Simulation of peptide folding with explicit water—a mean solvation method

ARTICLE *in* PROTEINS STRUCTURE FUNCTION AND BIOINFORMATICS · FEBRUARY 1999

Impact Factor: 2.63 · DOI: 10.1002/(SICI)1097-0134(19990215)34:3<295::AID-PROT3>3.0.CO;2-T

PROTEINS: Structure, Function, and Genetics 34:295–302 (1999)

Simulation of Peptide Folding With Explicit Water—A Mean Solvation Method

Xiong-Wu Wu and Shen-Shu Sung*
The Lerner Research Institute, The Cleveland Clinic Foundation, Cleveland, Ohio

ABSTRACT **A new approach to efficiently calculate solvent effect in computer simulation of macromolecular systems has been developed. Explicit solvent molecules are included in the simulation to provide a mean solvation force for the solute conformational search. Simulations of an alanine dipeptide in aqueous solution showed that the new approach is significantly more efficient than conventional molecular dynamics method in conformational search, mainly because the mean solvation force reduced the solvent damping effect. This approach allows the solute and solvent to be simulated separately with different methods. For the macromolecule, the rigid fragment constraint dynamics method we developed previously allows large time-steps. For the solvent, a combination of a modified force-bias Monte Carlo method and a preferential sampling can efficiently sample the conformational space. A folding simulation of a 16-residue peptide in water showed high efficiency of the new approach. Proteins 1999;34:295–302.** © 1999 Wiley-Liss, Inc.

Key words: molecular dynamics; Monte Carlo method; computer simulation; mean solvation force; conformational search; protein folding

INTRODUCTION

Solvent plays a very important role in conformational studies of macromolecules in solution. However, the simulation of a macromolecule in solution needs to include a large number of solvent molecules, which increases the computation load dramatically. The fast motion of solvent molecules prevents large time-steps in molecular dynamics (MD) simulation. The existence of solvent molecules and their thermal motion have a damping effect on the conformation changes of macromolecules.

To overcome these difficulties, various continuum approaches for the solvent effect have been developed, such as the Brownian dynamics method,[1, 2] the solvent-accessible surface area-based method,[3–5] and the hydration shell volume-based method.[6–7] Further simplified methods, such as the approximate surface area calculation[8] and the pairwise approximation,[9] have also been proposed. How well these methods can mimic the solvent environment is still under investigation.

The continuum approach can represent an average solvent effect over different solvent configurations, but it does not provide detailed information about specific interactions, such as the hydrogen bond, between the macromolecule and solvent molecules. When solvent plays a structural role in the macromolecular conformation, explicit solvent molecules are necessary in the calculation. In order to overcome the difficulties resulting from the small time-step and the solvent damping effect, we propose a mean solvation method for macromolecular simulations in solution.[27]

MODEL AND METHODS

The basic idea of this method is to use the mean solvation force, instead of instant solute-solvent interaction, to guide the solute conformational search. Explicit solvent molecules are simulated to provide the mean solvation force. The solute does not interact directly with specific solvent molecules, and their damping effect on the solute conformation change is thus reduced. The simulations for solute and for solvent are separated, and, therefore, different methods can be used for the simulations of the solute and the solvent. To increase the computational efficiency of large systems, an efficient Monte Carlo (MC) method for solvent simulation may be used. For the macromolecular solute, the rigid fragment constraint dynamics algorithm we have developed previously[10] is a good choice for an efficient simulation.

Solute Simulation

The solute and solvent are separated into two subsystems, u and v, respectively. The force acting on an atom in subsystem u consists of the interactions within u and the interaction with v. Thus, the equation of motion for atom i of the solute molecule becomes

$$m_i \ddot{\mathbf{r}}_i = \mathbf{f}_i + \mathbf{s}_i \quad (1)$$

where $\mathbf{f}_i$ is the force from other solute atoms in subsystem u, $\mathbf{s}_i$ is that from the solvent in subsystem v, m_i is the mass, and $\ddot{\mathbf{r}}_i$ is the acceleration of atom i. The solute-solvent interaction depends on the conformation of both solute and solvent. For a specific solute conformation, the solvent can take many conformations with different solute-solvent

*Correspondence to: S. Sung, Cleveland Clinic Foundation/FF3, 9500 Euclid Avenue, Cleveland OH 44195. E-mail: sungs@cesmtp.ccf.org
Received 22 June 1998; Accepted 23 October 1998

interactions. The average effect of solvent on the motion of atom i can be expressed by the following equation:

$$m_i \langle \ddot{\mathbf{r}}_i \rangle_v = \mathbf{f}_i + \langle \mathbf{s}_i \rangle_v \tag{2}$$

where $\langle \rangle_v$ represents an ensemble average over the solvent conformation space. The ensemble average of solute-solvent interaction is the mean solvation force. Since the mean solvation force is an average over many solvent conformations for a given solute conformation, it depends on the solute conformation, not on instant solvent conformations. The mean solvation force can be calculated from a simulation on solvent for a given solute conformation,

$$\bar{\mathbf{s}}_i = \langle \mathbf{s}_i \rangle_v = \frac{1}{M} \sum_{j=1}^{M} \mathbf{s}_j \tag{3}$$

where M is the sampling size. Eq. (2) and (3) provide the basis of the mean solvation method. For each solute conformation, a MC or MD simulation of solvent can provide the mean solvation force according to Eq. (3). The results will be used in Eq. (2) to solve the equation of motion of the solute.

Because it is time-consuming to carry out a long solvent simulation for each solute conformation, we proposed the following approximate approach to calculate the mean solvation forces. In general, the conformation change of a macromolecule is much slower than the motion of solvent molecules. In other words, the mean solvation forces, which depend on the solute conformation, change very slowly. Therefore, all the solvent conformations sampled during a short period of time can be used to estimate the mean solvation forces for a solute conformation in this period. Such an approximation can increase the sampling size by many times, depending on the length of the period of time. The mean solvation forces for the current solute conformation can be estimated from the most recently sampled M solvent conformations. That is, Eq. (3) can be approximated as

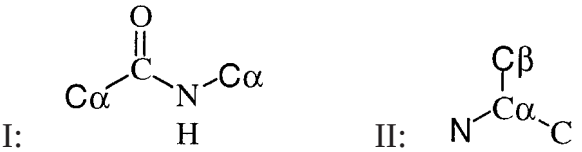
$$\bar{\mathbf{s}}_i^{(n)} = \frac{1}{M} \sum_{j=n-M+1}^{n} \mathbf{s}_j^{(j)} \tag{4}$$

where $\mathbf{s}_i^{(j)}$ represents the interaction between the atom i and solvent at conformation j. To simplify the mean solvation force calculation in a simulation, we derived the following equation from Eq. (4).

$$\begin{aligned} \bar{\mathbf{s}}_i^{(n)} &= \frac{1}{M} \left[\sum_{j=n-M+1}^{n-1} \mathbf{s}_i^{(j)} + \mathbf{s}_i^{(n)} \right] \\ &= \frac{M-1}{M} \frac{1}{M-1} \sum_{j=n-M+1}^{n-1} \mathbf{s}_i^{(j)} + \frac{1}{M} \mathbf{s}_i^{(n)} \\ &\cong (1-x) \bar{\mathbf{s}}_i^{(n-1)} + x \mathbf{s}_i^{(n)} \end{aligned} \tag{5}$$

Eq. (5) simplifies the calculation by updating the mean solvation forces from that at the previous conformations and the force at present conformation. The ensemble averaging factor, $x = 1/M$ is related to the sampling size. x takes a value between 0 and 1, depending on how fast the conformation changes. The slower the solute conformation changes, the more accurate the mean solvation force is. For a given system, a proper value of x should be carefully chosen and tested. Using a large x (a small M) may result in a large fluctuation in the mean solvation force. Using a small x (a large M) may result in a poorer approximation for fast conformation changes of the solute.

For the solute simulation, the rigid fragment constraint dynamics algorithm may be used to achieve high efficiency. The method has been described in detail previously,[10] and its basic features are outlined here. For semiflexible macromolecules, stable structural elements are treated as rigid fragments, and their motion is determined by rigid body mechanics. Rigid fragment constraints, defined as combinations of distance constraints and position constraints, are introduced to limit internal molecular motion to required mode. For example, the backbone of a peptide may be represented by two types of rigid fragments, I and II:



Two consecutive rigid fragments were constrained by two position constraints at nitrogen (or carbonyl carbon) and α-carbon atoms. Thus, only the ϕ and ψ dihedral angles, C-N-Cα-C and N-Cα-C-N, are variable (the ϕ, ψ peptide model).

The constraint forces are solved separately for each rigid fragment constraint and iteratively until all constraint conditions are satisfied within a given tolerance at each time-step, as is done for the bond length constraint in the SHAKE algorithm.[11] The orientation of a rigid fragment is represented using the quaternion parameters, and both translation and rotation are solved by the leap-frog formulation. This algorithm increases simulation efficiency through larger time-steps. In addition, because a large number of degrees of freedom are removed in a semiflexible model, the conformation space is reduced significantly. In a microcanonical ensemble simulation, the total energy was conserved satisfactorily with time-steps as large as 20 fsec. The new feature of the current study is the inclusion of the mean solvation forces in the equation of motion of the solute. At each MD step of the solute simulation, solvent is simulated to reach equilibrium (or near equilibrium) with the solute conformation to provide the mean solvation forces. The simulation for solvent is described below.

Solvent Simulation

At each MD step of the solute simulation, solvent molecules are simulated to provide the mean solvation

force. Many different methods may be used for solvent simulation. A simple and direct choice is to use the MD simulation for solvent as well. This method needs a long simulation to provide mean solvation forces of a reasonable accuracy. For small systems, this method is a straightforward approach.

For large systems, the computational efficiency becomes critical. Therefore, we developed an efficient MC method, including a modified force bias method and preferential sampling, to calculate the mean solvation forces for solute. Rao et al. have developed a force-bias method[12] and showed that it is two- to three-fold more efficient than the conventional Metropolis sampling method. In their method, molecules translate and rotate according to a distribution related to the forces acting on them.

$$T(\mathbf{R}'/\mathbf{R}) = \begin{cases} Q^{-1}(\mathbf{R}) \exp\left[\lambda\beta\mathbf{F}(\mathbf{R})\cdot\delta\mathbf{r} + \lambda\beta\mathbf{N}(\mathbf{R})\cdot\delta\boldsymbol{\alpha}\right] & \delta\mathbf{r},\ \delta\boldsymbol{\alpha} \in \mathrm{D} \\ 0 & \delta\mathbf{r},\ \delta\boldsymbol{\alpha} \notin \mathrm{D} \end{cases} \tag{6}$$

where $\delta\boldsymbol{r} = \boldsymbol{r}' - \mathbf{r}$ represents the displacement of the center of mass (COM), $\delta\boldsymbol{\alpha} = \delta\alpha\mathbf{n}$ is a vector representing the rotation (its direction is $\mathbf{n}$, the axis of rotation, and its magnitude is the rotation angle, $\delta\alpha$), $\mathbf{R}$ represents the configuration of the system, $\mathbf{F}(\mathbf{R})$ and $\mathbf{N}(\mathbf{R})$ are the force and torque on the molecule, λ is a parameter ranging from 0 to 1, and Q is a normalization constant depending on $\lambda\beta\mathbf{F}(\mathbf{R})$ and $\lambda\beta\mathbf{N}(\mathbf{R})$. The domain D is defined by

$$\begin{aligned} x - \frac{\Delta r}{2} \le x' \le x + \frac{\Delta r}{2} \\ y - \frac{\Delta r}{2} \le y' \le y + \frac{\Delta r}{2} \\ z - \frac{\Delta r}{2} \le z' \le z + \frac{\Delta r}{2} \\ -\frac{\Delta\alpha}{2} \le \delta\alpha \le \frac{\Delta\alpha}{2} \end{aligned} \tag{7}$$

where Δr and $\Delta\alpha$ specify the range of the translation and the rotation.

Examining the distribution function, Eq. (6), we found that it is biased not only according to the direction of the move relative to the force or torque, but also to its magnitude. In addition, with the displacement domain defined by Eq. (7), which forms a cube in the Cartesian coordinates, the move is also biased according to its direction in the coordinate system. A translation toward the corners of the domain has higher probability than elsewhere. For rotation, if the angle between the rotation axis, $\mathbf{n}$, and the torque is less than 90°, the most probable rotation angle will be $\Delta\alpha/2$, otherwise, $-\Delta\alpha/2$. If the force or torque is very strong, the move tends to concentrate to the boundary of the domain. To eliminate this problem, we modified the force-bias method of Rao at al.[12]

We adopted the following probability function instead:

$$T(\mathbf{R}'/\mathbf{R}) = \begin{cases} Q^{-1}(\mathbf{R}) \exp\left[\lambda\beta\Delta r\mathbf{F}(\mathbf{R})\cdot\mathbf{l} + \lambda\beta\Delta\alpha\mathbf{N}(\mathbf{R})\cdot\mathbf{n}\right] & \delta\mathbf{r},\ \delta\boldsymbol{\alpha} \in \mathrm{D} \\ 0 & \delta\mathbf{r},\ \delta\boldsymbol{\alpha} \notin \mathrm{D} \end{cases} \tag{8}$$

where $\mathbf{l}$ and $\mathbf{n}$ are the unit vectors in the directions of the translation and the rotation, and the actual move is given as

$$\delta\mathbf{r} = \delta r\mathbf{l} \tag{9a}$$

$$\delta\boldsymbol{\alpha} = \delta\alpha\mathbf{n} \tag{9b}$$

The displacement domain is defined as

$$\begin{aligned} 0 \le \delta r \le \Delta r \\ 0 \le \delta\alpha \le \Delta\alpha \end{aligned} \tag{10}$$

Therefore, the sampling is biased only according to the direction of the move relative to the force or the torque. The magnitudes of the translation and rotation are chosen randomly according to the following equations:

$$\begin{aligned} \delta r = \sqrt[3]{\xi_1}\,\Delta r \\ \delta\alpha = \xi_2\,\Delta\alpha \end{aligned} \tag{11}$$

where ξ_1 and ξ_2 are random numbers ranging from 0 to 1. The translation and rotation directions are force-biased according to the following equations:

$$\begin{aligned} \phi'_t &= 2\pi\xi_3 \\ \cos\theta'_t &= \frac{1}{\lambda\beta F\Delta r}\ln\left\{\xi_4[\exp(\lambda\beta F\Delta r) - \exp(-\lambda\beta F\Delta r)] + \exp(-\lambda\beta F\Delta r)\right\} \end{aligned} \tag{12}$$

$$\begin{aligned} \phi'_r &= 2\pi\xi_5 \\ \cos\theta'_r &= \frac{1}{\lambda\beta N\Delta\alpha}\ln\left\{\xi_6[\exp(\lambda\beta N\Delta\alpha) - \exp(-\lambda\beta N\Delta\alpha)] + \exp(-\lambda\beta N\Delta\alpha)\right\} \end{aligned} \tag{13}$$

where ξ is a random number ranging from 0 to 1, θ'_t is the angle between the translation direction and the force $\mathbf{F}$, ϕ'_t represents the direction (which is random between 0 and 2π) of the projection of the translation in the plane perpendicular to $\mathbf{F}$, and θ'_r, ϕ'_r are similarly defined for the direction of the rotational axis relative to the torque $\mathbf{N}$. Except for these modifications, other details of the method can be found in the original force-bias MC method.[12]

Of the large number of solvent molecules, only a small portion is located near the solute and make important contributions to solvation. The motion of the solvent

molecules far from the solute does not need to be updated very frequently. Therefore, Owicki's preferential sampling method[13] was used to focus more on solvent molecules near the solute.

To save computing time, the neighbor list technique was used to calculate interactions. Because updating the neighbor list also takes a significant portion of computer time and memory, solvent molecules are first selected (first selection) to have their neighbor list constructed according to a probability function $B_i(r^i_{\min})$, which decreases as the minimum distance of the solvent molecules from solute atoms increases. For the solvent molecules with a neighbor list, the moving molecules are chosen (second selection) according to a probability function $C_i(r^i_{\min})$. The overall probability for these two steps of selection is the product of $B_i(r^i_{\min})$, and $C_i(r^i_{\min})$, as in Eq. (14),

$$P_i(r^{\,i}_{min}) = B_i(r^{\,i}_{min})\, C_i(r^{\,i}_{min}), \tag{14}$$

with

$$B_i(r^{\,i}_{min}) = \frac{b_0 x^2 + b_1}{x^2 + b_1}$$

$$x = r^{\,i}_{min} - r_b + |r^{\,i}_{min} - r_b|$$

$$C_i(r^{\,i}_{min}) = \frac{c_0 y^2 + c_1}{y^2 + c_1}$$

$$y = r^{\,i}_{min} - r_c + |r^{\,i}_{min} - r_c| \tag{15}$$

where r^i_{min} is the minimum distance between solvent molecule i and solute atoms, and r_b, b_0, b_1, r_c, c_0, and c_1 are adjustable parameters. $B_i(r^i_{min})$ equals one for all solvent molecules within the range of r_b from a solute, and declines beyond this range. $C_{i,}(r^i_{\min})$, equals one within the range of r_c and declines beyond this range. Figure 1 shows the value of these probability functions.

In summary, the solvent simulation is carried out using a combination of the modified force-bias MC method and preferential sampling. The procedure for solvent simulation is outlined as follows:

1) At a certain time interval, neighbor lists are updated for solvent molecules chosen according to the first selection probability, $B_i(r^i_{min})$.
2) The solvent molecules in the first selection are further selected for trial move according to the second selection probability, $C_i(r^i_{min})$.
3) Each trial move is generated randomly according to the force-biased distribution function, Eq. (8).
4) At the new position, the force and torque are calculated to evaluate the reverse transition function, $T(\mathbf{R}/\mathbf{R}')$.
5) The new position is accepted if min $[1, [T(\mathbf{R}'/\mathbf{R})P(\mathbf{R}')]/[T(\mathbf{R}/\mathbf{R}')P(\mathbf{R})] \exp[-(\epsilon' - \epsilon)/kT]] > \xi$. Otherwise, the move is rejected.
6) Go to Step 1 until the end of the simulation.

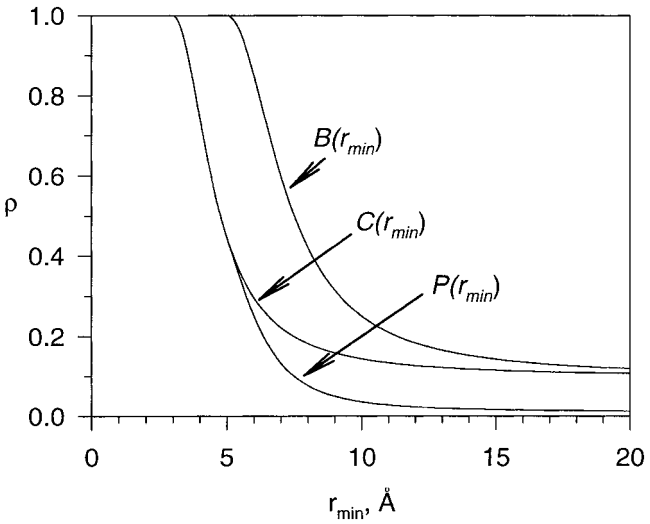


Fig. 1. Probability functions for preferential sampling. The neighbor lists of the solvent molecules are constructed according to the first selection probability $B(r_{min})$. The solvent molecules with a neighbor list are chosen to move according to the second selection probability $C(r_{min})$. The product of the two functions, $P(r_{min})$, gives the total probability of the two selections for each solvent molecule.

The simulations were carried out using atom-based molecular dynamics methods at T = 300 K. AMBER parameters[14] were used. Nonpolar hydrogen atoms were combined with their binding heavy atoms, as in the united atom model. An alanine dipeptide and a 16-residue synthetic peptide were tested. In the simulations, the peptide was placed in a box of equilibrium TIP3P water molecules.[15] The initial conformation was created by deleting water molecules whose oxygen atoms were within 2 Å of any peptide atoms, and energy-minimized to remove close contact. A cubic periodic boundary condition was applied in simulations. The neighbor list was updated every 10 MD steps.

RESULTS AND DISCUSSION

The mean solvation method has been tested on a simple system, the alanine dipeptide in aqueous solution, to study its efficiency in conformational search. The peptide-water system was contained in a 20 Å × 20 Å × 20 Å box, including 217 water molecules in addition to the dipeptide. For the alanine dipeptide, bond lengths were fixed with the SHAKE algorithm,[11] and a time-step of 2 fsec was used. Interactions were truncated at 8 Å.

The conformation of the alanine dipeptide was represented by the two dihedral angles, ϕ and ψ, because other internal coordinates show only small fluctuations around their equilibrium values. For comparison, we first carried out a conventional MD simulation for 500 psec. The ϕ and ψ angle distribution is shown in Figure 2. The peptide conformations were concentrated mainly in two regions. One was around (−70, 140), and the other around (−90, −40). Between these two regions there was a low-density gap, which corresponded to an energy barrier. Figure 3 shows the value of the dihedral angle, ψ, during the

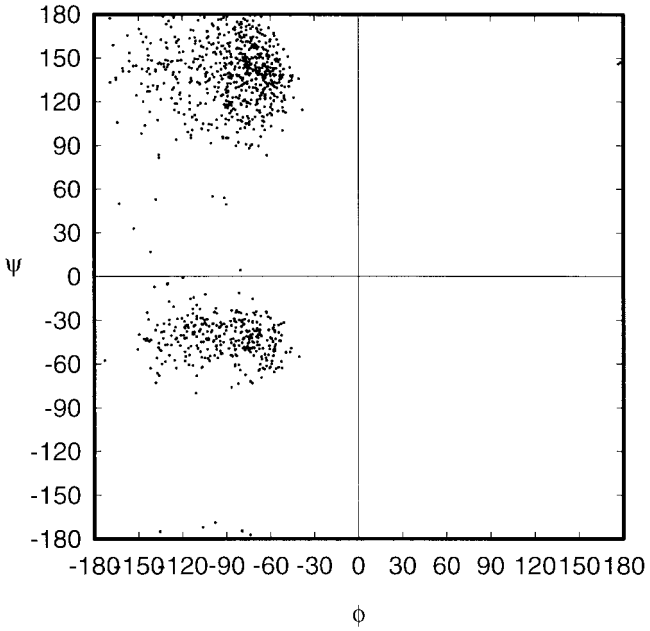


Fig. 2. The ϕ, ψ distribution of the alanine dipeptide in water in a conventional MD simulation.

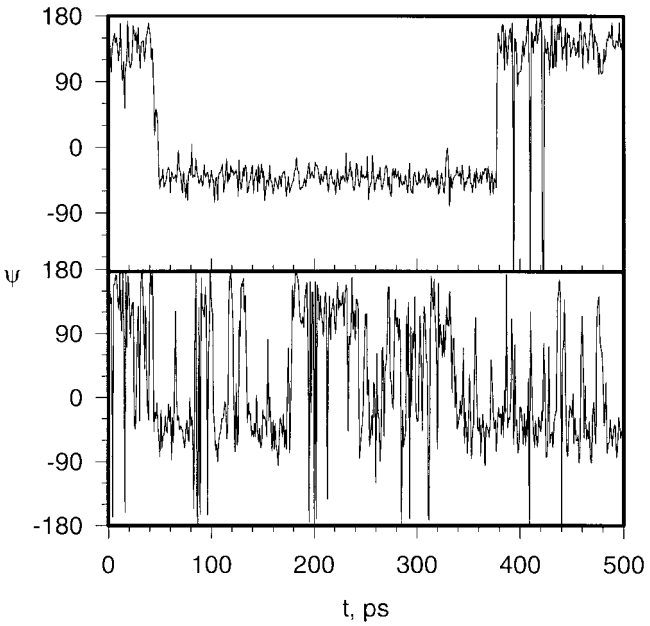


Fig. 3. The ψ angles of the alanine dipeptide during the simulation. The upper figure shows the angles during the conventional MD simulation, and the lower figure shows the angles during the mean solvation MD simulation.

simulation. During this 500-psec simulation only two transitions, at about 40 psec and 380 psec, were observed. Because only two transitions occurred, Figure 2 may not represent an accurate sampling of the transition between the low energy regions.

With the mean solvation method, the same system was simulated. In this simulation, the solute moved according to the mean solvation force in addition to the instant interactions among the solute atoms, while the solvent moved according to the instant interaction. An ensemble averaging factor of $x = 0.1$ was used. The difference between the previous MD simulation and this simulation is that the latter used the mean solvation force in the solute simulation, whereas the former used the instant interaction with the solvent.

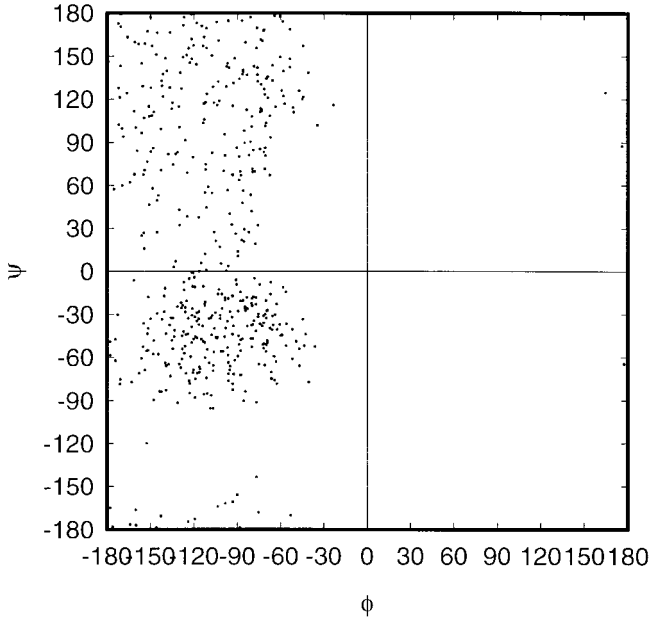


Fig. 4. The ϕ, ψ distribution of the alanine dipeptide in water in the mean solvation MD simulation.

The ϕ, ψ distribution from this simulation is shown in Figure 4. The distribution dispersed more than that in Figure 2.

In addition to the peptide conformations concentrated in the two low energy regions around (−70, 140) and (−90, −40), Figure 4 shows substantial distribution in the transition area between these two regions, which corresponds to the energy barrier. It means that the molecule spent more time in the transition region than in Figure 2 and the probability of overcoming energy barriers increased. The difference between Figure 2 and Figure 4 demonstrated that the mean solvation method increased the transition frequency between the two low energy regions. This acceleration is clearly shown by the ψ angle changes in Figure 3.

With the conventional MD method, only two major transitions were observed during the 500 psec period (upper figure). With the mean solvation method, much more transitions were observed during the same period (lower figure).

The agreement in the solution structures between the conventional MD simulation and the mean solvation force simulation was examined. Figure 5 shows the radial distribution function between atom H_5 (the first amide hydrogen) of the peptide and atom O_w of water, and

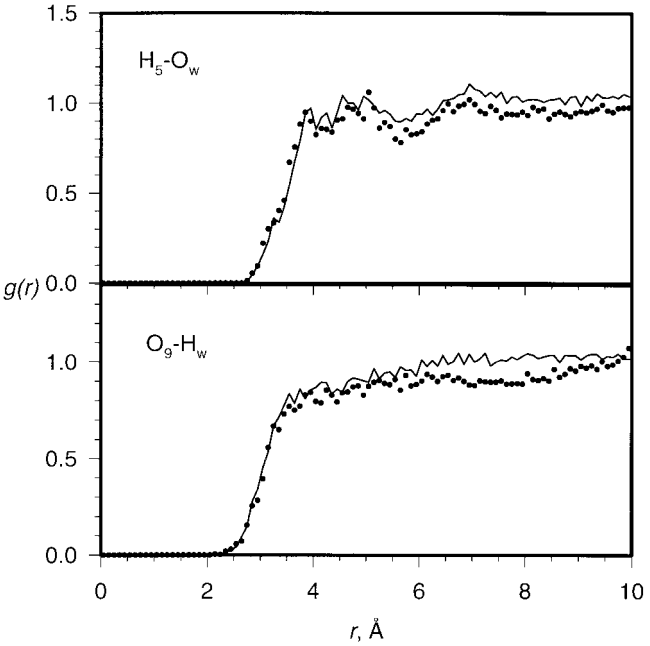
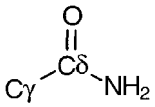


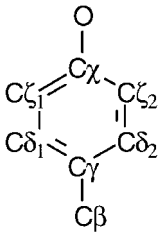
Fig. 5. The radial distribution functions from the conventional MD simulation (solid lines) and from the mean solvation MD simulation (filled circles). The upper figure is the radial distribution function between H_5 (the first amide hydrogen) of the alanine dipeptide and O_w of water, and the lower figure is the function between O_9 (the second carbonyl oxygen) of the dipeptide and H_w of water.

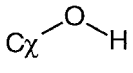
between atom O_9 (the second carbonyl oxygen) of the peptide and H_w of water. The two simulations show very similar radial distribution functions.

To test the method on longer chains, a 16-residue synthetic peptide[16] in aqueous solution was simulated. Its sequence is Ac-AAQAAAAQAAAAQAAY-NH_2. Circular dichroism measurement showed that it has about 50% helical content in aqueous solution. In this simulation, the rigid fragment constraint dynamics was used for the peptide and the efficient Monte Carlo method described previously was used for the solvent. The backbone of the 16-residue peptide was represented by a series of rigid fragments I and II. The side chain of each glutamine was represented by the rigid fragment III:

III: 

The rigid fragment III was constrained with backbone rigid fragment II by three distance constraints, between C_α and C_γ, C_β and C_γ, and C_β and C_δ, so that only the dihedral angles, X-C_α-C_β-X, X-C_β-C_γ-X, and X-C_γ-C_δ-X in the side chain were variable. The side chain of tyrosine were represented by two rigid fragments, IV and V:

IV: 

V: 

The rigid fragment IV was constrained with the backbone rigid fragment II by one position constraint at Cβ and a distance constraint between C_α and C_γ. The rigid fragment V was constrained with rigid fragment IV by two position constraints at C_χ and O. Only the dihedral angles X-C_α-C_β-X, X-C_β-C_γ-X, and X-C_χ-O-H in the side chain were variable. The terminal groups of the peptide were defined as rigid fragments similar to the rigid fragment I.

A time-step of 20 fsec was used for the peptide solute simulation using the rigid fragment constraint dynamics method. A tolerance of 0.0005 Å was used for rigid fragment constraints. Nonbonding interactions were truncated at 10 Å. The peptide was placed in a cubic box of 1,826 water molecules. Each side of the cubic box was 40 Å. The force-bias parameter was set as $\lambda = 0.2$ and the parameters in the preferential sampling were set as $b_0 = 0.1$, $b_1 = 10$, $r_b = 5$ Å, $c_0 = 0.1$, $c_1 = 10$, $r_c = 3$ Å. Thus, all the solvent molecules within 5Å of the solute atoms were selected for constructing their neighbor list. Beyond this range, solvents were selected according to the probability function $B(r^i)$, which has the minimum value 0.1. For every MD step of solute, the solvent molecules were simulated with the MC method for $2N$ steps, where $N = 1{,}826$ is the number of the solvent molecules. Because the water molecules near the peptide have a much higher probability of moving, 2N steps make the solvent conformation in (or near) equilibrium with the solute.

We simulated the folding process starting from a fully extended conformation. Helix folding was observed in a 1,000 psec simulation. Figure 6 shows some conformations during helix folding. As the simulation started, the peptide first relaxed from the fully extended conformation. The residue 9, among other residues, showed helical dihedral angles early in the simulation, but it did not propagate to form helical segments. At 75 psec, a hydrogen bond formed between the oxygen of residue 1 and the backbone hydrogen of residue 4. The first helix turn formed at the N terminus at about 85 psec. Then the helix turn propagated, forming a helical segment that included residues 1 to 5, at about 180 psec. The 3_{10}-helix and α-helix segment exchanged frequently. From 180 psec, this helix segment propagated to residue 11 within 20 psec. After another 100 psec, the helix segment propagated to residue 12. After that, the peptide conformation did not undergo major changes until 680 psec. At 720 psec, a complete helix formed.

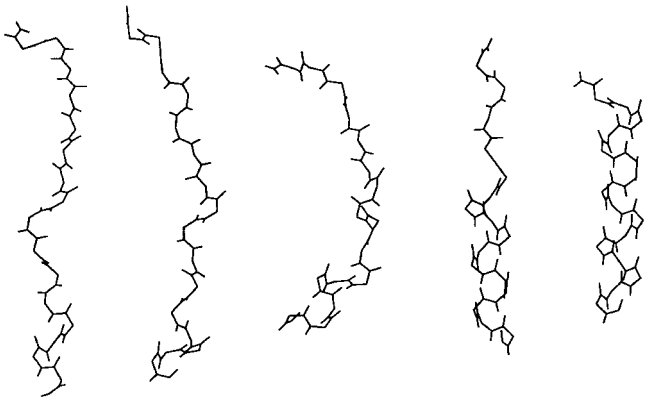


Fig. 6. Selected conformations during the simulation of the 16-residue synthetic peptide. From left to right, they are the conformations at 75, 85, 180, 200, and 720 psec. The side chains are not shown and the N-terminal of the peptide is at the bottom.

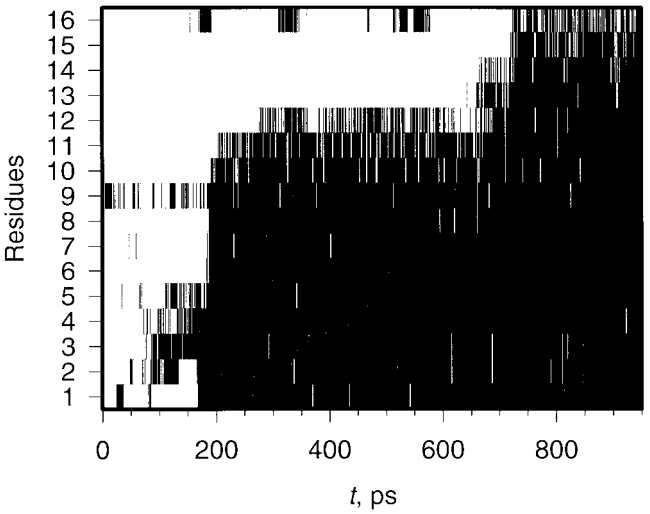


Fig. 7. Helix segment locations during the simulation of the synthetic peptide at 300K. The horizontal axis is the simulation time, and the vertical axis is the residue number in the sequence. If the ϕ, ψ dihedral angles of a residue are within 30 degrees of the standard helical values [−57, −47], the residue is defined as helical and is represented by a vertical bar at its position in the sequence.

As in our previous studies,[17,18] a helix segment plot has been used to illustrate the folding process, as shown in Figure 7. In this figure, the horizontal axis is the simulation time, and the vertical axis is the residue number in the sequence. If the ϕ, ψ dihedral angles of a residue are within 30 degrees of the standard helical values [−57, −47], the residue is defined as helical and is represented by a vertical bar at its position in the sequence. This figure clearly shows the folding process of the synthetic peptide. Helix folding of the peptide has been simulated previously with the solvent effect included in the energy function.[17,18] With explicit solvent molecules, the helical initial conformation or extra helical constraints were usually used for simulations.[19–22] In contrast, our simulation was carried out with non-helical initial conformation and with explicit

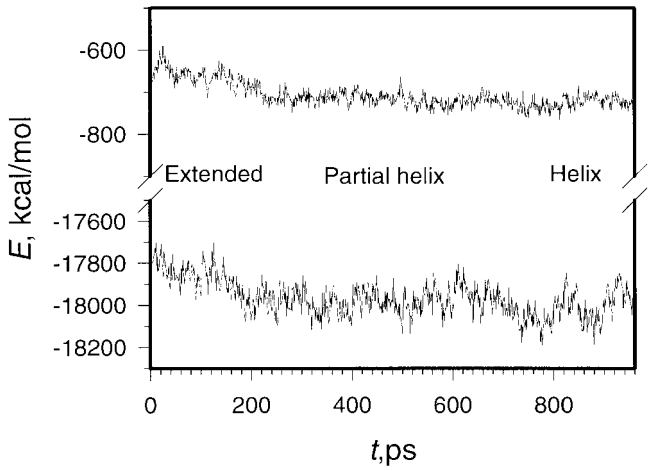


Fig. 8. Interaction energies of the synthetic peptide during the simulation. The upper curve is the energy of the peptide, including both the solute-solute and the solute-solvent interactions, and the lower curve is the total potential energy of the peptide-solvent system.

water molecules. This simulation demonstrated the efficiency of the mean solvation method in the solute conformational search.

According to the peptide conformation, the simulation may be divided into three periods: the extended period from 0 to 200 psec (including the transition period from the extended conformation to partial helix), the partial helical period from 200 to 700 psec, and the complete helical period from 700 to 1000 psec. From 0 to 200 psec, there is a large energy decrease, as shown in Figure 8. At about 700 psec, corresponding to forming the complete helix, there is a small energy decrease, which is less clear because the energy fluctuation is large. Kinetically, the folding is accelerated because the solvent damping effect is reduced by the mean solvation approach. The current study focuses on the efficiency of the conformational search. The discussion of the folding mechanism and the adequacy of the force field for folding simulations can be found in other publications.[17,18,23,24]

This simulation demonstrated the high efficiency of the new approach in conformational search. Because this method allows separate simulations for the solute and the solvent, large time-steps could be used for the solute MD simulation and the efficient MC method for the solvent. The rigid fragment simulation increased the time-steps for solute by 10 times, and the efficient MC simulation for the solvent took about the same amount of CPU time as the MD simulation of the solute. Therefore, the total computation time may be reduced by about an order of magnitude compared with the conventional MD method. For the simulation of the 16-residue synthetic peptide in water on the SGI Indigo2 R10000 computer, the CPU time needed for 1 psec dynamics was 1.5 minutes using the mean solvation method, and 12 minutes using the conventional MD method. An eight-fold reduction of the computing time has been achieved.

Furthermore, the conformation change in the new approach was much faster than in the conventional MD

simulation because the solvent damping effect is reduced. The synthetic peptide folded into a helix within 1,000 psec using the mean solvation method. With the conventional MD method, a simulation was carried out for 2000 psec, and the peptide remained as a random coil. Experimental evidences showed that the helix folding time scale is in the order of 10 to 100 nanoseconds.[25–26] The mean solvation method accelerated the folding by a factor of 10 to 100, making a helix folding process accessible in a MD simulation. The synthetic peptide simulation (1000 psec) took about 25 hours CPU time on the SGI R10000 computer. Assuming that the simulation of a small protein (of less than 100 amino acids) takes about 10 times more CPU time than the synthetic peptide for each time-step of simulation, the protein folding events with the real time scale in the order of 10 to 100 nanoseconds will takes 10 days using the mean solvation method. If this estimate is reasonable, the folding events shorter than 1 microsecond are accessible computationally with weeks of calculations or with supercomputers. Experimentally, protein folding from a denatured state to the native state usually takes milliseconds to minutes. Further increase of the efficiency of the conformational search is needed to simulate the whole folding process.

CONCLUSIONS

We have developed a mean solvation method for efficiently simulating the conformation changes of macromolecules with explicit solvent molecules. A mean solvation force, instead of instant solute-solvent interaction, was used in the equation of motion of the solute. An approximate approach to efficiently calculate the mean solvation force was proposed. Our test simulation on an alanine dipeptide in solution showed that the mean solvation force approach reduced the solvent damping effect and accelerated solute conformational search.

By using the mean solvation force in the equation of motion, solute and solvent can be simulated with different methods. Large time-steps may be applied for solute simulation by using the rigid fragment constraint dynamics algorithm we developed previously. An efficient MC approach, which combines the force-bias method and preferential sampling, can be used for the solvent simulation. In our modified force-bias MC method, the translation and rotation are biased only according to their directions, avoiding the uneven distribution of the move. Preferential sampling focuses on the solvent molecules near the solute to reduce the computation cost. The test on a 16-residue synthetic peptide in water demonstrated the high efficiency of the method for peptide-folding simulations.

REFERENCES

1. Ermak DL, McCammon JA. Brownian dynamics with hydrodynamic interactions. J Chem Phys 1978;69:1352–1360.
2. Evans GT. Liquid state dynamics of alkane chains. ACS Adv Chem Ser 1983;204:423.
3. Wesson L, Eisenberg D. Atomic solvation parameters applied to molecular dynamics of proteins in solution. Protein Sci 1992;1:227–235.
4. Lee B, Richards FM. The interaction of protein structures: Estimate of static accessibility. J Mol Biol 1971;55:379–400.
5. Richmond TJ. Solvent accessible surface area and excluded volume in proteins, analytical equations for overlapping spheres and implications for the hydrophobic effect. J Mol Biol 1984;178: 63–89.
6. Kang YK, Gibson KD, Nemethy G, Scheraga HA. Free energies of hydration of solute molecules. 4. Revised treatment of the hydration shell model. J Phys Chem 1988;91:4739–4742.
7. Gibson KD, Scheraga HA. Exact calculation of the volume and surface area of fused hard-sphere molecules with unequal atomic radii. Mol Phys 1987;62:1247–1265.
8. Wodak SJ, Janin J. Analytical approximation to the accessible surface area of proteins. Proc Natl Acad Sci USA 1980;77:1736–1740.
9. Kurochkina N, Lee B. Hydrophobic potential by pairwise surface area sum. Protein Eng 1995;8:437–442.
10. Wu XW, Sung SS. A constraint dynamics algorithm for the simulation of semiflexible macromolecules. J Comput Chem 1998; 19:1555–1566.
11. Ryckaert JP, Ciccotti G, Berendsen HJC. Numerical integration of the cartesian equations of motion of a system with constraints: Molecular dynamics of n-alkanes. J Comput Phys 1977;23: 327–341.
12. Rao M, Pangali C, Berne BJ. On the force bias Monte Carlo simulation of water: Methodology, optimization, and comparison with molecular dynamics. Mol Phys 1979;37: 1773–1798.
13. Owicki JC. In: Lukos PG, editor. Computer modeling of matter. Washington, DC: American Chemical Society; 1978. p 159–171.
14. Weiner SJ, Kollman PA, Case DA, et al. A new force field for molecular mechanical simulation of nucleic acids, and proteins, J Am Chem Soc 1984;106:765–784.
15. Jorgensen WL, Chandreskhar J, Madura JD, Impey RW, Klein ML. Comparison of simple potential functions for simulating liquid water. J Chem Phys 1983;79:926–935.
16. Scholtz JM, York EJ, Stewart JM, Baldwin RL. A neutral, water-soluble, α-helical peptide: The effect of ionic strength on the helix-coil equilibrium, J Am Chem Soc 1991;113: 5102–5104.
17. Sung SS, Wu XW. Molecular dynamics simulations of synthetic peptide folding. Proteins 1996;25: 202–214.
18. Sung SS, Wu XW. Molecular dynamics simulations of helix folding: The effects of amino acid substitution. Biopolymers 1997; 42:633–644.
19. Tirado-Rives J, Maxwell DS, Jorgensen WL. Molecular dynamics and Monte Carlo simulations favor the α-helical form for alanine-based peptides in water. J Am Chem Soc 1993;115:11590–11593.
20. Daggett, V., Levitt, M. Molecular dynamics simulations of helix denaturation. J. Mol. Biol. 223:1121–1138, 1992.
21. Hermans J, Anderson AG, Yun RH. Differential helix propensity of small apolar side chains studied by molecular dynamics simulations. Biochemistry 1992;31:5646–5653.
22. Tobias DJ, Brooks CL. III Thermodynamics and mechanism of α helix initiation in alanine and valine peptides. Biochemistry 1991;30:6059–6070.
23. Park B, Levitt M. Energy functions that discriminate X-ray and near-native folds from well-constructed decoys. J Mol Biol 1996; 258:367–392.
24. Creighton TE, editor. Protein folding. New York: W.H. Freeman and Company; 1992. p 1–547.
25. Williams S, Causgrove TP, Gilmanshin R, et al. Fast events in protein folding: Helix melting and formation in a small peptide. Biochemistry 1996;35:691–697.
26. Thompson PA, Eaton WA, Hofrichter J. Laser temperature jump study of the helix-coil kinetics of an alanine peptide interpreted with a 'kinetic zipper' model. Biochemistry 1997;36:9200–9210.
27. Wu XW, Sung SS. Peptide folding simulations with explicit water–an equilibrium solvation method. Book of abstract, 212th ACS National Meeting, American Chemical Society, Orlando, FL, August 25–29, 1996. p comp 0225.