920.
13. Feller, S. E.; MacKerell, A. D., Jr. J Phys Chem B 2000, 104, 7510.
14. MacKerell, A. D., Jr.; Bashford, D.; Bellott, M.; Dunbrack, R. L., Jr.; Evanseck, J. D.; Field, M. J.; Fischer, S.; Gao, J.; Guo, H.; Ha, S.; Joseph–McCarthy, D.; Kuchnir, L.; Kuczera, K.; Lau, F. T. K.; Mattos, C.; Michnick, S.; Ngo, T.; Dguyen, D. T.; Prodhom, B.; Reiher, W. E., III; Roux, B.; Schlenkrich, M.; Smith, J. C.; Stote, R.; Straub, J.; Watanabe, M.; Wiórkiewicz–Kuczera, J.; Yin, D.; Karplus, M. J Phys Chem B 1998, 102, 3586.
15. Wang, J.; Cieplak, P.; Kollman, P. A. J Comput Chem 2000, 21.
16. Essmann, U.; Perera, L.; Berkowitz, M. L.; Darden, T.; Lee, H.; Pedersen, L. G. J Chem Phys 1995, 103, 8577.
17. Greengard, L. The Rapid Evaluation of Potential Fields in Particle Systems; MIT: Cambridge, 1987.
18. Smith, W. Comp Phys Commun 1991, 62, 229.
19. Plimpton, S. J Comput Phys 1995, 117, 1.
20. Plimpton, S.; Hendrickson, B. J Comput Chem 1996, 17, 326.
21. Kalé, L.; Skeel, R.; Bhandarkar, M.; Brunner, R.; Gursoy, A.; Krawetz, N.; Phillips, J.; Shinozaki, A.; Varadarajan, K.; Schulten, K. J Comput Phys 1999, 151, 283.
22. Ishida, H.; Kidera, A. J Chem Phys 1998, 109, 3276.
23. Jorgensen, W. L.; Chandrasekhar, J.; Madura, J. D.; Impey, R. W.; Klein, M. L. J Chem Phys 1983, 926, 926.
24. Marchi, M.; Procacci, P. J Chem Phys 1998, 109, 5194.
25. Feller, S. E.; Venable, R. M.; Pastor, R. W. Langmuir 1997, 13, 6555.
26. Nagle, J. F.; Zhang, R.; Tristram–Nagle, S.; Sun, W.; Petrache, H.; Suter, R. M. Biophys J 1996, 70, 1419.
27. Seelig, A.; Seelig, J. Biochemistry 1974, 13, 4839.
28. Douliez, J.-P.; Lénard, A.; Dufourc, E. Biophys J 1995, 68, 1727.