Controlling protein molecular dynamics: How to accelerate folding while preserving the native state

Christian H. Jensen,[a] Dmitry Nerukh,[b] and Robert C. Glen[c]
Department of Chemistry, Unilever Centre for Molecular Science Informatics, University of Cambridge, Cambridge CB2 1EW, United Kingdom

(Received 9 August 2008; accepted 14 October 2008; published online 11 December 2008)

The dynamics of peptides and proteins generated by classical molecular dynamics (MD) is described by using a Markov model. The model is built by clustering the trajectory into conformational states and estimating transition probabilities between the states. Assuming that it is possible to influence the dynamics of the system by varying simulation parameters, we show how to use the Markov model to determine the parameter values that preserve the folded state of the protein and at the same time, reduce the folding time in the simulation. We investigate this by applying the method to two systems. The first system is an imaginary peptide described by given transition probabilities with a total folding time of 1μs. We find that only small changes in the transition probabilities are needed to accelerate (or decelerate) the folding. This implies that folding times for slowly folding peptides and proteins calculated using MD cannot be meaningfully compared to experimental results. The second system is a four residue peptide valine-proline-alanine-leucine in water. We control the dynamics of the transitions by varying the temperature and the atom masses. The simulation results show that it is possible to find the combinations of parameter values that accelerate the dynamics and at the same time preserve the native state of the peptide. A method for accelerating larger systems without performing simulations for the whole folding process is outlined. © *2008 American Institute of Physics*. [DOI: 10.1063/1.3025888]

I. INTRODUCTION

There are many variations of molecular dynamics (MD) which seek to accelerate the folding of peptides and proteins. In the area of biomolecular simulations one of the most widely used methods is replica-exchange MD.[1–4] In this method several MD simulations of the same system are run concurrently at different temperatures. The selection of the temperatures is a poorly understood process. At given times the simulations can exchange temperatures. Low and high temperatures in the simulations allow the system to explore more of the phase space than is achievable (for a similar computational effort) with standard MD. Another method for speeding up the conformational changes is accelerated MD or hyperdynamics.[5–7] In this method an extra term is added to the potential energy at the values below a given threshold. Because this reduces the energy barriers between the states, the system explores the phase space faster. A method designed specifically for accelerating MD of peptide and protein systems uses the construction of a Markov model for conformational transitions.[8,9] The model allows the simulation to be broken into pieces that can be run on independent computers in a way similar to replica-exchange MD. However, unlike replica-exchange, the simulations are all run at the same temperature. This technique has been pioneered, in particular, in the Folding@Home project and taken to the extent where hundred of thousands of computers can participate in a simulation.

All the methods mentioned above are used to accelerate the dynamics of various molecular systems, in particular, protein and peptide simulations. Normally the acceleration comes at a price. In nature, proteins are generally only stable in some temperature interval, above and below this they unfold. Therefore simply raising the temperature may speed up the transitions between some conformational states, but it may also make some states inaccessible, including the native state. A similar problem can be expected when the model parameters, the force field, are changed. In the case of replica-exchange MD the problem is resolved by letting the temperature change, but the interval in which to change the temperature is not always clear. In the present paper we use a Markov model to describe protein and peptide folding. In our case the model is constructed from MD data in a way similar to that employed by the Folding@Home group. However, our goal is very different. We aim to investigate how the folding dynamics of a peptide should be changed in order to reduce the folding time. In order to change the dynamics we assume that the transition probabilities between the conformational states can be changed by varying parameters in the MD simulation such as the temperature or the force field. We impose two requirements on the changes of the Markov model: (1) the folded conformational state must stay the same and (2) the change must accelerate folding. This methodology is applied to two systems. First, a hypothetical sys-

[a]Electronic mail: chj22@cam.ac.uk.
[b]Electronic mail: dn232@cam.ac.uk.
[c]Electronic mail: rcg28@cam.ac.uk.

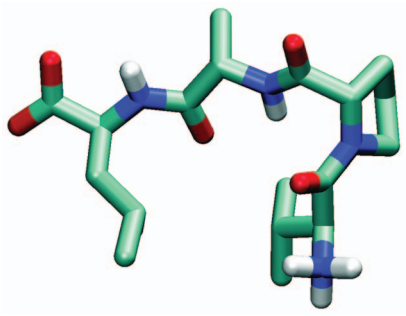

FIG. 1. (Color) VPAL.

tem described by given transition probabilities between the states with a total folding time of 1 μs is investigated. We show that by making very small changes in the probabilities it is possible to reduce the folding time of the system by a few orders of magnitude. Because these changes are small it is possible to alter parameters in the MD simulations to accommodate this. This also implies that the folding time of slow folding peptides and proteins cannot be reliably compared to folding times obtained experimentally. Second, we apply the approach to MD simulations of a four reside peptide valine-proline-alanine-leucine (VPAL), Fig. 1. The goal is to find how to vary parameters of the simulation in order to change the Markov model so as to both preserve the folded state and reduce the folding time. The parameters that we have chosen to vary are the temperature and the masses of the system atoms (equivalent to changing the force field).

II. THEORY

In our investigation configurational states are defined by clustering the simulated trajectory. This is done by analyzing the Ramachandran plots of the residues of the peptide. Each Ramachandran plot is clustered independently and the molecule's configurations are found as a combination of the cluster indices from all the plots. The Markov model is described by a state vector v that holds probabilities of the configurations and a transition matrix T. Examples of state vectors could be (1, 0, 0, 0) or (0.5, 0.5, 0, 0). Here the system has a total of four possible states obtained from two clusters on two Ramachandran plots. In the first case the system is with 100% probability in state 1. In the second case the system is in state 1 with 50% probability and in state 2 with 50% probability. Note that the total probability of the state vector has to sum to 100% since the peptide has to be in some configuration. The transition matrix simply holds the probability that the system is transferred from one state to another at the next time step. $T_{11}=0.5$ would mean that there is a 50% probability of the system remaining in the same state. $T_{21}=0.25$ means that there is a 25% probability of the system changing from state 1 to 2. Because the total probability of the state vector has to be conserved the requirement $\Sigma_i T_{ij}=1$ is imposed, where i and j run over all states. Given that the system has state vector v_t at time t, the state vector at time $t+\Delta t$ can be calculated as $v_{t+\Delta t}=Tv_t$. Whether the dynamics of the system can actually be described by a Markov model can be determined by investigating how the eigenvalues of the transition matrix vary with time step Δt.[10,11]

Assuming the dynamics of protein folding are described completely by the Markov model with transition matrix T, we can use the model to investigate how to accelerate the dynamics most efficiently. Because T is a transition matrix it has eigenvalues in the range of 0–1 with one eigenvalue being 1. In the following it is assumed that the eigenvalues are ordered in descending order so that λ_0 corresponds to the eigenvalue 1. The time evolution of the system is then given by

$$v_{t+n\Delta t}=T^n v_t=\sum_i \lambda_i^n |\lambda_i\rangle\langle\lambda_i| v_t. \tag{1}$$

At the limit $n\to\infty$ only the largest eigenvalue, equal to 1, survives while all other eigenvalues, being less than 1, tend to zero. Therefore, the eigenvector $|\lambda_0\rangle$ corresponds to the equilibrium distribution of states. The speed at which the system approaches the equilibrium distribution is described by all the other eigenvalues that are less than 1. To speed up the dynamics we must therefore reduce these eigenvalues, in particular, the second largest eigenvalue since it describes the slowest convergence in the system. As mentioned in Sec. I there are many methods that seek to accelerate the dynamics. However, the common problem is that apart from accelerating the convergence they also generally change the equilibrium distribution of states. Therefore, for a correct acceleration we impose *two* requirements: (1) the equilibrium distribution of states must be the same as for the original system and (2) the method must reduce the folding time. These two requirements can be written as

$$\langle\lambda_0|\Delta T|\lambda_0\rangle=0 \tag{2}$$

and

$$\langle\lambda_1'|T'|\lambda_1'\rangle<\langle\lambda_1|T|\lambda_1\rangle. \tag{3}$$

The prime ($'$) marks the changed system and $\Delta T=T'-T$. To obtain a change in the transition matrix we vary parameters of the molecular model $\alpha,\beta,\gamma,\ldots$. In the case when we have small changes in the transition matrix we can assume that

$$\Delta T(\alpha,\beta,\gamma,\ldots)\approx\Delta T(\alpha)+\Delta T(\beta)+\Delta T(\gamma)+\cdots. \tag{4}$$

This is a very useful approximation since it allows varying each parameter, in turn, when we investigate how T changes with the parameters. Using first order perturbation theory we obtain (see Appendix)

$$\langle\lambda|\Delta T|\lambda\rangle\approx\delta\lambda, \tag{5}$$

where $\lambda'=\lambda+\delta\lambda$. Therefore, assuming that the changes in the transition matrix are small, the conditions given by Eqs. (2) and (3) can be written as

$$\langle\lambda_0|\Delta T(\alpha)|\lambda_0\rangle+\langle\lambda_0|\Delta T(\beta)|\lambda_0\rangle+\langle\lambda_0|\Delta T(\gamma)|\lambda_0\rangle+\cdots=0 \tag{6}$$

and

$$\langle\lambda_1|T(\alpha)|\lambda_1\rangle+\langle\lambda_1|\Delta T(\beta)|\lambda_1\rangle+\langle\lambda_1|\Delta T(\gamma)|\lambda_1\rangle+\cdots<0. \tag{7}$$

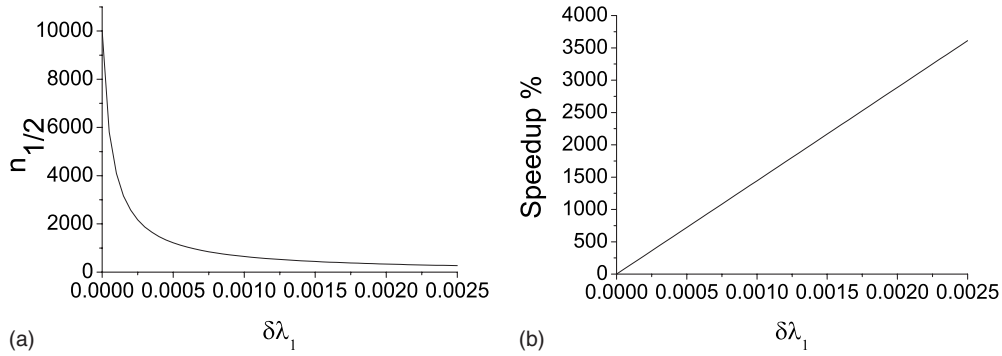

FIG. 2. (a) The folding halftime of the eigenvalue vs the change in eigenvalue. (b) The percentage speedup vs the change in the eigenvalue.

III. APPLICATION I

For a small peptide of realistic size the folding time can be expected to be, for example, 1μs. If the transition matrix is constructed with a time step of 100 ps, the number of steps required for folding is $n=1\mu\text{s}/100\text{ps}=10\,000$. Let us assume that this time is equivalent to the time it takes to reduce the part of the initial state spanned by $|\lambda_1\rangle$ to half of its value, i.e., the halftime of the eigenstate $|\lambda_1\rangle$, Eq. (1): $\lambda_1^n=\frac{1}{2}$.Then, we designate the folding time as a "folding half time" that can be calculated as

$$n_{1/2}=-\frac{\ln 2}{\ln \lambda_1}. \tag{8}$$

By rearranging the eigenvalue itself can be found

$$\lambda_1=\left(\tfrac{1}{2}\right)^{1/(n_{1/2})}. \tag{9}$$

Suppose that we have changed the dynamics and, as a result, the eigenvalue λ_1 has changed by an amount $\delta\lambda_1$. The halftime for this new eigenvalue is

$$n'_{1/2}=-\frac{\ln 2}{\ln(\lambda_1-\delta\lambda_1)}. \tag{10}$$

For the folding time of 10 000 steps the corresponding eigenvalue is $\lambda_1=0.999\,930\,688$, Eq. (9). The speedup with different values of $\delta\lambda_1$ as found using Eq. (10) is shown in Fig. 2. From Eq. (10) it is clear that the longer the folding time of the peptide or protein, the larger the speedup for a given change $\delta\lambda_1$. Therefore, the speedup will be most significant for proteins which fold slowly.

These considerations also have important consequences for the accuracy of the folding time obtained in MD simulations. It is clear from the above that proteins with long folding times are very sensitive to the changes in the transition matrix and, in turn, the force field. Since any force field is only approximately correct, this means that calculated folding times are significantly inaccurate, even though the folded state reached in the simulation is correct. It is therefore not meaningful to make a comparison between a simulated folding time and that determined experimentally, especially for slowly folding proteins.

Finally it should be noted that the results rely on the Markovian behavior of the system on the 100 ps time scale. We have shown in our previous work that this is indeed the case for a small peptide.[11] Work is under way to elucidate the time scale at which larger peptide and protein systems behave Markovian.

IV. APPLICATION II

A. Method

We have investigated how to accelerate a MD simulation of a four residue peptide VPAL at the temperature of 300 K. To do this we calculated the transition matrix from the simulation trajectories. We then wished to find out how to change the transition matrix to accelerate the dynamics. The change in the transition matrix is accomplished by changing only two parameters of the molecular model: the temperature and the masses of the atoms.

We first investigate how the transition matrix changes with the parameters varying in an interval around their original values. For the simulations with different parameter values we construct transition matrices and hence find how the transition matrix varies with the parameters.

All the simulations were performed using the software package GROMACS 3.2.[12] The peptide, Fig. 1, was solvated in 874 SPC water molecules. The force field 53a6 (Refs. 13–15) optimized for bimolecular systems interacting with water was used. Periodic boundary conditions with a box of size $3.0\times3.0\times3.0$ Å^3 were used. The temperature was kept constant using the Berendsen thermostat.[16] The atomic positions were recorded every 0.5 ps. The integration algorithm was a Verlet type and the integration step was 0.002 ps. The system was equilibrated before it was sampled for 200 ns.

To find the likely effect of varying the parameters we assume that the transition state theory is valid [i.e., the transition rate is proportional to $\exp(-\Delta E/k_BT)$]. It is then clear that an increase in temperature will lead to an increase in transition rate and vice versa (this is essentially what is exploited in replica-exchange MD). To find the effect of varying the masses we need to look at Newton's second law. The variation in the masses is described by the introduction of the unified parameter α so that the new masses are αm,

$$(\alpha m)a=-\frac{\delta V}{\delta r}, \tag{11}$$

or

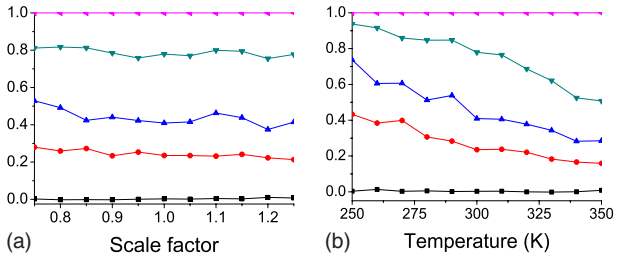

FIG. 3. (Color online) The variation in eigenvalues with varying (a) scaling and (b) temperature.

$$ma = -\frac{\delta(V/\alpha)}{\delta r}. \tag{12}$$

From this it can be seen that varying the masses by a factor of α is equivalent to varying the potential energy by a factor of $1/\alpha$, which is changing the force field of the model. Therefore from transition state theory, an increase in the masses will increase the transition rate and vice versa, in other words, the change in temperature with the factor α should be equivalent to a change in the masses by the same factor. However, the effect of varying the masses is expected to be less. This is because we use a thermostat in the simulations. To keep the temperature constant the velocities of the atoms are adjusted to compensate for the changes in the masses.

B. Results

In our investigation we have varied the temperature from 250 to 350 K in steps of 10 K while keeping the scaling constant α at 1.0. We have also varied the scaling α from 0.75 to 1.25 in steps of 0.05 while keeping the temperature constant at 300 K. We then calculated the transition matrices for each of these simulations, as described in Sec. II. There are a total of five conformational states (see Ref. 11 for details). The transition matrix from the simulation at 300 K and scaling at 1.0 is as follows:

$$T = \begin{pmatrix} 0.7729 & 0.5324 & 0.2807 & 0.1238 & 0.1041 \\ 0.1950 & 0.4472 & 0.1856 & 0.0221 & 0.1952 \\ 0.0004 & 0.0007 & 0.0054 & 0.0033 & 0.0061 \\ 0.0290 & 0.0041 & 0.4142 & 0.7220 & 0.2153 \\ 0.0028 & 0.0156 & 0.1172 & 0.1288 & 0.4793 \end{pmatrix}. \tag{13}$$

In Fig. 3 the eigenvalues are plotted for the variation in scaling (a) and temperature (b). It can be seen that an increase in temperature increases the speed of conformational transitions between the states. This is because the eigenvalues decrease as the temperature is raised. However, the effect is barely visible in the case of the change in scaling. Overall the changes are as predicted in Sec. IV A.

When considering reasonable boundaries for variations in the temperature and scaling, the changes in the transition matrix are small. The conditions that the changes must satisfy to accelerate the folding and preserve the native state are given by Eqs. (6) and (7). To find which parameter sets satisfy these conditions, we calculate the different elements in the equations. In Fig. 4 the elements $\langle\lambda_0|\Delta T|\lambda_0\rangle$ and $\langle\lambda_1|\Delta T|\lambda_1\rangle$ are shown for the variation in temperature. $\langle\lambda_0|\Delta T|\lambda_0\rangle$ is a measure of how the equilibrium distribution of states is changed by a given change in temperature. Similarly, $\langle\lambda_1|\Delta T|\lambda_1\rangle$ is a measure of the acceleration achieved (the change in the second largest eigenvalue) for a given change in temperature. In Fig. 5 the same quantities are plotted for the variation in scaling.

The parameter values that satisfy condition (6) can be found simply by overlapping the $\langle\lambda_0|\Delta T|\lambda_0\rangle$ graphs of Figs. 4 and 5. From these we then take the combinations which also satisfy condition (7) by overlapping the $\langle\lambda_1|\Delta T|\lambda_1\rangle$ graphs of Figs. 4 and 5. The parameter set that gives the best acceleration will also give the most negative value on the left hand side in Eq. (7) and can, therefore, be easily identified.

The above procedure would be instructive in finding the parameters that speed up the folding of the peptide system. Unfortunately our simulations to date do not produce good-enough statistics, longer trajectories are required and these are currently being generated. However, by looking at the range of values it is clear that there is a parameter set which satisfies the conditions and thus leaves the folded state unchanged but accelerates the overall dynamics.

V. OUTLOOK

The method requires the complete knowledge of the dynamics (converged transition matrix) for finding the optimal acceleration parameters. This presents a problem for a larger protein since the conformational space is computationally impossible to sample exhaustively. A possible solution could be to investigate how to accelerate parts of the protein, and

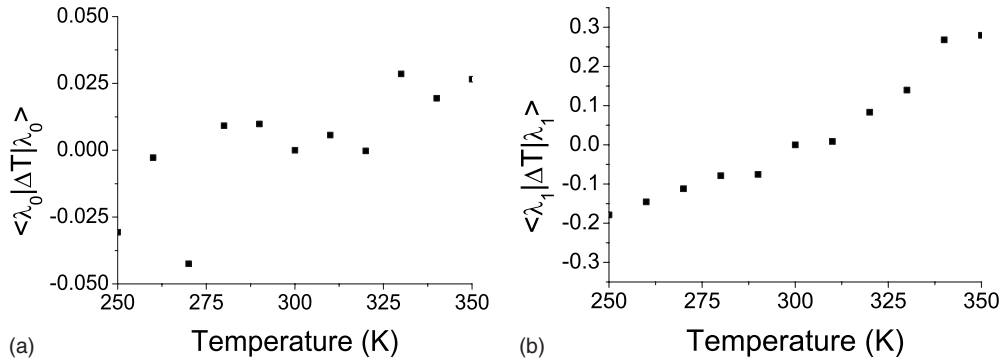

FIG. 4. The variation in $\langle\lambda_0|\Delta T|\lambda_0\rangle$ and $\langle\lambda_1|\Delta T|\lambda_1\rangle$ with varying temperature. The former is a measure of the change in the equilibrium distribution of states while the latter is a measure of the acceleration.

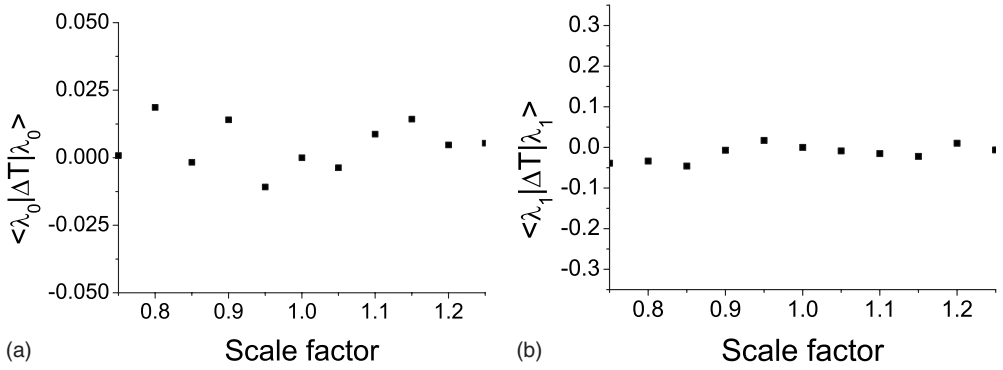

FIG. 5. The variation in $\langle\lambda_0|\Delta T|\lambda_0\rangle$ and $\langle\lambda_1|\Delta T|\lambda_1\rangle$ with varying scaling. The former is a measure of the change in the equilibrium distribution of states while the latter is a measure of the acceleration.

then incorporate this into the simulation. This would work if the conformations adopted by the separate parts are similar to those of the whole protein. It is clear that the larger the parts, the more similar the conformations are likely to be. Whether this is possible is the subject of our current work. However, with good control over the force field, the idea looks promising.

VI. CONCLUSIONS

We introduced a method for accelerating the dynamics of a peptide or protein in a way which does not change the equilibrium distribution of states (the folded state). The method was applied to two systems. The first system was an imaginary peptide with a folding time of $1\mu s$. We found that the folding time could be reduced significantly by making very small changes in the transition matrix. In general it was found that peptides and proteins that fold slowly are most sensitive to acceleration. This result also implies that folding times of peptides and proteins calculated using MD simulations cannot be meaningfully compared to experimental results, especially for slowly folding molecules.

We also tested the acceleration method on a four residue peptide VPAL. The parameters we varied in order to alter the transition matrix T were the temperature and the atom masses. From the simulations we have obtained the expected behavior for the temperature variation. By varying the masses, however, there was almost no change in T. This was attributed to the effect of the thermostat. For both cases we expect more pronounced effect for longer simulation times. Nevertheless, the results demonstrate the possibility of finding the combinations of the temperature and masses that accelerate the dynamics and at the same time, preserve the native state.

ACKNOWLEDGMENTS

The work was supported by Unilever and the European Commission (EC Contract No. 012835-EMBIO).

APPENDIX: APPROXIMATION OF λ'

In the following we find an approximation for λ' using first order perturbation theory. We have that

$$T|\lambda\rangle = \lambda|\lambda\rangle. \tag{A1}$$

For the changed dynamics of the system the new transition matrix T' has corresponding eigenvalues and eigenvectors,

$$T'|\lambda'\rangle = \lambda'|\lambda'\rangle. \tag{A2}$$

Assuming that the changes are small we write this as the original equation with small perturbations added,

$$(T+\delta T)(|\lambda\rangle + |\delta\lambda\rangle) = (\lambda+\delta\lambda)(|\lambda\rangle + |\delta\lambda\rangle). \tag{A3}$$

Expanding on both sides gives

$$T|\lambda\rangle + \delta T|\lambda\rangle + T|\delta\lambda\rangle + \delta T|\delta\lambda\rangle = \lambda|\lambda\rangle + \delta\lambda|\lambda\rangle + \lambda|\delta\lambda\rangle + \delta\lambda|\delta\lambda\rangle. \tag{A4}$$

The first two terms on either side are equal and therefore cancel each other. The terms that are of the second order in the change can be neglected because we have assumed small changes. By rearranging the remaining terms we obtain

$$\delta T|\lambda\rangle \approx \delta\lambda|\lambda\rangle + \lambda|\delta\lambda\rangle - T|\delta\lambda\rangle. \tag{A5}$$

Now $\langle\lambda|T'|\lambda\rangle$ can be approximated as

$$\langle\lambda|T'|\lambda\rangle = \langle\lambda|T+\delta T|\lambda\rangle = \langle\lambda|T|\lambda\rangle + \langle\lambda|\delta T|\lambda\rangle, \tag{A6}$$

$$\approx \lambda + \langle\lambda|\delta\lambda|\lambda\rangle + \langle\lambda|\lambda|\delta\lambda\rangle - \langle\lambda|T|\delta\lambda\rangle, \tag{A7}$$

$$= \lambda + \delta\lambda + \lambda\langle\lambda|\delta\lambda\rangle - \lambda\langle\lambda|\delta\lambda\rangle, \tag{A8}$$

$$= \lambda + \delta\lambda = \lambda', \tag{A9}$$

where Eq. (A7) follows from Eq. (A6) by substituting Eq. (A5). Therefore, for small changes

$$\langle\lambda|T'|\lambda\rangle \approx \lambda'. \tag{A10}$$

This is a very useful result because it provides an easy way of calculating new eigenvalues when the matrix is changed from T to T'.

[1] Y. Sugita and Y. Okamoto, Chem. Phys. Lett. **314**, 141 (1999).
[2] X. Periole and A. E. Mark, J. Chem. Phys. **126**, 014903 (2007).
[3] A. Baumketner and J. E. Shea, Theor. Chem. Acc. **116**, 262 (2006).
[4] K. P. Ravindranathan, E. Gallicchio, R. A. Friesner, A. E. McDermott, and R. M. Levy, J. Am. Chem. Soc. **128**, 5786 (2006).
[5] D. Hamelberg, J. Mongan, and J. A. McCammon, J. Chem. Phys. **120**, 11919 (2004).
[6] A. F. Voter, Phys. Rev. Lett. **78**, 3908 (1997).

[7] A. F. Voter, J. Chem. Phys. **106**, 4665 (1997).
[8] N. Singhal, C. D. Snow, and V. S. Pande, J. Chem. Phys. **121**, 415 (2004).
[9] G. Jayachandran, V. Vishal, and V. S. Pande, J. Chem. Phys. **124**, 164902 (2006).
[10] W. C. Swope, J. W. Pitera, and F. Suits, J. Phys. Chem. B **108**, 6571 (2004).
[11] C. H. Jensen, D. Nerukh, and R. C. Glen, J. Chem. Phys. **128**, 115107 (2008).
[12] D. Van der Spoel, E. Lindahl, B. Hess, G. Groenhof, A. E. Mark, and H. J. C. Berendsen, J. Comput. Chem. **26**, 1701 (2005).
[13] B. Hess and N. F. A. van der Vegt, J. Phys. Chem. B **110**, 17616 (2006).
[14] C. Oostenbrink, T. A. Soares, N. F. A. van der Vegt, and W. F. van Gunsteren, Eur. Biophys. J. **34**, 273 (2005).
[15] C. Oostenbrink, A. Villa, A. E. Mark, and W. F. Van Gunsteren, J. Comput. Chem. **25**, 1656 (2004).
[16] H. J. C. Berendsen, J. P. M. Postma, W. F. van Gunsteren, A. DiNola, and J. R. Haak, J. Chem. Phys. **81**, 3684 (1984).