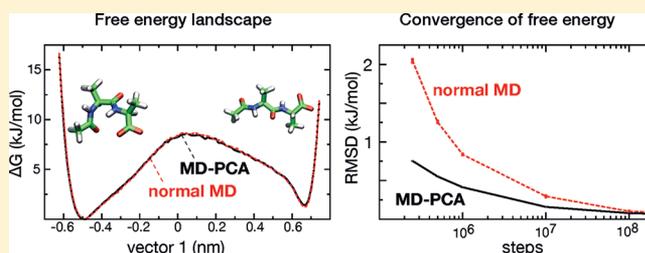


Molecular Dynamics in Principal Component Space

Servaas Michielssens,^{*,†} Titus S. van Erp,[‡] Carsten Kutzner,[‡] Arnout Ceulemans,[†] and Bert L. de Groot[‡][†]Department of Chemistry and [‡]Centre for Surface Chemistry and Catalysis, Katholieke Universiteit Leuven, B-3001 Leuven, Belgium[‡]Computational Biomolecular Dynamics Group, Max Planck Institute for Biophysical Chemistry, Göttingen, Germany

ABSTRACT: A molecular dynamics algorithm in principal component space is presented. It is demonstrated that sampling can be improved without changing the ensemble by assigning masses to the principal components proportional to the inverse square root of the eigenvalues. The setup of the simulation requires no prior knowledge of the system; a short initial MD simulation to extract the eigenvectors and eigenvalues suffices. Independent measures indicated a 6–7 times faster sampling compared to a regular molecular dynamics simulation.



INTRODUCTION

Protein dynamics is essential for function. However, protein dynamics, at the atomic level, is notoriously hard to probe experimentally. Therefore, molecular simulations are used routinely to study structure–function relationships in proteins. Due to the high dimensionality of biomolecules, this is a computationally demanding technique; for many biological processes, the time scales reached by simulation are too short to observe important functional processes in a statistically relevant manner. Thus, extensions of the current methods are necessary to address important biological problems. There are many attempts to enhance sampling of biomolecules. Here, we focus especially on algorithms that enhance sampling without using preselected order parameters. Some algorithms focus on rapid generation of new structures, while others focus on sampling of structures in a well-defined ensemble. Examples of the former are tConcord,¹ the anisotropic network model Monte Carlo (ANM-MC) method,² or Frodan.³ Examples of the latter are mostly extensions of the molecular dynamics (MD) or Monte Carlo (MC)⁴ methods, for example, replica exchange (RE),⁵ replica exchange with flexible tempering (REFT),⁶ library based MC,⁷ or accelerated MD.⁸ The first class has the advantage that it is fast and has more freedom in generating new structures. The disadvantage is that it is more difficult to derive thermodynamic properties from its results because the statistical weight of every structure in those ensembles is unknown. The second class of algorithms samples each structure with the right statistical weight but is generally computationally more intensive.

RE, for example, requires one to perform several simulations at different temperatures in parallel. The swapping between configurations of low and high temperatures speeds up the exploration of the conformational space of the cold systems compared to straightforward MD. Because the number of replicas needed to achieve an acceptable swapping probability increases rapidly with the system size, the temperature RE

method is in practice often not efficient for large biomolecular systems. In this article, we present a new algorithm of the second class, which gives a considerable speed up compared to MD without the need to perform multiple simulations in parallel.

One of the main problems of MD sampling for biomolecular systems is that the energy landscapes of biomolecules are highly anisotropic. As a result of this anisotropy, the time integration steps are limited by the fastest degrees of freedom (DOFs). A promising way to reduce these fast frequencies is to adjust the masses because they only affect the dynamics, not the ensemble. If one is only interested in ensemble properties, the masses given to the atoms can be chosen freely; they can even be assigned differently to each DOF. This idea has been exploited in numerous studies to achieve better sampling.^{9–14} Three different schemes can be distinguished. In a first scheme, the masses are chosen so that they compensate for the anisotropy of the energy surface. The simplest example of this is increasing the mass of hydrogen to attenuate the hydrogen vibration frequency; as a consequence, the time step can be increased.⁹ This scheme can be applied in a very general way by coupling a different mass to every DOF using a mass tensor.^{13,14} In this way, all vibrations have the same frequency, and the time step is no longer limited to very high frequencies that are mostly of little interest. A second scheme is lowering the mass of the solvent to reduce the solvent viscosity and increase the sampling.¹² Scaling the mass by a factor of λ decreases the viscosity by a factor of $\sqrt{\lambda}$. Lin and Tuckerman¹² applied this not only to the solvent but also to the side chains.

Special Issue: Macromolecular Systems Understood through Multi-scale and Enhanced Sampling Techniques

Received: October 17, 2011

Revised: December 21, 2011

Published: January 20, 2012

The mass of the solvent was scaled by a factor of 0.1, the backbone was scaled by 1 (i.e., physical masses were kept), and the side chains were scaled by an intermediate factor, thereby obtaining a decoupling of solvent, side chain, and backbone motions. A final scheme is increasing the mass to obtain larger momenta. This has been heavily used by Mao and co-workers.^{11,15–17} They generate velocities for the normal (lower) masses and combine those with the higher masses in the sampling. Therefore, as noticed by Berendsen and co-workers,⁹ they are essentially sampling at higher temperature, and there is a combined effect of the increased mass and the higher temperature; this results in an incorrect ensemble.

In this work, an algorithm is presented that changes the masses to compensate for the anisotropy of the potential energy surface. Principal component analysis (PCA) is used to identify the underlying anisotropy of the potential energy surface. This information will be used to make a suitable choice for the masses. Information from PCA of presampled structure ensembles has proven useful to enhance sampling.^{18,19} The PCA is carried out in Cartesian space by diagonalization of the variance–covariance matrix. The resulting eigenvectors or principal components (PCs) span a $3N$ -dimensional space, where N is the number of particles in the system. Therefore, all DOFs are present in this $3N$ -dimensional space, including bond length and angle fluctuations. PCA yields a global overview of the anisotropy of the different DOFs. Therefore, given a structure ensemble, PCA can be used to guide sampling algorithms to specifically enhance exploration of soft DOFs that are typically undersampled, for example, in MD.

Essential dynamics sampling¹⁸ and temperature-enhanced essential dynamics replica exchange (TEE-REX) are two examples of algorithms that utilize PCs from short MD runs or tConcord ensembles to successfully enhance sampling.¹⁹ However, these methods have in common that the generated ensembles are not well-defined and likely only approximately approach canonical ensembles. The novelty of our work is that we only change the masses according to the PCA eigenvalues, thereby guaranteeing canonical sampling.

METHODS

The Algorithm. In this work, an algorithm is presented where sampling is done in the space of the PCA vectors instead of in the Cartesian space. Using the leapfrog integrator²⁰ and the thermostat of Bussi et al.,²¹ we can write the update step as follows

$$\mathbf{u}\left(t + \frac{1}{2}\Delta t\right) = \alpha\mathbf{u}\left(t - \frac{1}{2}\Delta t\right) + \frac{\Delta t}{\boldsymbol{\mu}}\mathbf{g}(t) \quad (1)$$

$$\mathbf{q}(t + \Delta t) = \mathbf{q}(t) + \Delta t\mathbf{u}\left(t + \frac{1}{2}\Delta t\right) \quad (2)$$

with

$$\alpha = \sqrt{\frac{K_t}{K}} \quad (3)$$

Here, \mathbf{q} , \mathbf{u} , and \mathbf{g} are the collective positions, velocities, and forces, respectively. $1/\boldsymbol{\mu}$ is the diagonal mass matrix having inverse PCA masses $1/\mu_i$ on its diagonal. \mathbf{q} , \mathbf{u} , and \mathbf{g} can be determined from the Cartesian coordinates, velocities, and forces by multiplying them by the matrix of eigenvectors (\mathbf{C}), for example, $\mathbf{q}(t) = \mathbf{C}(\mathbf{x}(t) - \langle \mathbf{x} \rangle)$, with $\mathbf{x}(t)$ as the Cartesian positions at time t and $\langle \mathbf{x} \rangle$ as the average position obtained

from the reference simulation. The velocity rescaling factor α comes from the thermostat of Bussi et al.²¹ It is found by dividing the kinetic energy drawn from a canonical equilibrium kinetic energy distribution K_t by the instantaneous kinetic energy K . Note that overall translation or rotation are absent in the kinetic energy because the corresponding PCs are not updated. This guarantees a canonical distribution with a minimal disturbance of the momentum. μ_i is the mass associated with a certain coordinate q_i and can be chosen freely. Different possibilities will be discussed in the Results and Discussion section. A flowchart of the full algorithm is given in Figure 1. Note that two steps in the MD loop in collective

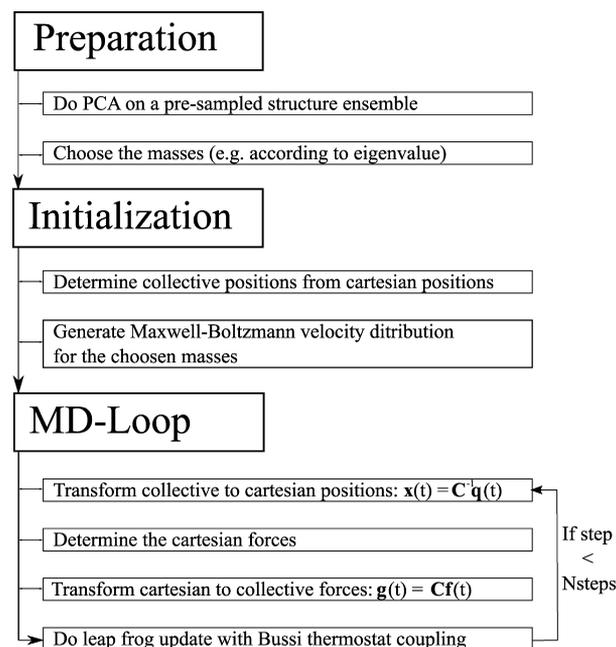


Figure 1. Flowchart of the MD-PCA algorithm.

coordinates are absent in the MD loop in Cartesian coordinates. This is the projection of the coordinates from the collective space to the Cartesian space and the reversed projection of the forces. This is because the force field is expressed in Cartesian coordinates and the forces are determined in Cartesian coordinates. As a result, the code in collective coordinates has two extra matrix vector multiplications compared to the main MD loop in Cartesian coordinates, but this remains a small overhead compared to the determination of the forces. The codes runs approximately 10% slower than the standard MD code for the dialanine system tested in this paper.

Simulation Setup. All simulations were carried out on dialanine with the OPLS-AA force field.²² The system was built using Pymol.²³ For the initial MD simulations to determine the PCA vectors, the standard GROMACS²⁴ simulation package was used. The solvent was modeled using the generalized born implicit solvent model of Still and co-workers²⁵ implemented in GROMACS.²⁶ The system was propagated with a leapfrog integrator with a time step of 1 fs using the thermostat of Bussi et al.²¹ to control the temperature. Overall rotation and translation were constrained with the holonomic rototranslational constraints algorithm of Amadei et al.²⁷ Using rototranslational constraints produces an ensemble with $3N - 6$ degrees of freedom, and because of that, only $3N - 6$ PCs

were used in the MD algorithm in PCA coordinates (MD-PCA). This again results in a simulation without overall rotation and translation. It should be noted that neglecting overall rotation has an influence on the ensemble generated; this effect can be especially pronounced when large conformational transitions occur. Including overall rotation will give an additional stabilization to more extended states of the system. This is however not a fundamental problem because the effect of rotation can always be recovered by a correction proposed by Amadei et al.²⁷

MD-PCA was carried out with a modified version of GROMACS using the same thermostat as the reference simulation but with a time step dependent on the masses chosen such that stable integration is guaranteed. Unless mentioned otherwise, the MD-PCA simulation was carried out for 5×10^8 steps using eigenvectors extracted from a 1 ns MD simulation, masses equal to the inverse square root of the eigenvalue, and a time step of 5 fs.

Statistical Inefficiency. To analyze the simulations, we use a measure called the statistical inefficiency,²⁸ which gives us the variance on the mean taking into account that we have correlated data. If we have uncorrelated data, we can compute the error in the mean of a property \mathcal{A} simply by dividing the variance of \mathcal{A} by the number of data points

$$\sigma^2(\langle \mathcal{A} \rangle_t) = \frac{\sigma^2(\mathcal{A})}{N} \quad (4)$$

with $\langle \mathcal{A} \rangle_t$ the average of \mathcal{A} over a certain time interval t . It is well-known that data points generated by molecular dynamics simulations are correlated. We can take this into account using the statistical inefficiency.²⁸ To compute this, we divide the simulation in n_b blocks of τ_b data points. The mean value of \mathcal{A} for each block ($\langle \mathcal{A} \rangle_b$) can be calculated as

$$\langle \mathcal{A} \rangle_b = \frac{1}{\tau_b} \sum_{\tau=1}^{\tau_b} \mathcal{A}(\tau) \quad (5)$$

and the mean values from all of the blocks can be used to estimate the variance

$$\sigma^2(\langle \mathcal{A} \rangle_b) = \frac{1}{n_b} \sum_{b=1}^{n_b} (\langle \mathcal{A} \rangle_b - \langle \mathcal{A} \rangle_t)^2 \quad (6)$$

For large τ_b , we expect the variance to be inversely proportional to τ_b as the blocks become large enough to be statistically uncorrelated. The aim of the statistical inefficiency method is to estimate the constant of proportionality. We define the statistical inefficiency s as

$$s = \lim_{\tau_b \rightarrow \infty} \frac{\tau_b \sigma^2(\langle \mathcal{A} \rangle_b)}{\sigma^2(\mathcal{A})} \quad (7)$$

We can now find the statistical inefficiency by plotting s as a function of $\tau_b^{1/2}$; the plateau gives the statistical inefficiency. The variance can now be computed as

$$\sigma^2(\langle \mathcal{A} \rangle_t) = \frac{s}{N} \sigma^2(\mathcal{A}) \quad (8)$$

The lower the statistical inefficiency, the more uncorrelated the data points, which is a prerequisite for efficient sampling.

RESULTS AND DISCUSSION

Canonical Sampling. To verify that the MD-PCA algorithm samples the correct ensemble, we compare the free energy along the first eigenvector with a reference MD simulation of 1.5 μ s. The free energy along this coordinate is computed as $-RT \ln(P(x)/P_{\max}(x))$, with $P(x)$ as the probability of state x . The free-energy profile shows two minima corresponding to a closed and an extended conformation, as indicated in Figure 2. The rmsd between

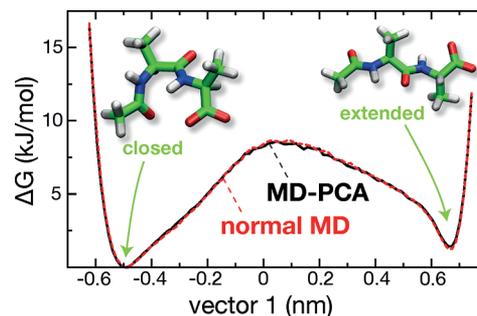


Figure 2. Free energy along eigenvector 1: black for the MD-PCA simulation and red for a normal MD simulation. The structures corresponding to the different minima are indicated.

the two curves shown in the figure is 0.01 kJ mol^{-1} , showing that both methods sample the same ensemble.

Influence of the Choice of the Mass. The statistical inefficiency is used to measure the efficiency of the algorithm. Figure 3 compares the statistical inefficiency analysis of

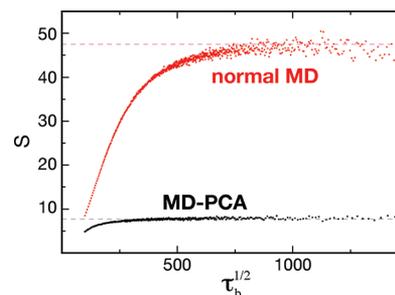


Figure 3. Statistical inefficiency: black for the MD-PCA simulation and red for a normal MD simulation.

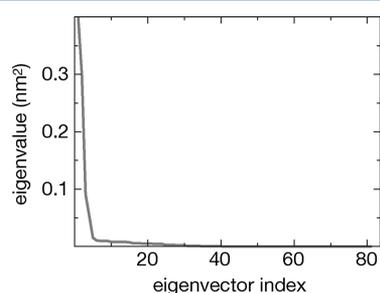
the MD and the MD-PCA simulations. The property that we use here for the statistical inefficiency is the ratio between the extended and closed conformations of dialanine. A structure is assigned to the closed state if the projection on the first eigenvector is smaller than -0.1 nm and to the extended state if the projection is larger than 0.4 nm ; structures in the transition region are not counted. The closed state is given a value of 0, and the extended is given state a value of 1. For the MD simulation, a statistical inefficiency of $s = 48$ was found, and for MD-PCA, it was $s = 8$ (Table 1). This means that to obtain the same variance in the mean ratio, six times fewer data points are needed for MD-PCA compared to a normal MD simulation.

The masses should be chosen such that the motions of the slow modes are amplified while at the same time motions of the fast modes damped. This can be achieved by choosing the masses inversely proportional to the PCA eigenvalues (λ_i). Because the eigenvalues are a measure of the mean-square positional fluctuation along the corresponding eigenvector, a

Table 1. Time Step, Number of Crossings, and Statistical Inefficiency for the Reference Simulation and MD-PCA with Different Lengths of Initial Simulations

simulation	time step (fs)	no. of crossings	statistical inefficiency
MD	1	11844	48
$1/\sqrt{\lambda}$	5	47912	8
$1/\lambda$	8	20261	25
$\mu_1 = 1.0, \text{rest}(1/\sqrt{\lambda})$	5	48325	8
$\mu_1 = 1.0, \mu_2 = 1.0, \text{rest}(1/\sqrt{\lambda})$	5	47861	8
$\mu_1 = 1/\sqrt{\lambda}, \text{rest} = 1000.0$	5	7184	80
$\mu_{1,2} = 1/\sqrt{\lambda}, \text{rest} = 1000.0$	6	13149	50
$\mu_{1-5} = 1/\sqrt{\lambda}, \text{rest} = 1000.0$	5	15092	32
equal masses (10 g/mol)	2	16526	30

logical choice would be to choose the masses μ equal to $1/\sqrt{\lambda}$. In Figure 4, the eigenvalue spectrum of dialanine from a 1 ns

**Figure 4.** Eigenvalue spectrum for dialanine extracted from a 1 ns simulation.

simulations is given. With the choice of $\mu = 1/\sqrt{\lambda}$, this results in a value of approximately 1.5 for μ_1 and 1.9 for μ_2 and goes up to 595 for μ_{76} . In Table 1, different choices of masses are compared with respect to the statistical inefficiency and how often the structure crosses the energy barrier between the two minima. Note that the time step is always chosen to be as large as possible to obtain stable integration. It can be seen in Table 1 that the choice of the mass as $1/\sqrt{\lambda}$ is the optimal choice. By choosing the mass as $1/\lambda$, the time step can be increased to 8 fs, but the overall sampling is not improved; the number of crossings is reduced by more than a factor of 2, and the statistical inefficiency follows the same trend. Because sampling in PCA space is most important for the vectors with lower eigenvalues, the possibility of choosing the mass as $(1/\lambda)^{1/2}$ in the first, first and second, and first through fifth eigenvectors with high masses assigned to the other vectors (1000.0 g/mol) was also considered. It can be seen (Table 2) that choosing the masses according to this scheme does not improve the sampling. The large damping of the other vectors also has an influence on the sampling in the other subspace. In the same spirit, an inverse scheme was tried where a lower mass was chosen for the first vectors, and all of the others were kept at $1/\sqrt{\lambda}$. This as well results in sampling similar to choosing all of the masses equal to $1/\sqrt{\lambda}$.

Influence of the Length of the Initial Simulation. In Table 2, the influence of the length of the initial simulation used to extract the eigenvalues and eigenvectors is shown. The masses are always chosen as $1/(\lambda_i)^{1/2}$. Note that the time step is always chosen to be as large as possible to obtain stable integration. To determine this time step, a MD simulation in a

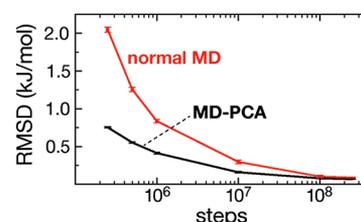
Table 2. Time Step, Number of Crossings, and Statistical Inefficiency for the Reference Simulation and MD-PCA with Different Length of Initial Simulations

simulation	time step (fs)	no. of crossings	no. of crossings in initial simulation	statistical inefficiency
MD	1	11844	/	48
PCA on 1 ps	8	23777	0	17
PCA on 10 ps	7	32795	0	16
PCA on 100 ps	7	31491	3	17
PCA on 1 ns	5	47464	22	8
PCA on 10 ns	5	47912	299	8
PCA on 100 ns	5	48214	2903	8
PCA on 500 ns	5	47758	11844	8

microcanonical ensemble is performed; the largest time step that conserves the total energy is used. It can be seen that already for a short initial simulation of 1 ps, we obtain a considerable improvement over normal MD simulations.

The eigenvalues are not converged here, and especially, the eigenvalues of the first eigenvectors are lower than those in the case of the longer simulations (≥ 1 ns). This results in larger masses and is the reason why a longer time step can be chosen in this case. From an initial simulation length of 1 ns onward, eigenvectors and eigenvalues are reasonably well converged, and a maximal efficiency is reached. This can be explained by the number of crossings in the initial simulation (Table 2). Starting from initial simulation lengths of 1 ns or more, there is already a significant exploration of the two minima with ≥ 22 crossings between them so that no further improvement can be noted for longer initial simulations.

Other Efficiency Measures. In order to compute the statistical inefficiency on a property that is independent of the PCA analysis, it is also computed on the potential energy. The same trend is seen for this measure; $s = 9$ for the MD simulation, and $s = 1.7$ for the MD-PCA. In Figure 5, the

**Figure 5.** Convergence of the free energy along PCA vector 1.

convergence of the free energy to the reference 1.5 μ s simulation is shown for a normal MD simulation and for a MD-PCA simulation. This graph was obtained by dividing the trajectory in blocks and computing for each block the root-mean-square deviation (rmsd) to the reference ensemble, the result in the graph is the average for blocks with the length indicated on the x -axis. To reach a rmsd of 0.4 kJ/mol, 10^6 steps of MD-PCA are needed, and 6×10^6 steps of a normal MD simulation are necessary, indicating a six-fold higher convergence.

The mean-square displacement in the space of the first two eigenvectors is computed (Figure 6). It can be seen that the

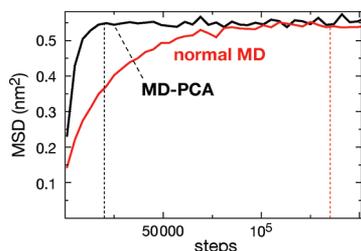


Figure 6. Mean-square displacement: black for the MD-PCA simulation and red for a normal MD simulation.

MD-PCA converges to the maximal value in 20 000 steps, while the MD simulations need 135 000 steps to reach this value. This again points to a 6–7-fold faster exploration of the space of the first two eigenvectors.

In Figure 7, the average step size along the different eigenvectors is shown. It can be seen that with MD-PCA, much

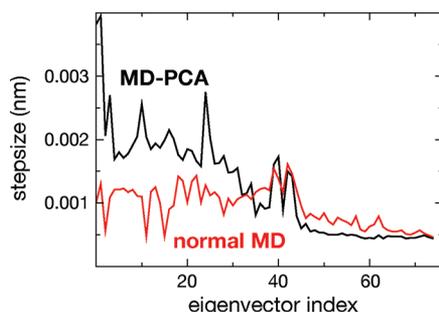


Figure 7. Average step size along the different eigenvectors: black for the MD-PCA simulation and red for a normal MD simulation.

larger steps can be taken along the first eigenvectors, which are the vectors considered most important for the conformational transitions. The difference decreases with the eigenvector index, and for the vectors corresponding to bond and angle vibrations, the MD algorithm takes on average larger steps. Overall, we see that the MD-PCA algorithm ensures a faster exploration of the slow modes and reduces the average step size in the fast modes, resulting in a faster exploration of the conformation space.

CONCLUSION

This work demonstrates how simulation in PC space with properly adjusted masses can enhance sampling of the canonical ensemble with just a small modification of a regular MD code. The improved sampling is demonstrated using several measures, including convergence of the free-energy landscape, statistical inefficiency of the potential energy/extended closed ratio, and mean-square displacement, which all point to a factor of 6 improvement for the dialanine test case. The masses can easily be adjusted rationally by setting them as the inverse square of the eigenvalues. Conveniently, no prior knowledge of the system is needed to set up the simulations; a short initial simulation for the initial PCA suffices.

AUTHOR INFORMATION

Corresponding Author

*E-mail: servaas.michielsens@chem.kuleuven.be.

ACKNOWLEDGMENTS

This work has been supported by the Fund for Scientific Research-Flanders (FWO).

REFERENCES

- (1) Seeliger, D.; Haas, J.; de Groot, B. L. *Structure* **2007**, *15*, 1482–1492.
- (2) Kantarci-Carsibasi, N.; Haliloglu, T.; Doruker, P. *Biophys. J.* **2008**, *95*, 5862–5873.
- (3) Farrell, D. W.; Speranskiy, K.; Thorpe, M. F. *Proteins* **2010**, *78*, 2908–2921.
- (4) Metropolis, N.; Rosenbluth, A. W.; Rosenbluth, M. N.; Teller, A. H.; Teller, E. *J. Chem. Phys.* **1953**, *21*, 1087.
- (5) Sugita, Y.; Okamoto, Y. *Chem. Phys. Lett.* **1999**, *314*, 141–151.
- (6) Moors, S. L. C.; Michielsens, S.; Ceulemans, A. *J. Chem. Theory Comput.* **2011**, *7*, 231–237.
- (7) Mamonov, A. B.; Bhatt, D.; Cashman, D. J.; Ding, Y.; Zuckerman, D. M. *J. Phys. Chem. B* **2009**, *113*, 10891–10904.
- (8) Hamelberg, D.; Mongan, J.; McCammon, J. A. *J. Chem. Phys.* **2004**, *120*, 11919–11929.
- (9) Feenstra, K.; Hess, B.; Berendsen, H. J. *Comput. Chem.* **1999**, *20*, 786–798.
- (10) Gee, P.; van Gunsteren, W. *Chem.—Eur. J.* **2006**, *12*, 72–75.
- (11) Mao, B.; Friedmann, A. *Biophys. J.* **1990**, *58*, 803–805.
- (12) Lin, I.-C.; Tuckerman, M. E. *J. Phys. Chem. B* **2010**, *114*, 15935–15940.
- (13) Bennett, C. J. *Comput. Phys.* **1975**, *19*, 267–279.
- (14) Tsuchida, E. *J. Chem. Phys.* **2011**, *134*, 267–279.
- (15) Mao, B. *Biophys. J.* **1991**, *60*, 611–622.
- (16) Mao, B.; Maggiora, G.; Chou, K. *Biopolymers* **1991**, *31*, 1077–1086.
- (17) Mao, B. *Biophys. J.* **1991**, *60*, 966–973.
- (18) Kubitzki, M. B.; de Groot, B. L. *Biophys. J.* **2007**, *92*, 4262–4270.
- (19) Amadei, A.; Linssen, A. B. M.; de Groot, B. L.; van Aalten, D. M. F.; Berendsen, H. J. C. *J. Biomol. Struct. Dyn.* **1996**, *13*, 615–625.
- (20) Hockney, R.; Goel, S.; Eastwood, J. J. *Comput. Phys.* **1974**, *14*, 148–158.
- (21) Bussi, G.; Zykova-Timan, T.; Parrinello, M. *J. Chem. Phys.* **2009**, *130*, 074101.
- (22) Jorgensen, W. L.; Maxwell, D. S.; TiradoRives, J. *J. Am. Chem. Soc.* **1996**, *118*, 11225–11236.
- (23) DeLano, W. L. *The PyMOL Molecular Graphics System*; DeLano Scientific LLC: Palo Alto, CA, 2008.
- (24) Hess, B.; Kutzner, C.; van der Spoel, D.; Lindahl, E. *J. Chem. Theory Comput.* **2008**, *4*, 435–447.
- (25) Qiu, D.; Shenkin, P. S.; Hollinger, F. P.; Still, W. C. *J. Phys. Chem. A* **1997**, *101*, 3005–3014.
- (26) Larsson, P.; Lindahl, E. *J. Comput. Chem.* **2010**, *31*, 2593–2600.
- (27) Amadei, A.; Chillemi, G.; Ceruso, M.; Grottesi, A.; Di Nola, A. *J. Chem. Phys.* **2000**, *112*, 9–23.
- (28) Allen, M.; Tildesley, D. *Computer Simulations of Liquids*, 1st ed.; Clarendon Press: Oxford, U.K., 1987.