JCTC Journal of Chemical Theory and Computation

Article

pubs.acs.org/JCTC

Improvements in Markov State Model Construction Reveal Many Non-Native Interactions in the Folding of NTL9

Christian R. Schwantes[†] and Vijay S. Pande*[,†,‡,§]

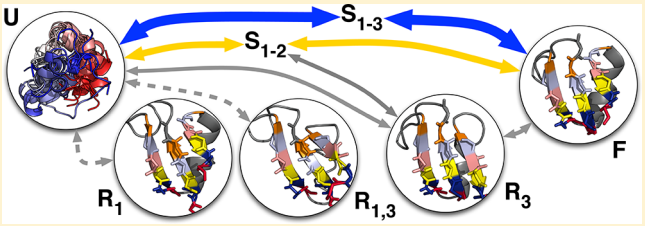
[†]Department of Chemistry, Stanford University, Stanford, California, United States

[‡]Biophysics Program, Stanford University, Stanford, California, United States

[§]Department of Computer Science, Stanford University, Stanford, California, United States

Supporting Information

ABSTRACT: Markov State Models (MSMs) provide an automated framework to investigate the dynamical properties of high-dimensional molecular simulations. These models can provide a human-comprehensible picture of the underlying process and have been successfully used to study protein folding, protein aggregation, protein ligand binding, and other biophysical systems. The MSM requires the construction of a discrete state-space such that two points are in the same state if they can interconvert rapidly. In the following, we suggest an improved method, which utilizes second order Independent Component Analysis (also known as time-structure based Independent Component Analysis, or tICA), to construct the state-space. We apply this method to simulations of NTL9 (provided by Lindorff-Larsen et al. *Science* **2011**, *334*, 517−520) and show that the MSM is an improvement over previously built models using conventional distance metrics. Additionally, the resulting model provides insight into the role of non-native contacts by revealing many slow time scales associated with compact, non-native states.



INTRODUCTION

Molecular simulation can provide atomic-level insight into problems in protein folding,[1−3] protein aggregation,[4] protein−ligand binding,[5] and many other biophysical systems of interest. Recent technological advances, in the form of distributed computing (Folding@home)[6] and specialized hardware (Anton),[7] have allowed for simulations to study folding events on the timecales of 10 ms and 100 μs, respectively. The result of such a large-scale simulation, however, is a high-dimensional time series that is difficult to understand. Constructing a human-comprehensible picture of such a data set is nontrivial, and analysis generally benefits from dimensionality reduction to focus on the important features in the data.

Many techniques have been used in the past for removing unimportant degrees of freedom. For example, in the protein folding field, many choose to project the high-dimensional data onto a few order parameters and analyze the data in this lower-dimensional space.[8−10] These analyses can be effective but come with the underlying assumption that all other degrees of freedom are irrelevant. If this is not the case, then the results will be incorrect, for instance by placing two pieces of data at the same point of the projection when they are actually separated by a kinetic barrier.[11,12] A more robust approach to this type of scheme would be to select projections that best represent the data in an algorithmic manner, as this would eliminate most human biasing.

Principal Component Analysis (PCA) is an automated method that attempts to find the linear combinations of input coordinates that best explain the variance in the data.[13] Using PCA, one can find orthogonal degrees of freedom that account for the highest amount of variance and then remove degrees of freedom that do not account for much variance in the data. This strategy can be useful in protein folding, for example, since the folding process is a large-scale motion.[14−16] However, to accurately estimate kinetic properties of the system using PCA, we must assume that the kinetically slow directions correspond to high variance directions. This need not be the case, and so PCA may not be effective in analyzing subtler motions.

There have also been other automated dimensionality reduction methods applied to protein folding, such as ISOMAP[17] (and its relative SciMAP[18]), Diffusion Maps,[19] and Sketch-Map,[20] that attempt to construct a lower-dimensional, Euclidean space for the data. These methods all require a distance metric that is calculated in the high-dimensional space. In the limit of infinite data, these methods only require the distance metric to determine "kinetic-relatedness" (i.e., whether two conformations can interconvert rapidly) at short distances. Because of this, most structural metrics will perform quite well, and the result will be a reduced representation that preserves the kinetics in the high-dimensional space. For most problems, however, we are not in this data-rich regime; in fact, we are typically in the data-poor regime. As a result, the quality of the lower-dimensional representation will depend on the distance metric's ability to determine kinetic-relatedness between any

Received: October 10, 2012

   dx.doi.org/10.1021/ct300878a | *J. Chem. Theory Comput.* XXXX, XXX, XXX−XXX

two conformations that may not be structurally close. This dependence does not mean the methods are unable to build low-dimensional representations of the high-dimensional space, but the methods are limited by the distance metric they employ. Additionally, these methods tend to be computationally difficult and so do not lend themselves to large data sets, such as those we frequently encounter in the protein folding community.

Another class of techniques, such as the Markov State Model (MSM), uses dimensionality reduction in the form of clustering methods. These methods transform the high-dimensional space into a discrete set of states by grouping points that are close in the high-dimensional space. An MSM uses this state decomposition and calculates rates of interconversion between all states.[21−26] The quality of an MSM is dependent on the state decomposition, so it is very important to construct a state space that accurately captures the kinetics of the underlying system. To do this, however, we require that each state only contains conformations that can interconvert rapidly.

Recent work has improved the MSM building process in many ways, such as using a milestoning approach[25] or a maximum likelihood estimator for the transition matrix.[27] In addition, there have been recent improvements in the process of defining a state space; one such improvement is the utilization of better clustering algorithms. For example, the K-Centers algorithm has worked well in the past, but more advanced methods like hybrid K-Medoids or Ward's method have shown a marked improvement.[27−31] However, each clustering method requires a distance metric that can estimate wether two conformations can interconvert rapidly. Historically, RMSD in the atomic positions has been used with some success in addition to new metrics based on alternative representations of the protein conformation[22,32,33] or metrics that use a dimensionality reduction technique like PCA.[34]

Since most metrics defined on protein conformations are merely assumed to estimate the interconversion time between them, we believed we could improve the clustering scheme if we designed a metric with this assumption in mind. Moreover, the problem with using a geometric distance for clustering is that we may be ignoring important degrees of freedom that do not decorrelate quickly. With this motivation, we believed the best distance metric would be one that is designed so that orthogonal degrees of freedom decorrelate quickly.

In the following, we utilize a projection-based distance metric motivated by kinetics, second order independent component analysis.[35−37] Briefly, this method specifically picks degrees of freedom that decorrelate slowly. In this manner, we can remove degrees of freedom that actually do decorrelate quickly. We then project the high-dimensional data onto these slow degrees of freedom and cluster the data in this lower-dimensional space. We show that applying this method to simulations of NTL9, generously shared by Lindorff-Larsen et al.,[38] constructed an improved MSM as well as provided additional insight into the folding process of NTL9.

METHODS

Second Order Independent Component Analysis. Our goal is to find linear combinations of coordinates in the input data such that we can project onto only a few of them without losing important kinetic information. Since other projection methods simply assume that orthogonal degrees of freedom are kinetically irrelevant, we would like to design projections so that this is the case.

The method we have used was first introduced as a solution to the blind source separation problem, which attempts to find a set of independent source signals that can explain a multidimensional data set.[35] The details of the original problem can be found in Molgedey and Schuster[35] as well as more recent reviews,[36] but we will not discuss them here. This analysis has been successfully been applied in many fields and has many titles (e.g., time-structure based Independent Component Analysis [tICA][37] and second order independent component analysis[35,36]); we refer to the method as tICA.

The problem is generally stated as finding maximally independent components in the data. However, we outline one possible proof below (the details can be found in the SI in a section entitled "Detailed Proof of the Solutions to the tICA Problem") that shows that the solution of the tICA problem also corresponds to finding the slowest degrees of freedom, as monitored by their autocorrelation functions. We note that this proof is analogous to a classical proof of the solution to the PCA problem (as given by Jolliffe[13]). To the authors' knowledge, this formulation has not been produced before, but we note that the result is not new.[36] We believe that the proof aids in understanding the motivation of the method as well as the properties of the solutions.

Given a multidimensional data set, our goal is to find the linear combinations of the input coordinates that maximize the autocorrelation function of that projection, while constraining each linear combination to be uncorrelated to the previous ones. We do this in a series of maximizations, at each step finding a new tICA component (tIC) that is the slowest subject to being uncorrelated to all the previously found tIC's.

Let $\{\mathbf{X}_t\}_{t=0}^{N_f-1}$ be a multidimensional, discrete time-series where each snapshot is a column vector, of dimension d, corresponding to an arbitrary, vectorized representation of a protein conformation. For example, each snapshot could be a set of dihedral angles of a peptide. (Though to account for periodicity, one must represent each angle, ϕ, as two coordinates: cos ϕ and sin ϕ.[39]) In the above notation, $\mathbf{X}_t$ is a snapshot of the protein at time t, and N_f is the total number of frames in the data set. We note also that for notational convenience we will think of the data set as a single, very long trajectory. However, the results are easily generalized to multiple trajectories by calculating the average correlations over all snapshots and all trajectories.

For convenience, we will use bra-ket notation to denote inner and outer products. Therefore we write $\mathbf{X}_t$ as $|\mathbf{X}_t\rangle$. As with conventional PCA, we need to remove the mean of our data, and so we do all calculations with $|\delta\mathbf{X}_t\rangle = |\mathbf{X}_t\rangle - \mathbb{E}[|\mathbf{X}_t\rangle]$. Before we begin, recall that the definition of the autocorrelation function of a one-dimensional time-series (e.g., r_t) is given by

$$\mathrm{auto}(r_t;\,\Delta t) = \frac{\mathbb{E}[(\delta r_t)(\delta r_{t+\Delta t})]}{\mathbb{E}[(\delta r_t)(\delta r_t)]} \tag{1}$$

The autocorrelation is a function of the parameter Δt, which we refer to as the correlation lag time. Slower degrees of freedom are those whose autocorrelation functions approach zero slowly. This means that given some Δt, the slowest degrees of freedom will have the largest autocorrelation value. Specifically, if we pick a relevant correlation lag time (we will discuss how to select this parameter in the Results and Discussion section), then the goal is to find a projection vector, $|\alpha_0\rangle$, to maximize the following objective function (f):

$$f(|\alpha_0\rangle) = \frac{\mathbb{E}[\langle\alpha_0|\delta\mathbf{X}_t\rangle\langle\alpha_0|\delta\mathbf{X}_{t+\Delta t}\rangle]}{\mathbb{E}[\langle\alpha_0|\delta\mathbf{X}_t\rangle\langle\alpha_0|\delta\mathbf{X}_t\rangle]} \quad (2)$$

This objective function is the autocorrelation function of the projection of $|\delta\mathbf{X}_t\rangle$ onto $|\alpha_0\rangle$, which is simply the inner product: $\langle\alpha_0|\delta\mathbf{X}_t\rangle$. Since the inner product is symmetric, we can simplify this expression in terms of outer products:

$$f(|\alpha_0\rangle) = \frac{\mathbb{E}[\langle\alpha_0|\delta\mathbf{X}_t\rangle\langle\delta\mathbf{X}_{t+\Delta t}|\alpha_0\rangle]}{\mathbb{E}[\langle\alpha_0|\delta\mathbf{X}_t\rangle\langle\delta\mathbf{X}_t|\alpha_0\rangle]} \quad (3)$$

We can now see that the outer products are the same as the time-lag correlation matrix and covariance matrices:

$$\mathbf{C}^{(\Delta t)} = \mathbb{E}[|\delta\mathbf{X}_t\rangle\langle\delta\mathbf{X}_{t+\Delta t}|] \quad (4)$$

$$\Sigma = \mathbb{E}[|\delta\mathbf{X}_t\rangle\langle\delta\mathbf{X}_t|] \quad (5)$$

We note that as with any statistical technique applied to real data, we are limited by the number of samples we have in the trajectory. In particular, because of the correlation lag time, there are only $N_f - \Delta t$ samples for the time-lag correlation matrix, whereas there are N_f samples for the covariance matrix. If Δt is small relative to N_f, then the quality of the sample time-lag correlation matrix will be similar to the quality of the covariance matrix.

Using the above notation, we can rewrite the objective function from eq 2 as

$$f(|\alpha_0\rangle) = \frac{\langle\alpha_0|\mathbf{C}^{(\Delta t)}|\alpha_0\rangle}{\langle\alpha_0|\Sigma|\alpha_0\rangle} \quad (6)$$

We now need to constrain our optimization so that we can find solutions $|\alpha_0\rangle$. In conventional PCA, the PCs are constrained to have unit length. This would look like $\langle\alpha_0|\alpha_0\rangle = 1$ in our tICA problem. We instead constrain our tIC's to have unit variance: $\langle\alpha_0|\Sigma|\alpha_0\rangle = 1$. Since our goal is to calculate distances along these projections, we wish to make them unitless. If we did not do this, then projections that happen to have high variance would dominate the distance calculation, when we actually want slow projections to dominate. It is possible, however, that different normalizations may improve the distance calculation further, and this will be a topic of future research. (For example, one might consider weighting the solutions by their "slowness" so that the slowest degrees of freedom contribute the most to the distance calculation.)

Conveniently, this constraint allows us to simplify the objective function and construct an optimization problem that is analogous to the PCA problem:

$$\max_{|\alpha_0\rangle} f(|\alpha_0\rangle) = \max_{|\alpha_0\rangle}\langle\alpha_0|\mathbf{C}^{(\Delta t)}|\alpha_0\rangle$$
$$\text{subject to:}$$
$$\langle\alpha_0|\Sigma|\alpha_0\rangle = 1 \quad (7)$$

From here, we note that there are many ways to solve this problem. For instance, the procedure would be analogous to a proof for the variational principle in quantum mechanics, which attempts to find the eigenfunction, $|\psi\rangle$, that minimizes its energy:

$$E(|\psi\rangle) = \frac{\langle\psi|H|\psi\rangle}{\langle\psi|\psi\rangle}$$

Additionally, we can make an argument analogous to that used by Jolliffe[13] to find the solution to the PCA problem, which we outline below. Briefly, the proof proceeds by maximizing the objective function in eq 7, followed by finding the next optimal solution ($|\alpha_1\rangle$) subject to it being uncorrelated with $|\alpha_0\rangle$. We solve each step using the method of Lagrange multipliers. The first solution, $|\alpha_0\rangle$, is shown to be a solution to the generalized eigenvalue problem given in eq 8.

$$\mathbf{C}^{(\Delta t)}|\alpha_0\rangle = \lambda_0\Sigma|\alpha_0\rangle \quad (8)$$

We now find another projection that maximizes our objective function while being uncorrelated with $|\alpha_0\rangle$. This produces another optimization problem with two constraints:

$$\max_{|\alpha_1\rangle} f(|\alpha_1\rangle) = \max_{|\alpha_1\rangle}\langle\alpha_1|\mathbf{C}^{(\Delta t)}|\alpha_1\rangle$$
$$\text{subject to:}$$
$$\langle\alpha_1|\Sigma|\alpha_1\rangle = 1$$
$$\langle\alpha_1|\Sigma|\alpha_0\rangle = 0 \quad (9)$$

This can again be solved with Lagrange multipliers, and it can be shown that the solution, $|\alpha_1\rangle$, is a solution to the same generalized eigenvalue problem in eq 8.

$$\mathbf{C}^{(\Delta t)}|\alpha_1\rangle = \lambda_1\Sigma|\alpha_1\rangle \quad (10)$$

These steps can be repeated, and we find that the d solutions to the tICA problem are the d eigenvectors of the same generalized eigenvalue problem. Further, we can relate $f(|\alpha_i\rangle)$ to the ith eigenvalue:

$$\begin{aligned} f(|\alpha_i\rangle) &= \langle\alpha_i|\mathbf{C}^{(\Delta t)}|\alpha_i\rangle \\ &= \langle\alpha_i|(\lambda_i|\Sigma|\alpha_i\rangle) \\ &= \lambda_i\langle\alpha_i|\Sigma|\alpha_i\rangle \\ &= \lambda_i \end{aligned} \quad (11)$$

This means that if we order the eigensolutions by their eigenvalues, such that $\lambda_0 > \lambda_1 > \ldots > \lambda_{d-1}$, then $|\alpha_0\rangle$ is the slowest tIC, $|\alpha_1\rangle$ is the next slowest, etc. This result provides an algorithm for computing the $N \leq d$ slowest degrees of freedom from a multidimensional time series.

1. Compute $\mathbf{C}^{(\Delta t)}$ and Σ from the data.
2. Since the computed $\mathbf{C}^{(\Delta t)}$ may not be exactly symmetric, we add the transpose and divide by two for an estimate of the symmetric $\mathbf{C}^{(\Delta t)}$ (see note below).
3. Solve $\mathbf{C}^{(\Delta t)}|\alpha\rangle = \lambda\Sigma|\alpha\rangle$.
4. Pick N vectors by selecting the eigenvectors with the top N eigenvalues.

In general, the time-lag correlation matrix is symmetric as long as the underlying system is reversible in time. However, the sample time-lag covariance matrix may or may not be symmetric. The simplest solution is to symmetrize the matrix by adding its transpose. This procedure amounts to including each trajectory twice in the data set: once forward and once backward. Unfortunately, if the trajectories are not begun at the true equilibrium distribution, then the resulting calculation may be biased. This is related to the problem encountered when symmetrizing a counts matrix during the MSM construction process.[40]

As the data set generated by Lindorff-Larsen et al.[38] consists of four long trajectories, the bias is negligible in this work.

dx.doi.org/10.1021/ct300878a | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

However, for future analysis of data sets that contain many short trajectories, for example Folding@home data sets, we may need to develop a more robust way of calculating a symmetric estimate of the time-lag correlation matrix.

MSM Construction. MSMs are defined as a set of states and rates of transition between those states. The method is incredibly powerful and has been used successfully in many systems. Part of the power is the ability to automate the construction process, since this removes any bias the scientist may have. This has become possible with the help of new software packages.[27,41,42]

Briefly, the construction process has three steps:

1. Cluster the data by K-Centers, K-Medoids, Hierarchical Methods, or any appropriate clustering method, and assign all conformations to a state. This step transforms $\mathbf{X}_t \rightarrow s(t)$ where $s(t)$ is an integer corresponding to the assigned state at time t.

2. Given a lag time, τ, a transition between state i and state j would occur if at some time t, $s(t) = i$ and $s(t + \tau) = j$. We count all such transitions between all pairs of states in the data set. The result of this is a counts matrix, $\mathbf{C}$, where C_{ij} is the number of transitions between states i and j in the data set.

3. From the counts matrix, estimate the transition probability matrix, $\mathbf{T}$, whose elements, T_{ij}, correspond to the probability of transferring form state i to state j in one lag time. We used a Maximum Likelihood Estimator (MLE) described in Beauchamp et al.[27]

There have been many recent improvements working toward making the MSM construction process more automatic and robust, including new methods for defining state spaces using milestoning,[25] as well as new methods for calculating the transition probability matrix from the raw counts.[27] Additionally, recent improvements in the form of new metrics that can better determine whether two conformations can interconvert rapidly have shown that an MSM can be improved by using a different distance metric to build a better state space.[32,33]

To construct an MSM using the tICA method, we calculate the top N eigenvectors corresponding to the slowest N components (tIC's). The goal is then to define a distance metric that calculates distance only along these components. If $|\mathbf{A}\rangle$ corresponds to a protein conformation, first we define a $d \times N$ projection matrix, $\mathbf{P}$, whose columns are the N slowest tIC's, such that $\mathbf{P}^T|\mathbf{A}\rangle$ is given by

$$\mathbf{P}^T|\mathbf{A}\rangle = \begin{bmatrix} \langle \alpha_0|\mathbf{A}\rangle \\ \langle \alpha_1|\mathbf{A}\rangle \\ \vdots \\ \langle \alpha_{N-1}|\mathbf{A}\rangle \end{bmatrix} \tag{12}$$

This new vector is the reduced representation of the conformation $|\mathbf{A}\rangle$ corresponding to projecting onto the top N tIC's. We can now define the distance between two conformations $|\mathbf{A}\rangle$ and $|\mathbf{B}\rangle$ as

$$d(|\mathbf{A}\rangle, |\mathbf{B}\rangle) = \|\mathbf{P}^T|\mathbf{A}\rangle - \mathbf{P}^T|\mathbf{B}\rangle\|_2 \tag{13}$$

where $\|\cdot\|_2$ denotes the N-dimensional Euclidean norm. In words, we are projecting each conformation onto the N slowest degrees of freedom and calculating the Euclidean distance between points in this reduced space.

Although the tICA method can be applied to any vector representation of the protein conformation, we used a contact map approach where each conformation was represented as a list of all pairwise residue distances. The distances were calculated as the minimum distance between any two heavy atoms of the corresponding residues. We then used the distance metric defined in eq 13 to cluster a subset of the data set using the K-Centers algorithm to produce k generators. The conformations not used in clustering were then assigned to the closest generator using the same metric. All MSM construction was performed using the MSMBuilder package.[27]

We note that the residue−residue distance representation has redundant information in the coordinates as there are $[(N - 3)(N - 2)]/2$ coordinates (we only include pairs that are at least three residues apart) but only $3N - 6$ degrees of freedom in the system. Since we are limited to linear combinations of these coordinates, we will see different results by using different representations of the protein. We believe, however, that there is likely a kernel version of this method that can achieve solutions that are nonlinear in the input coordinates just as kernel-PCA can produce nonlinear solutions to the PCA problem. We leave this description for future work but believe it would be a useful extension to the tICA method.

RESULTS AND DISCUSSION

tICA Builds an Improved Markov State Model over Conventional Metrics. In MSM construction, there are a few tunable parameters that produce different models. These include the number of states constructed (k) and the lag time (τ) used to count transitions (see MSM Construction). It has been theoretically shown that increasing the number of states and the lag time will produce an MSM with the least discretization error.[26,43,44] However, there is a competing source of statistical error since increasing the number of states will reduce the number of samples of each state to state transition. Additionally, increasing the lag time will decrease the temporal resolution of the resulting MSM. For these reasons, we typically choose the number of states and lag time to be as small as possible while still reproducing the kinetics from the raw trajectory data. This approach allows us to balance the discretization and statistical errors to provide the most accurate MSM.

The tICA method adds two more parameters: the correlation lag time (Δt), which is used in building $\mathbf{C}^{(\Delta t)}$, and the number of tICs (N) onto which to project. To optimize these values, we built many MSMs using K-Centers clustering with the tICA distance metric. To assess which values were optimal, we calculated implied time scales at many lag times. The ith implied time scale (t_i) is given by eq 14.

$$t_i = \frac{\tau}{\log \lambda_i} \tag{14}$$

where τ is the lag time and λ_i is the ith eigenvalue of the transition probability matrix. Since the time scales are always underestimated relative to their true values, we can compare two MSMs by realizing that the slower time scales are indicative of a smaller discretization error.[44] This comparison is only fair, however, when the time scales correspond to the same eigenvector, which is likely only to be the case for the slowest time scale since it is typically attributable to the folding eigenvector.

We found that using six tIC's produced an MSM that could adequately reproduce the folding time scale in the raw data (as determined by the RMSD autocorrelation function). In fact, using too many tIC's tended to make the model too fast. This fact demonstrates the utility of the tICA metric to remove quickly decorrelating degrees of freedom that can add noise

dx.doi.org/10.1021/ct300878a | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

into the clustering protocol. The optimal correlation lag time was found to be 200 ns in NTL9. However, using a value between 100 and 500 ns produced similar MSMs, suggesting that the method is robust to choices of the correlation lag time. Preliminarily, we have observed that the optimal correlation lag time in a few other protein systems was 200 ns. We note, however, that for other applications (or even other proteins) this optimal value may be different. (see SI section: "Selection of Tunable Parameters for tICA"" for a more detailed explanation of the parameter choices as well as a discussion of the robustness of the tICA method.)

As compared to a previous model built using the RMSD metric,[29] the tICA MSM used an order of magnitude fewer states but reproduced the raw data with the same efficacy. We calculated the RMSD to the native state (PDB ID: 2HBA, residues 1−39) autocorrelation function and found that the MSM reproduced the raw data's autocorrelation function (Figure 1). The tICA MSM's time scale distribution was

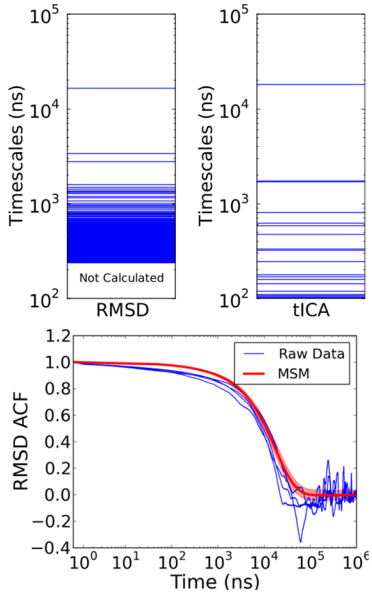


Figure 1. Top: The tICA metric yielded an MSM with a much larger separation of timescales in the 100 ns to 1 μs time range. These time scales were associated with new, compact states in the folding dynamics of NTL9. Bottom: One way of validating an MSM is to calculate autocorrelation functions from the model and compare them to the autocorrelation function calculated directly from the trajectories. The RMSD autocorrelation function of the MSM (red) matched the RMSD autocorrelation function of all four trajectories. The red shading represents one standard deviation in the autocorrelation function calculated from the MSM.

qualitatively different from the previous model. The result was a greater separation of time scales in the submillisecond regime (Figure 1). Interestingly, the tICA MSM was constructed with the K-Centers algorithm, which has its own limitations when applied to protein folding.[27] We applied the Hybrid K-Medoids clustering algorithm;[27] however, it did not change the implied time scales drastically. We hypothesize that the problems with K-Centers have a reduced impact in lower-dimensional spaces.

We also compared the tICA MSM to models built with the same number of states but using conventional PCA or a Euclidean norm on all residue−residue distances (contact map). We found that the tICA MSM produced a slower folding time scale than the PCA MSM and the contact map MSM (Figure 2) as well as provided greater resolution in the submicrosecond regime.

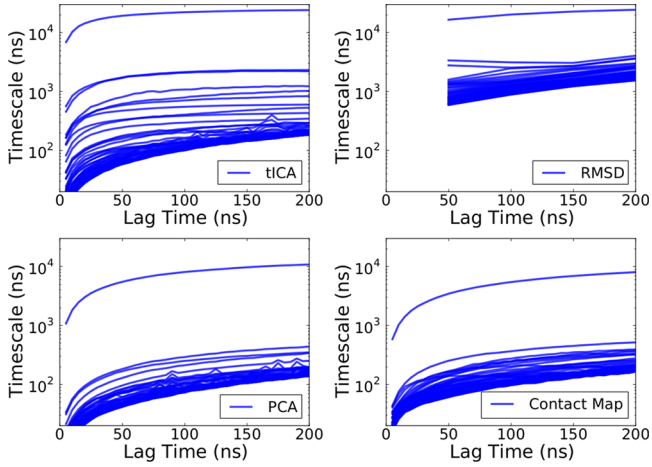


Figure 2. The tICA method (upper left) produced an MSM with slower timescales than the conventional PCA (lower left) as well as the contact map (lower right) approach. This improvement was due to the ability of the tICA method to remove quickly decorrelating degrees of freedom that add noise to the distance calculation. Additionally, the tICA MSM produces the folding time scale as well as the RMSD model built by Beauchamp et al.[29] The tICA MSM produced a qualitatively different time scale distribution from the RMSD method with many time scales corresponding to undiscovered states. We note that Beauchamp et al. used Ward's Method for clustering, which was limited to only a subset of the data. The authors only analyzed the trajectories subsampled by 50 ns, which is why there are no data for lag times below 50 ns.

Slow Time Scales in NTL9 Correspond to Register-Shifted States. In the tICA MSM, the slowest time scale (~18 μs) corresponded to a relaxation between folded and unfolded states, while the second and third slowest time scales corresponded to two different register-shifted states (Figure 3). This is a known phenomenon, and register-shifted conformations have been reported in MSMs of many protein studies.[29,33,45] In our MSM, the first register shift occurred in strand one (residues 1−6) and was previously observed in the RMSD MSM reported in Beauchamp et al.[29] This state had an equilibrium population of approximately 0.5% (5kT). All energies are free energies in units of kT relative to the native state in the MSM. The second register shifted state occurred in strand three (residues 35−39) and was not observed previously; it had an equilibrium population of ~0.1% (6.6kT). Additionally, the fifth slowest time scale (~620 ns) corresponded to the register shift in strand one described above equilibrating with a state with a different register shift in strand three.

Interestingly, but perhaps unsurprisingly, each register shifted state had a corresponding shift in the hydrophobic core contacts. For example, in the strand one shift, the entire strand was flipped as compared to the native state causing the hydrophobic residues normally in the core to be solvent exposed. The strand three shift produced a register shift in the core packing, in which PHE-29 packed where LEU-30 was packed in the native state (Figure 4).

All three of the observed register-shifted states had hydrophobic cores that remained intact. The strand two shift traded two buried hydrophobic residues for two other

dx.doi.org/10.1021/ct300878a | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

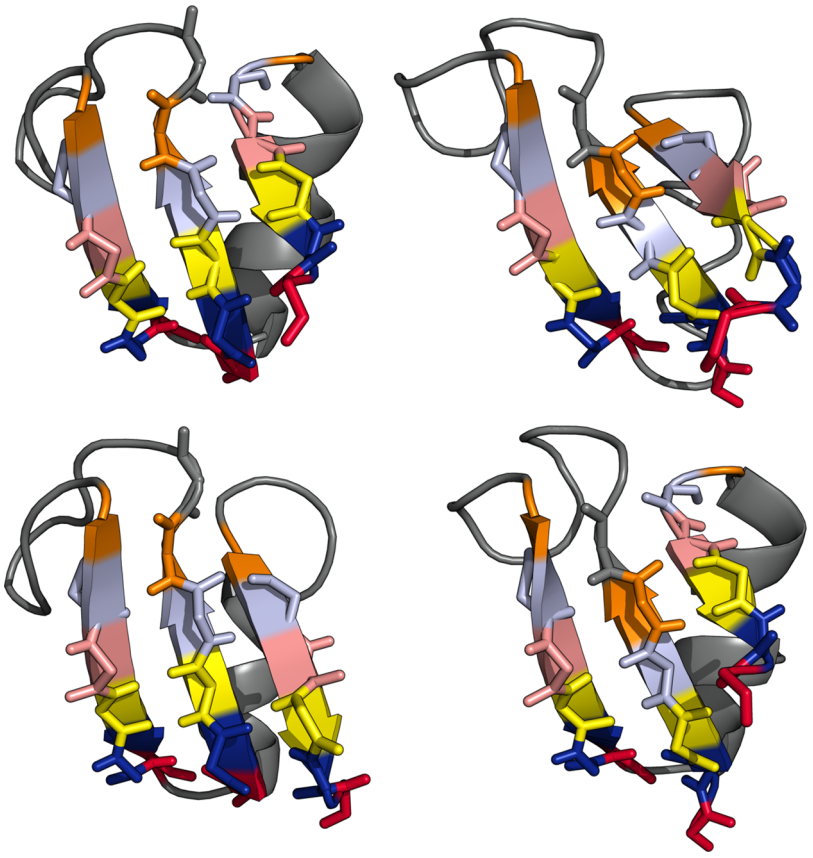


Figure 3. The native state of NTL9 has three beta strands (upper left). Note that strand two is the left-most strand, strand one is the center strand, and strand three is the right-most strand. There were three register shifted states observed in this simulation. The first occurred in strand one (lower right) and was observed in previous studies.[29] This strand one shift could also occur with a shift in strand three's location (upper right). Finally, strand three could shift with strand one in the correct orientation (lower left). The latter two were not observed in previous analysis.

hydrophobic residues (Figure 4), and the strand three shift resulted in different hydrophobic contacts without removing any. These results indicate that register-shifted states may only occur when there is a corresponding shift in the hydrophobic packing that is still favorable. Other states would not be nearly as stable since they would disrupt the core by either removing a hydrophobic residue or placing a polar residue within it.

These register-shifted states shed light on the role of non-native interactions in protein folding, which has been a topic of major debate in the field.[3,46−51] Together with recent results,[29,33] these findings show that non-native interactions are relevant in many protein systems and potentially important to the protein folding process.

The tICA Metric Reveals Non-Native β Sheet Structures in NTL9. The sixth slowest time scale (~590 ns) corresponded to the formation of non-native β sheet structures. These configurations are particularly interesting because they occur in residues that are helical in the native state (Figure 5). This state had an equilibrium population of less than 0.1% ($7kT$).

Other non-native beta sheets occurred in the loop region between strands one and two (Figure 6). The fourth slowest time scale (~800 ns) corresponded to conversion between the folded state and an extended β sheet between strands one and two, whereas in the native state, this loop was folded down to allow the hydrophobic core to pack behind strands one and two. This state had an equilibrium population of ~1.5% ($4kT$).

The tICA Metric Reveals a New, Partially Packed State in NTL9. In addition to the slow time scales, a high equilibrium flux eigenvector was revealed in the tICA MSM. Equilibrium flux is given by eq 15.[29]

$$\Phi_n = \|\phi_n\|^2 \tag{15}$$

where Φ_n is the flux of eigenvector n and ϕ_n is the right eigenvector of the transition probability matrix. For a more detailed explanation, see Beauchamp et al.,[29] but briefly, an eigenvector's flux represents the total population leaving all states due to the eigenvector. Intuitively, low flux eigenvectors may not be as relevant as they only correspond to small shifts of population.

The tICA metric revealed a second high-flux eigenvector (the first being the folding eigenvector) with an associated time scale of ~300 ns (Figure 7). This eigenvector corresponded to a near-native state that had the correct secondary structure but was missing key backbone hydrogen bonds in the core region. The result was a configuration with key hydrophobic residues that are not completely packed. This state had an equilibrium population of ~3.5% ($3.1kT$).

New States in the tICA MSM Are Visited in High Flux Pathways. We constructed a 20 state macrostate MSM from our microstate MSM by implementing the PCCA+ algorithm derived by Deuflhard and Weber.[52] From this macrostate model we wished to determine the folding pathways for NTL9. The two lowest free energy states corresponded to the completely folded and completely unfolded states. We used

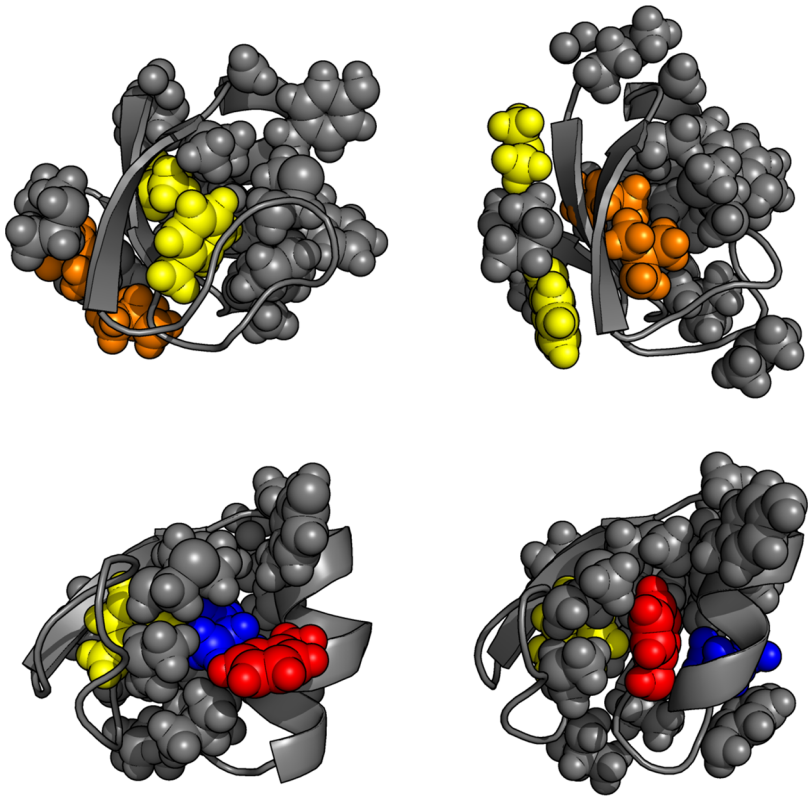


Figure 4. The register shifted states not only formed non-native hydrogen bonds, but they also made non-native hydrophobic contacts. The register shift in strand two caused hydrophobic residues that were natively packed (top left) in the core to become solvent exposed (top right). The register shifts in strand three caused a "register-shift" in the core packing (lower right), in that PHE-29 (red) was packed where LEU-30 (blue) was in the native state (lower left).

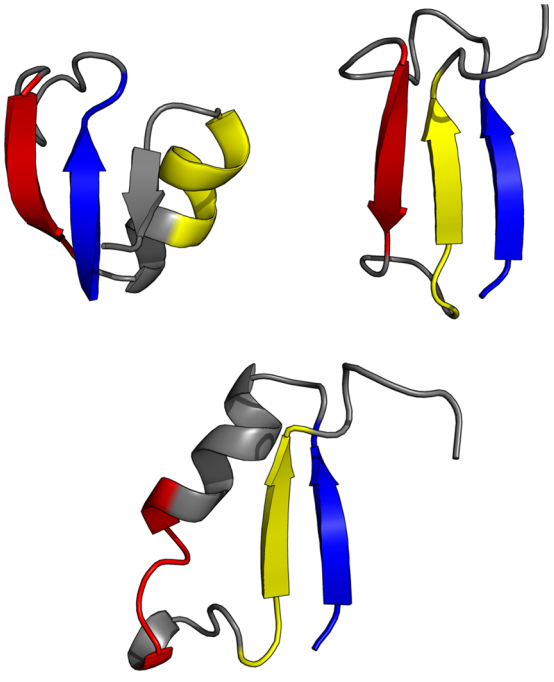


Figure 5. NTL9 was observed to partially fold to one of many non-native β sheet structures. These are particularly interesting because the non-native strand is helical in the native state. The native state is shown in the upper left, while two non-native β sheet structures are shown on the right and bottom. The yellow portion, which is helical in the native state, forms a sheet with residues 1−6 (blue), and less often with residues 17−21 (red).

these states as the source and sink states for calculating the net flux through our model as well as the highest flux pathways through our model by applying transition path theory.[53−55]

From our analysis, 25% of the total reactive flux transferred directly from the unfolded state to the folded state (Figure 8). The remaining flux was split between three paths; the first of which was directed through the partially packed state, which accounted for 17% of the total flux. We also observed two pathways that either first formed strand 1−3 (33%) or strand 1−2 (20%), which are consistent with results from previous studies.[38,56] Additionally, for the strand 1−3 pathway, roughly a quarter of the flux traveled through the partially packed state, whereas the strand 1−2 pathway had roughly half of its flux travel through the partially packed state. The partially packed state was not described in previous studies of WT NTL9[56] or in the original analysis of this data set.[38] Interestingly, the mutant used in this study is the K12M mutant of NTL9, which allows the protein to fold faster (1.5 ms vs 20 μs). The methionine in question is one of the hydrophobic contacts that is not formed in the partially packed state.

P_{fold} Depends on the Location of the Register Shift.

The P_{fold} is the probability that a conformation will fold before it unfolds. This value can be used to gain insight into the underlying system, as a large P_{fold} corresponds to a state that is kinetically close to the folded state, while a small P_{fold} corresponds to a state that is kinetically closer to the unfolded state. For the two states that had register shifts in strand one, we found they had P_{fold}'s close to zero, which means those conformations are highly likely to unfold before reaching the native state. The other register shifted state had a shift in strand

dx.doi.org/10.1021/ct300878a | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

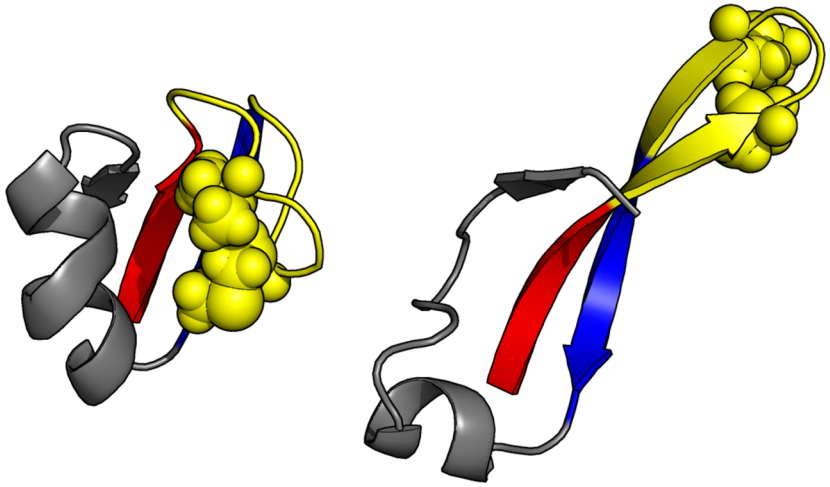


Figure 6. NTL9 was also observed to fold to a configuration that has an extended sheet between strands one and two (right). This extension does not allow the hydrophobic core to pack as it does in the native state (left).

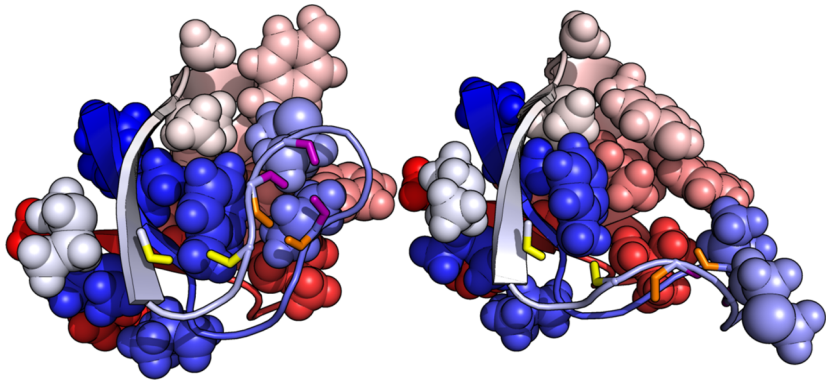


Figure 7. NTL9 also folded to a near-native state (right) that had the same secondary structure as the native state (left) but was missing several core contacts. This was correlated with a few backbone hydrogen bonds in the core region of the protein (colored orange, yellow, and purple) that were not formed in the partially packed state.

three, but interestingly its P_{fold} was approximately 0.6, meaning those conformations are likely to fold before unfolding. In fact, there was a folding pathway that accounted for 1.6% of the total flux through the MSM that traveled through the strand three register shift.

This qualitative difference is expected since the register shift in strand three only corresponds to four non-native, backbone hydrogen bonds. The shifts in strand one, however, contain at least eight non-native, backbone hydrogen bonds, so it will be easier to reverse the register shift in strand three without completely unfolding.

Previous, General Results Are Consistent with the tICA MSM. Our results generally agree with the findings of Lindorff-Larsen et al.,[38] who made two general conclusions. The first was that the unfolded ensemble in protein folding is characterized by compact states with numerous contacts formed. This is consistent with the compact, yet non-native states we observed, including register shifted states and nonspecific beta sheets. Furthermore, we found that one of the register-shifted states is visited en route to the native state, though we note that this pathway accounts for very little of the reaction flux ($\sim$ 1.6%).

Lindorff-Larsen and co-workers also concluded that of their set of small proteins, only two folded along more than one pathway (NuG2 and NTL9). The pathways from our analysis provide a heterogeneous picture consistent with that description. The tICA MSM, however, is also able to resolve a partially packed state that we observed in pathways accounting for ~40% of the total reaction flux.

CONCLUSIONS

There are many practical limitations that one may encounter when constructing an MSM. For example, increasing the number of states or the lag time used will typically produce a model with slower kinetics, which may be necessary to reproduce the raw data. However, larger state decompositions increase the statistical error in calculating transition probabilities, and larger lag times produce models with poor temporal resolution. Toward this end, we have shown that designing a better distance metric can build models with fewer states while using inferior clustering algorithms (K-Centers).

The result of these improvements is an enhanced picture of the folding of NTL9. We find that there are many compact unfolded states characterized by a large number of non-native contacts. Specifically, there are many register shifted states, as well as non-native β sheet configurations that exist in the unfolded ensemble. We believe that the relevance of these non-native states has been underestimated because our analysis has been limited. However, our improvements suggest that

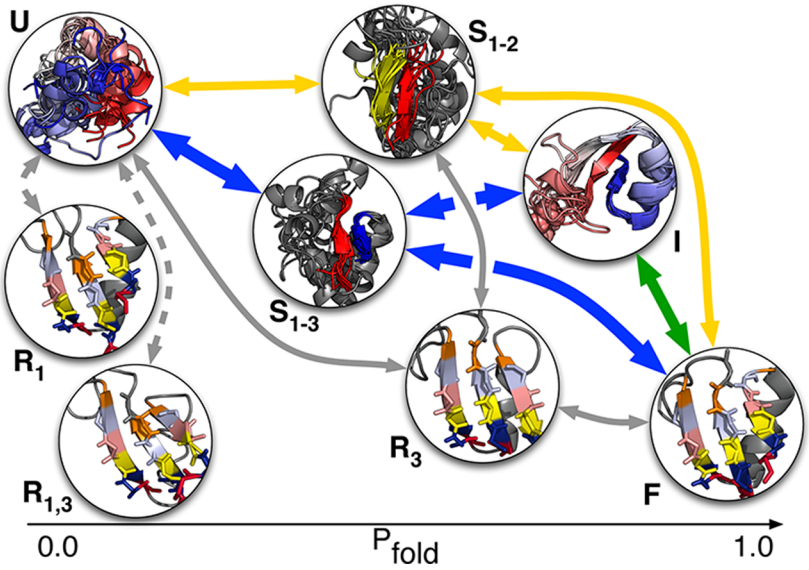


Figure 8. In the tICA MSM, 25% of all reactive flux transferred from the unfolded to folded states in one step. The remaining flux was primarily split between three paths. The first path, which accounted for 17% of the flux, visited a partially packed intermediate between the unfolded and folded states. The second and third paths corresponded to forming strand 1−3 (33%) or strand 1−2 (20%) first. When forming strand 1−3 first, a third of the reactive flux proceeded through the partially packed intermediate, while when forming strand 1−2 first, half of the flux proceeded through this intermediate. Additionally, The register shifts that occurred in strand one (R_1 and $R_{1,3}$) had P_{fold}'s close to zero, whereas the shift in strand three (R_3) had a $P_{fold} \sim 0.6$. In fact, the strand three register shift was visited by 1.6% of the reactive flux from the TPT analysis. We note that the width of the solid arrows above indicate their relative fluxes in the MSM.

kinetically motivated metrics will provide a tool to find new intermediates that previous analyses did not observe.

■ ASSOCIATED CONTENT

Supporting Information

A more detailed proof of the tICA solution is presented as well as insight into how we picked tunable parameters for the tICA method applied to MSM construction. This material is available free of charge via the Internet at http://pubs.acs.org/.

■ AUTHOR INFORMATION

Corresponding Author

*E-mail: pande@stanford.edu.

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

We would like to thank Kyle A. Beauchamp, Thomas J. Lane, Robert McGibbon, and Jeff Weber for useful discussions throughout the development of this project. The Pande group gratefully acknowledges support from NIH and NSF, in particular, grants NIH R01-GM062868 and NSF-MCB-0954714. In addition, the authors acknowledge the following award for providing computing resources that have contributed to the research results reported within this paper: MRI-R2: Acquisition of a Hybrid CPU/GPU and Visualization Cluster for Multidisciplinary Studies in Transport Physics with Uncertainty Quantification. This award is funded under the American Recovery and Reinvestment Act of 2009 (Public Law 111-5).

■ REFERENCES

(1) Ensign, D. L.; Kasson, P. M.; Pande, V. S. *J. Mol. Biol.* **2007**, *374*, 806−816.
(2) Lane, T. J.; Bowman, G. R.; Beauchamp, K.; Voelz, V. A.; Pande, V. S. *J. Am. Chem. Soc.* **2012**, *133*, 18413−18419.
(3) Bowman, G. R.; Voelz, V. A.; Pande, V. S. *J. Am. Chem. Soc.* **2012**, *133*, 664−667.
(4) Lin, Y.-S.; Bowman, G. R.; Beauchamp, K. A.; Pande, V. S. *Biophys. J.* **2012**, *102*, 315−324.
(5) Buch, I.; Giorgino, T.; De Fabritiis, G. *Proc. Natl. Acad. Sci. U. S. A.* **2011**, *108*, 10184−10189.
(6) Shirts, M.; Pande, V. S. *Science* **2000**, *290*, 1903−1904.
(7) Shaw, D. E.; et al. *Commun. ACM* **2008**, *51*, 91−97.
(8) Clementi, C.; Jennings, P. A.; Onuchic, J. N. *J. Mol. Biol.* **2001**, *311*, 879−890.
(9) Clementi, C.; Plotkin, S. S. *Protein Sci.* **2004**, *13*, 1750−1766.
(10) Hills, R. D., Jr.; Brooks, C. L., III. *J. Mol. Biol.* **2008**, *382*, 485−495.
(11) Krivov, S. V.; Karplus, M. *Proc. Natl. Acad. Sci. U. S. A.* **2004**, *101*, 14766−14770.
(12) Muff, S.; Caflisch, A. *Proteins* **2008**, *70*, 1185−1195.
(13) Jolliffe, I. T. *Principal Component Analysis*; Springer: New York, 2002; pp 1−9.
(14) Amadei, A.; Linssen, A. B. M.; Berendsen, H. J. C. *Proteins* **1993**, *17*, 412−425.
(15) Hummer, G.; García, A. E.; Garde, S. *Proteins* **2001**, *42*, 77−84.
(16) Mu, Y.; Nguyen, P. H.; Stock, G. *Proteins* **2005**, *58*, 45−52.
(17) Tenenbaum, J. B.; de Silva, V.; Langford, J. C. *Science* **2000**, *290*, 2319−2323.
(18) Das, P.; Moll, M.; Stamati, H.; Kavraki, L. E.; Clementi, C. *Proc. Natl. Acad. Sci. U. S. A.* **2006**, *103*, 9885−9890.
(19) Coifman, R. R.; Lafon, S. *Appl. Comput. Harmonic Anal.* **2006**, *21*, 5−30.
(20) Ceriotti, M.; Tribello, G. A.; Parrinello, M. *Proc. Natl. Acad. Sci. U. S. A.* **2011**, *108*, 13023−13028.
(21) Swope, W. C.; Pitera, J. W.; Suits, F. *J. Phys. Chem. B* **2004**, *108*, 6571−6581.
(22) Noé, F.; Horenko, I.; Schütte, C.; Smith, J. C. *J. Chem. Phys.* **2007**, *126*, 155102.
(23) Noé, F.; Fischer, S. *Curr. Opin. Struct. Biol.* **2008**, *18*, 154−162.
(24) Pande, V. S.; Beauchamp, K.; Bowman, G. R. *Methods (San Diego, Calif.)* **2010**, *52*, 99−105.

(25) Schütte, C.; Noé, F.; Lu, J.; Sarich, M.; Vanden-Eijnden, E. *J. Chem. Phys.* **2011**, *134*, 204105.
(26) Prinz, J.-H.; Wu, H.; Sarich, M.; Keller, B.; Senne, M.; Held, M.; Chodera, J. D.; Schütte, C.; Noé, F. *J. Chem. Phys.* **2011**, *134*, 174105.
(27) Beauchamp, K. A.; Bowman, G. R.; Lane, T. J.; Maibaum, L.; Haque, I. S.; Pande, V. S. *J. Chem. Theory Comput.* **2012**, *7*, 3412−3419.
(28) Gonzalez, T. F. *Theor. Comput. Sci.* **1985**, *38*, 293−306.
(29) Beauchamp, K. A.; McGibbon, R.; Lin, Y.-S.; Pande, V. S. *Proc. Natl. Acad. Sci. U. S. A.* **2012**, 17807−17813.
(30) Ward, J. H., Jr. *J. Am. Stat. Assoc.* **1963**, *58*, 236−244.
(31) Müllner, D. 2011, arxiv: 1109.2378.
(32) Zhou, T.; Caflisch, A. *J. Chem. Theory Comput.* **2012**, 2930−2937.
(33) Kellogg, E. H.-M.; Lange, O.; Baker, D. *J. Phys. Chem. B* **2012**, 11405−11413.
(34) Riccardi, L.; Nguyen, P. H.; Stock, G. *J. Phys. Chem. B* **2009**, *113*, 16660−16668.
(35) Molgedey, L.; Schuster, H. *Phys. Rev. Lett.* **1994**, *72*, 3634−3637.
(36) Blaschke, T.; Berkes, P.; Wiskott, L. *Neural. Comput.* **2006**, *18*, 2495−2508.
(37) Naritomi, Y.; Fuchigami, S. *J. Chem. Phys.* **2011**, *134*, 065101.
(38) Lindorff-Larsen, K.; Piana, S.; Dror, R. O.; Shaw, D. E. *Science* **2011**, *334*, 517−520.
(39) Altis, A.; Nguyen, P. H.; Hegger, R.; Stock, G. *J. Chem. Phys.* **2007**, *126*, 244111.
(40) Scalco, R.; Caflisch, A. *J. Chem. Phys. B* **2011**, *115*, 6358−6365.
(41) Bowman, G. R.; Beauchamp, K. A.; Boxer, G.; Pande, V. S. *J. Chem. Phys.* **2009**, *131*, 124101.
(42) Senne, M.; Trendelkamp-Schroer, B.; Mey, A. S.; Schütte, C.; Noé, F. *J. Chem. Theory Comput.* **2012**, *8*, 2223−2238.
(43) Sarich, M.; Noé, F.; Schütte, C. *Multiscale Model. Simul.* **2012**, *8*, 1154−1177.
(44) Djurdjevac, N.; Sarich, M.; Schütte, C. *Multiscale Model. Simul.* **2012**, *10*, 61−81.
(45) Lin, Y.-S.; R., B. C.; Kyle, B.; Voelz, V. A.; Tokmakoff, A.; Pande, V. S. Submitted 2012.
(46) Wolynes, P. G.; Onuchic, J. N.; Thirumalai, D. *Science* **1995**, *267*, 1619−1620.
(47) Clementi, C.; Nymeyer, H.; Onuchic, J. N. *J. Mol. Biol.* **2000**, *298*, 937−953.
(48) García, A. E.; Onuchic, J. N. *Proc. Natl. Acad. Sci. U. S. A.* **2003**, *100*, 13898−13903.
(49) Lammert, H.; Wolynes, P. G.; Onuchic, J. N. *Proteins* **2012**, *80*, 362−373.
(50) Bowman, G. R.; Pande, V. S. *Proc. Natl. Acad. Sci. U. S. A.* **2010**, *107*, 10890−10895.
(51) Voelz, V. A.; Jäger, M.; Yao, S.; Chen, Y.; Zhu, L.; Waldauer, S. A.; Bowman, G. R.; Friedrichs, M.; Bakajin, O.; Lapidus, L. J.; Weiss, S.; Pande, V. S. *J. Am. Chem. Soc.* **2012**, *134*, 12565−12577.
(52) Deuflhard, P.; Weber, M. *Linear Algebra Appl.* **2005**, *398*, 161−184.
(53) Metzner, P.; Schütte, C.; Vanden-Eijnden, E. *Multiscale Model. Simul.* **2009**, *7*, 1192−1219.
(54) Berezhkovskii, A.; Hummer, G.; Szabo, A. *J. Chem. Phys.* **2009**, *130*, 205102.
(55) Noé, F.; Schütte, C.; Vanden-Eijnden, E.; Reich, L.; Weikl, T. R. *Proc. Natl. Acad. Sci. U. S. A.* **2009**, *106*, 19011−19016.
(56) Voelz, V. A.; Bowman, G. R.; Beauchamp, K.; Pande, V. S. *J. Am. Chem. Soc.* **2010**, *132*, 1526−1528.