Building Markov state models along pathways to determine free energies and rates of transitions

Albert C. Pan[1] and Benoît Roux[1,2,a)]
[1]*Department Biochemistry and Molecular Biology, Gordon Center of Integrative Science, University of Chicago, Chicago, Illinois, 60637 USA*
[2]*Bioscience Division, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, Illinois, 60439 USA*

(Received 15 May 2008; accepted 26 June 2008; published online 12 August 2008)

An efficient method is proposed for building Markov models with discrete states able to accurately describe the slow relaxation of a complex system with two stable conformations. First, the reaction pathway described by a set of collective variables between the two stable states is determined using the string method with swarms of trajectories. Then, short trajectories are initiated at different points along this pathway to build the state-to-state transition probability matrix. It is shown, using a model system, how this strategy makes it possible to use trajectories that are significantly shorter than the slowest relaxation time to efficiently build a reliable and accurate Markov model. Extensions of the method to multiple pathways, as well as some common pitfalls arising from poorly relaxed paths or an inappropriate choice of collective variables, are illustrated and discussed. © *2008 American Institute of Physics*. [DOI: 10.1063/1.2959573]

I. INTRODUCTION

Computer simulations offer a unique virtual route to gain insight into the motion of macromolecules, providing information that cannot be obtained with existing experimental techniques. Typical simulations of fully atomistic systems of biological interest, however, are often limited to timescales of 10–100 ns, whereas a great majority of biological events of physiological importance, such as the gating of ion channels[1] and the switching of signal transduction proteins,[2] occur on timescales ranging from several microseconds to milliseconds. Performing brute force simulations of these systems for times long enough to determine accurate rates and relaxation timescales seems impractical, if not infeasible. Alternatively, with the increasing availability of distributed parallel computer clusters, the simulation of tens of thousands of short, independent trajectories can be executed with relative ease. Several theoretical framework exist to "stitch" together the information from these independent trajectories in order to create a coherent dynamical picture of the process.[3–11] The general idea consists in constructing, from the underlying detailed microscopic dynamics, a reduced stochastic model with transition probabilities upon which the long timescale behavior can be inferred.

Markov models (MMs) with discrete states, in particular, have recently emerged as a powerful and conceptually simple method for determining rates and energetics of systems by offering a formal way to combine dynamical information from a collection of trajectories shorter than the ultimate timescale of interest.[7–9] MMs with discrete states have been used to describe processes such as protein folding,[8–12] the hydration of carbon nanotubes,[13] membrane fusion,[14] and electron paramagnetic resonance spectra.[15]

Building a MM with discrete states for a dynamical process involves splitting up a system's phase space into regions, running independent short trajectories visiting each of those regions, and then tabulating the observed state-to-state transitions into a Markov matrix of transition probabilities. If the model constructed is indeed Markovian, then this matrix fully describes the system at long timescales. A major strength of the Markov approach is that any given independent trajectory need not visit every state, thereby avoiding the need to run long trajectories that are on the order of the slowest relaxation time in the system.

One of the fundamental difficulties in attempting to construct an accurate MM is the definition of states which must correctly map out the dynamically relevant regions of phase space. Failure to define appropriate states can severely affect the accuracy and usefulness of a MM.[7,8,16,17] An important requirement of Markov states is that they are defined by a set of collective variables which are "good" reaction coordinates, e.g., coordinates along which a system progresses slowly relative to degrees of freedom orthogonal to them.[18–20] While a set of collective variables discriminating between stable states may be enough to describe the thermodynamics of a system, good reaction coordinates are needed to describe the kinetics of transitions between those stable states in order to build a reliable model.

Many of the previous efforts to build MMs with discrete states have proceeded by clustering large numbers of configurations generated from long brute force trajectories (sometimes at high temperature) or replica exchange molecular dynamics simulations.[8,12,16] In the context of an effort aimed at mapping out the transition between two well-defined conformational states of a large biomolecular structure, however, the effectiveness of this strategy quickly becomes limited.

Clearly, an important requirement for constructing a re-

a)Electronic mail: roux@uchicago.edu.

liable reduced stochastic model is to include a set of representative states by focusing on the dynamically meaningful regions of phase space involved in the transition. This observation is a key insight of the "milestoning" algorithm of Faradjian and Elber,[10] according to which a non-Markovian hopping mechanism is constructed from underlying microscopic dynamics following predetermined states along a reaction coordinate. Here, we explore a somewhat related approach aimed at building a MM by using the points along the transition pathway determined from the string method with swarm of trajectories between two stable end states.[21] The two main ingredients of the present effort, the string method and Markovian dynamics, are justified by the following observations. The string method, given the right choice of reaction coordinates, is known to yield an isocommittor pathway able to efficiently localize the transition state region.[22,23] Thus, one may reasonably believe that the pathway obtained from the string method would be particularly advantageous for the purpose of building a reduced stochastic model. The choice of Markovian dynamics, on the other hand, is motivated by the great simplicity that it confers to the underlying stochastic scheme. For instance, the scheme may be easily generalized and extended to problems involving multiple reaction channels and pathways.

In Sec. II, we give details of the methodology we propose. Section III describes the computational details and results of applying those methods to a model system. Finally, we compare the present approach to other methods for defining the states of MMs and discuss its advantages and disadvantages in Sec. IV.

II. THEORETICAL BACKGROUND AND METHODS

A. Building a Markov state model

The first step in building a MM is to define a set of K states h_i, which spans phase space. That is, a set of indicator functions,

$$h_i(\boldsymbol{r}) = \begin{cases} 1, & \text{if } \boldsymbol{r} \text{ is in state } i \\ 0, & \text{otherwise,} \end{cases} \tag{1}$$

is defined such that $\Sigma_{i=1}^{K} h_i(\boldsymbol{r}) = 1$ for all $\boldsymbol{r}$, where $\boldsymbol{r}$ is a set of Cartesian or collective variables which describe a particular configuration of the system. The results from a series of free independent trajectories which travel between the states are then tabulated into a transition count matrix, $z_{ij}(\Delta t)$, which indicates the number of trajectories that began in state i and ended up in state j after a lag time Δt. Defining $n_i(\Delta t) = \Sigma_{j=1}^{K} z_{ij}(\Delta t)$ to be the total number of trajectories which leave state i in a time Δt (i.e., the total number of transitions observed in a particular row of z_{ij} for a given lag time), we then have that the transition probability to be in state j given that the system was in state i at a time Δt in the past is

$$p_{ij}(\Delta t) = z_{ij}(\Delta t)/n_i(\Delta t). \tag{2}$$

Refer to the Appendix for a discussion of how errors were estimated in the transition probability matrix using a Bayesian approach.[24,25]

Once the p_{ij} matrix is tabulated, many quantities of interest can be calculated, including mean first passage times and committor distributions.[9] In this work, we will primarily be concerned with the slowest relaxation time τ of the system and the probability distribution over the regions $\langle h_i \rangle$. The latter quantity is defined as

$$\langle h_i \rangle = \frac{\int d\boldsymbol{r}\ e^{-\beta U(\boldsymbol{r})} h_i(\boldsymbol{r})}{\int d\boldsymbol{r}\ e^{-\beta U(\boldsymbol{r})}}, \tag{3}$$

where U is the potential energy of the system and β is one over Boltzmann's constant times the temperature, $1/k_B T$. The average $\langle h_i \rangle$ is related to the free energy f_i of region i via the equation $f_i = -k_B T \ln\langle h_i \rangle$ and is given by the left eigenvector π_i of p_{ij} corresponding to a unit eigenvalue. That is, π_i is the solution to $\Sigma_{i=1}^{K} \pi_i p_{ij} = \pi_j$. The relaxation timescale τ is given as

$$\tau = -\frac{\Delta t}{\ln(\lambda_s)}, \tag{4}$$

where λ_s corresponds to the largest eigenvalue of p_{ij} less than 1. The properties of the transition probability matrix as well as derivations of Eqs. (3) and (4) have been discussed in detail elsewhere.[7] Here, we briefly elaborate on some of the expected properties of p_{ij}.

We begin by noting that p_{ij} is a stochastic matrix such that all the elements, $p_{ij} \geq 0$ and that $\Sigma_{j=1}^{K} p_{ij} = 1$ (row normal convention). The latter requirement is a statement of conservation of probability: given that the system began in state i, it must either transition to some other state j or remain in state i after a given lag time. The eigenvalues of such a matrix have the property that their modulus is always less than or equal to 1. The transition probability matrix is also ergodic such that every state can reach every other state in a finite amount of time. Ergodicity implies that there exists only one unit eigenvalue in the eigenspectrum of p_{ij}.[7]

Moreover, in reversible thermodynamic systems, detailed balance is always satisfied, i.e., $\pi_i p_{ij} = \pi_j p_{ji}$. We would expect detailed balance to hold for all Markov matrices constructed from equilibrium molecular dynamics simulations, for example. One important consequence of a matrix which obeys detailed balance is that all its eigenvectors and eigenvalues are real. Due to statistical fluctuations, a finitely sampled transition probability matrix may not strictly satisfy detailed balance. Although there exist ways to impose detailed balance (see, e.g., Ref. 16), we choose instead to take the real part of computed eigenvectors and eigenvalues as was done by Swope *et al.*[7,8]

B. Defining states along a path

A central element of the present effort is the strategy for defining the states h_i by relying on a pathway determined by the string method with swarms of trajectories.[21] Such a choice insures that the dynamically meaningful configurations required for a transition are included.[10]

The procedure begins by finding a putative pathway between stable states defined with a set of collective variables (i.e., distances, dihedrals, solvent densities, etc.). In a complex situation, such as a large conformational change, this

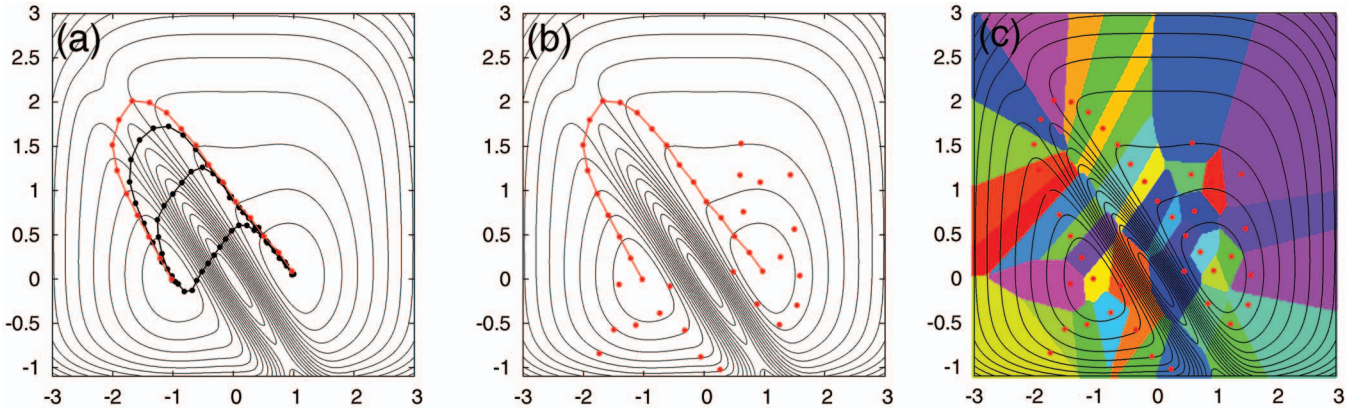

FIG. 1. (Color) Building a MM along a path for a simple potential. (a) First, dynamically important regions of phase space are mapped out along a pathway of interest. Here, this is done using the string method with swarms of trajectories. The series of paths show the convergence of an initial path to the most likely transition path (red). (b) Second, free simulations are run in both basins to increase coverage: although transition pathways are often narrow tubes in phase space, stable basins may be wider and would thus be inadequately represented by single points. Extra regions are added only if they are a certain minimum distance from all pre-existing points. (c) Finally, each of the points is used as a centroid for a region in the MM. Short trajectories are initiated from each of these regions and a transition matrix is tabulated. All illustrations in this figure are for an inverse temperature, $\beta=1.25$.

initial path will need to be refined in order to discover the most probable transition path, as in Fig. 1(a). Defining a basis for the MM using points along an initial unrefined path obtained from targeted molecular dynamics, for example, can be misleading since such a path may pass through regions of high free energy. This would give an incorrect picture of the underlying dynamical process as described by the MM and overestimate the rate of relaxation.

Although the dynamical pathway between two stable states may exist in a narrow phase space tube,[26] the regions around stable basins may be inaccurately represented by a single point. To check if fluctuations within the basins are under sampled, longer simulations are run within each basin and points are added which are greater than a certain length from pre-existing points. This procedure is illustrated in Fig. 1(b).

The points ("images") along the converged path, together with the extra points sampled from the basins, are then used as a "basis set" for defining the dynamical states h_i for the MM. A clustering scheme where each point is taken to be the centroid of a region can be used. That is, a region is defined around a given centroid such that it includes all points which are closer to that centroid than all other centroids. An illustration of this clustering scheme is shown in Fig. 1(c). The colored (Voronoi) polygons indicate the regions surrounding each centroid. This clustering scheme is easily generalized to multiple dimensions. Finally, short trajectories are run beginning from each region, and their transitions are tabulated into a transition matrix z_{ij}. Computational details as well as results of this methodology on the two dimensional surface shown in Fig. 1 are presented in the following section.

III. RESULTS

A. Computational details

All numerical simulations were run with Langevin dynamics on the model potential surface depicted in Fig. 1. We imagine a single particle of mass m evolving in two dimensions, $\boldsymbol{r}=(x,y)$, according to

$$m\ddot{\boldsymbol{r}}(t) = -\gamma m\dot{\boldsymbol{r}}(t) - \nabla U(\boldsymbol{r}) + \mathcal{R}(t), \tag{5}$$

where γ is the friction coefficient and $\mathcal{R}$ is a Gaussian random force such that $\langle \mathcal{R}(t)\rangle = 0$ and $\langle \mathcal{R}(t)\mathcal{R}(t')\rangle = 2\gamma m k_B T \delta(t-t')$.[27] The potential energy U is given by

$$\begin{aligned} U(x,y) = &-3\exp(-(x-1)^2+y^2) - 3\exp((x+1)^2+y^2) \\ &+ 15\exp\left(-\tfrac{8}{25}(x^2+y^2+20(x+y)^2)\right) \\ &+ \tfrac{32}{625}(x^4+y^4) + \tfrac{2}{5}\exp(-2-4y). \end{aligned} \tag{6}$$

The potential is meant to represent a two-state system with a nontrivial reaction pathway where both degrees of freedom, x and y, are necessary to describe the transition.[28]

The units of length, energy, and mass are angstroms (Å), kcal/mol, and atomic mass units (amu), respectively. That is, we use the AKMA system of units as used, for example, in the CHARMM simulation program.[29] The mass for all the simulations is set to 1000 amu, the molecular weight of approximately 10 amino acids, while the friction coefficient is set to 1.22 in units of inverse time. One time unit in the AKMA system is about 0.05 ps. These parameters give a diffusion constant of 0.01 Å^2/ps at 300 K ($\beta=1.68$). The choice of units attempts to make an analogy with an atomistic system (i.e., a solvated protein undergoing a conformational change). For integrating the Langevin equations of motion, we used an integration timestep, $\delta t=1$ (about 50 fs).

A MM was constructed for the two dimensional potential given by Eq. (6) at six different temperatures ranging from $\beta=0.50$ to 2.00. The string method with swarms of trajectories[21] was used to find the most probable transition path between the two basins beginning from an initial path of 20 images linearly interpolated from $x=-1$ to $x=1$. For each image, 100 short trajectories of length of 0.5 ps were launched from the same point with random initial Gaussian velocities and the image was moved to the average of the end points of those short trajectories. After all the images were updated, the pathway was reparametrized by interpolating a curve through the images and spacing them equidistantly along the curve. This procedure was repeated for 100 itera-

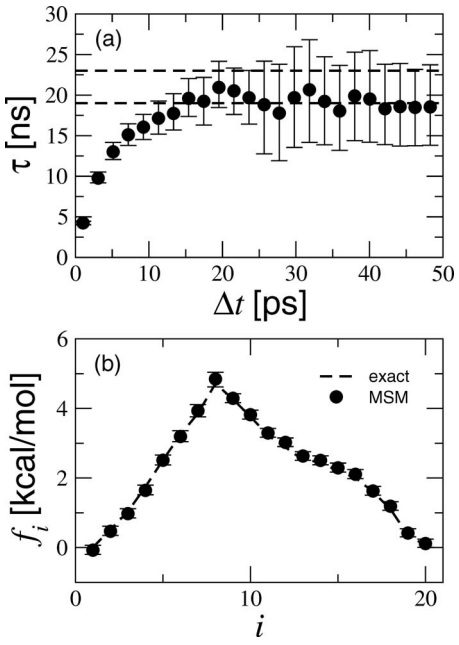

FIG. 2. Determining relaxation times and free energies with MMs. (a) The convergence of the relaxation time as a function of lag time. The two dashed lines represent the range of relaxation timescales estimated by a brute force simulation. Note that a relaxation time of 20 ns can be predicted from a series of short trajectories no longer than 50 ps. (b) The relative free energy of regions 1–20 along the string pathway (the free energy of the extra basin regions are not shown for clarity). The dashed curve is an exact result obtained by numerically integrating over the potential energy surface. The filled circles are estimates from a Markov state model and represent median values. The error bars show the 95% Bayesian credibility interval around the median (see the Appendix). Results are shown for $\beta=1.25$ and are representative of results at other temperatures.

tions. Snapshots of this procedure at $\beta=1.25$ are shown in Fig. 1(a). Refer to Refs. 21 and 23 for further details.

More images were added in each basin via extra simulations run in both stable regions for lengths ranging from 100 ps to 1.5 ns depending on the temperature. A point was added if it was at least a distance d away from all other points, where d was the average distance between points along the converged string. The length of the runs was chosen such that approximately 20 more points were added at every temperature. This procedure is illustrated in Fig. 1(b) for $\beta=1.25$.

The collection of points along the string and in each basin was taken as centroids of a state space for a MM [see Fig. 1(c)]. To build the MM, 50–1000 trajectories were run from each centroid with initial random Gaussian velocities for 50 ps. The transition counts were tabulated into a transition count matrix $z_{ij}(\Delta t)$ at different lag times Δt.

To make a direct comparison with the results from the multistate MM along a path, we determined the relaxation timescales from long trajectories by constructing a two-state MM where the states were defined by

$$h_A(x,y)=\begin{cases}1, & x+y\leq 0\\ 0, & \text{otherwise}\end{cases} \tag{7}$$

and

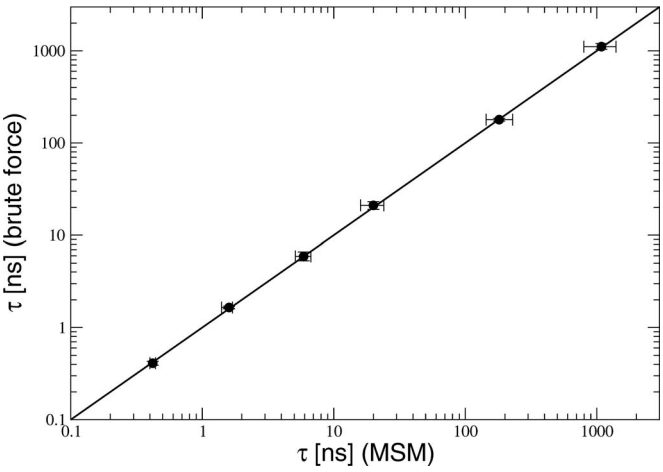

FIG. 3. Comparison of relaxation times calculated with MMs and brute force dynamics at six different temperatures ranging from $\beta=0.50$ to 2.00. The solid black line is $y=x$. There is quantitative agreement over several orders of magnitude. Error estimates are as in Fig. 2.

$$h_B(x,y)=\begin{cases}1, & x+y>0\\ 0, & \text{otherwise}.\end{cases} \tag{8}$$

These indicator functions divide the potential surface into two regions along the line $y=-x$. Since this is a two-state system, the slowest relaxation timescale determined from this two-state MM should be the same as that determined from the MM along the path. The exact free energies, $f_i=-k_BT\ln\langle h_i\rangle$, were computed by numerically integrating Eq. (3) over the potential given by Eq. (6).

B. Relaxation timescales and free energies

Figure 2(a) shows the typical convergence of the slowest relaxation time as a function of lag time at $\beta=1.25$. As discussed in Sec. II A, this relaxation timescale is given by Eq. (4) and explicitly depends on Δt. After a certain transient lag time (in this case ~20 ps), however, the predicted timescale from the MM becomes independent of Δt. This type of plateau behavior strongly implies the onset of Markovian behavior at the lag time when the plateau begins. Examination of the dependence of predicted timescales on lag time has been used previously as an indication of Markovian behavior and is considered one of the more stringent tests that such behavior exists.[7,30] Similar plateau behavior is observed at other temperatures (not shown).

Figure 2(b) shows the prediction of the free energy of a given state $\langle h_i\rangle$ along the string pathway from the MM at $\Delta t=20$ ps (filled circles) compared to the result from numerical integration at $\beta=1.25$. The free energy of the extra basin states is omitted for clarity. The agreement is quantitative within the estimated error. Once again, similar results are obtained at other temperatures (not shown).

The predictions of the MM for the relaxation timescales at different temperatures are compared to the results from brute force calculations in Fig. 3. The relaxation timescales at each β were extracted from plots like Fig. 2(a) after the onset of Markovian behavior in $\Delta t=20-30$ ps. Once again, the agreement is quantitative within the estimated error. We stress here that the MM is able to predict relaxation time-

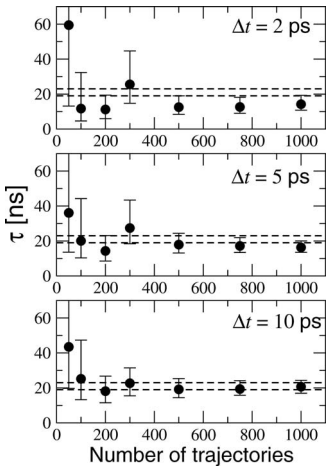

FIG. 4. Accuracy of Markov state model predictions as a function of lag time and number of trajectories at $\beta=1.25$. In each of the panels, the twin dashed lines indicate the range of values for the relaxation timescale predicted by the brute force calculations. The black circles indicate predictions of the MM as a function of the number of trajectories from each region. The three panels show snapshots of these predictions at increasing lag times. Error estimates are as in Fig. 2.

scales much longer than the length of the independent short trajectories used to construct the MM. At the lowest temperature studied, the timescale gap between short trajectories and estimated timescales was as much as five orders of magnitude.

Even in terms of aggregate simulation time, MMs are competitive with brute force simulations. For example, Fig. 4 shows the convergence of the relaxation timescale for $\beta=1.25$ as a function of the number of short trajectories launched from each of 40 regions to estimate the MM. The three panels represent increasing lag times from top to bottom. At short lag times, the relaxation timescale is generally underestimated even if 1000 trajectories are used. The error, however, is still well within the correct order of magnitude, even at $\Delta t=2$ ps. At $\Delta t=10$ ps with 200 trajectories, the MM estimate of τ is very accurate. Considering that there are 40 regions, this would give an aggregate simulation time of approximately 10 ns, only half of the relaxation timescale itself. Adaptive sampling,[24,25] where short trajectories are selectively launched from regions which contribute the most to the estimated statistical uncertainty, could serve to reduce the aggregate simulation time even further.

Finally, we point out some pitfalls when defining MMs with inappropriate states (Fig. 5). For this potential surface, the two metastable basins are quite broad. If only the points along the path are used as regions, as in Fig. 5(a), the resulting MMs systematically underestimate the relaxation timescales by 20%–30% in the range of temperatures studied.

On the other hand, if a badly relaxed path is used, as in Fig. 5(b), the resulting Markov matrices do not show any sign of convergence as a function of lag time for 50 ps. That is, $\tau(\Delta t)$ does not reach a plateau as in Fig. 2(a). Rough estimates of the relaxation timescale extracted from the transition matrices greatly overestimate the relaxation timescales by orders of magnitude. A similar situation could arise if states were clustered from a replica exchange or brute force molecular dynamics simulation where important transition state regions were under-represented in the sampling.

If an inappropriate choice of collective variables is made, as in Fig. 5(c) where only the x-coordinate is used to define states, the Markov matrices constructed also do not converge. Rough estimates in this case predicted timescales that underestimated the relaxation timescales by several orders of magnitude. For the x-coordinate MM, initial configurations were picked with their y values randomly chosen in the interval from $y=-1$ to $y=3$.

IV. DISCUSSION AND CONCLUSION

We have proposed an efficient method for constructing a MM by defining states along a transition path between stable states. First, a set of collective variables are chosen to describe the transition of interest. Second, an initial pathway is determined between two stable states of interest and refined using the string method with swarm of trajectories. If necessary, additional points are added within each basin by running free trajectories. Finally, the points obtained are taken

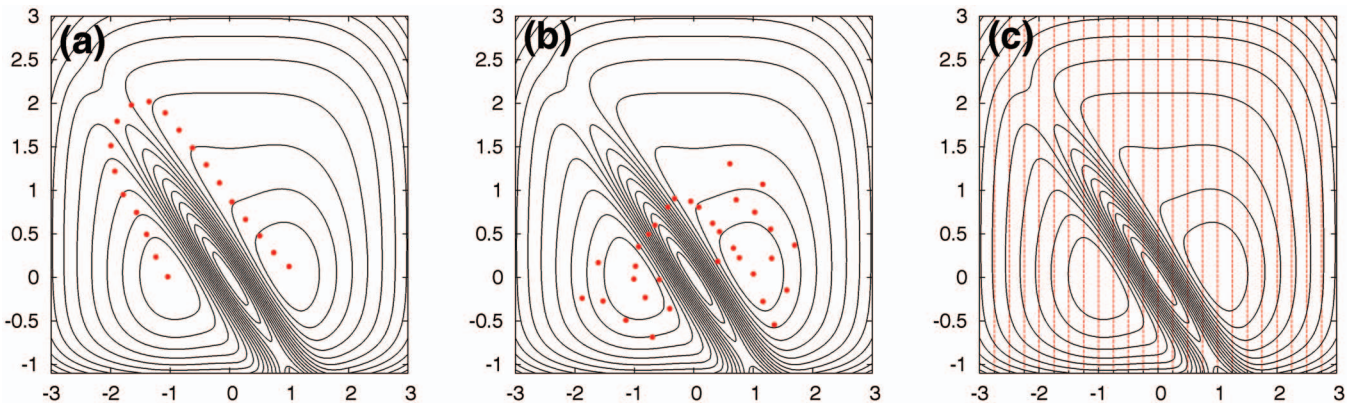

FIG. 5. (Color) Alternative MM state definitions. (a) MMs defined with only points along the relaxed pathway systematically underestimate the timescale by 20%–30%. (b) Using points along a badly relaxed pathway did not yield converged MMs [i.e., there was no plateau in $\tau(\Delta t)$ as a function of Δt]. Any timescales extracted from the resulting model overestimated the timescales by orders of magnitudes. Similarly, (c) using an inappropriate choice of variables to define the MM—in this case, using only the x-coordinate—also yielded an unconverged MM with rough estimates of timescales several orders of magnitude faster than the true timescales.

as centroids defining regions in a MM and short trajectories are initiated from each of those regions to construct a transition probability matrix.

Using a model system, we showed that this method can accurately predict timescales and free energies with short independent trajectories which can be several orders of magnitude shorter than the timescale of interest. Even when the aggregate simulation time is taken into account, the MM still represents a much more efficient use of simulation time than simply running a long brute force simulation of the same length. We have also shown that care needs to be taken when choosing order parameters and finding an appropriately converged pathway when constructing MMs. Applications of this method to fully atomistic protein systems are underway.

From a practical point of view, constructing MMs from pathways refined using the string method with swarms of trajectories is a natural extension of the programs and scripts used to implement the swarm method itself. Once the pathway has converged, longer trajectories, instead of the short swarm trajectories used to refine the path, can be run from each state and used to construct a MM.

One could argue that defining states along paths limits the amount of phase space that can be explored, whereas complex conformational changes of biomolecules may occur via many pathways through multiple stable states. This argument would seem to imply that using replica exchange or a high temperature dynamics run to find initial configurations would allow for a better coverage of states which are unbiased. The efficacy of these methods on, e.g., explictly solvated membrane proteins, however, is unclear.

Even if an enhanced sampling method is successfully applied, it would still be difficult to identify the location of important regions such as dynamical bottlenecks. For instance, any canonical sampling method such as replica exchange will visit these regions only in proportion to their Boltzmann weight, making the probability of sampling transition regions exponentially unlikely relative to configurations in stable basins. Clustering of the configurations generated by such methods would tend to extract centroids corresponding to a set of states that are visited frequently. The states defined by the centroids thus constructed are actually designed to point to metastable configurations, rather than the rarely occurring configurations in the transition regions. As a consequence, the rarely occurring configurations are located at the boundaries between the centroids (i.e., the walls of the Voronoi cells), and are thus poorly resolved by the collection of states used to build the MM. Although biasing methods to focus the sampling on a specific region of phase space exist,[7] it is still a challenge to localize transition state regions in a complex conformational transition *a priori*. Including these transition regions that are visited rarely, however, is necessary for building an accurate and efficient "coarse-grained" MM description.

In contrast, the pathway defined by the string method guarantees a uniform (non-Boltzmann) representation of all relevant configurations needed to account for the complete transition. It has also been shown previously that, given the right choice of reaction coordinates, the string method yields isocommittor pathways, which localize transition state regions.[21–23] Moreover, methods such as the finite temperature string method[22,31] which allows the sampling of fluctuations away from the dynamical transition tube can be used, in principle, to find alternative pathways and metastable states which are off path.

Even if multiple states are known from experiments, it would still be efficient to construct a MM by finding multiple pathways between each pair of states. The MM methodology, itself, is very general and does not require a set of states which are well ordered along a path. Indeed, given a MM constructed over a set of states, quantities such as committors and likely transition paths can be determined allowing one to infer the physically relevant connectivity of the states *a posteriori*.

We are not the first to propose constructing MMs along pathways. Singhal *et al.*[9] proposed to estimate transition probabilities from a transition path sampling[20] (TPS) calculation. Our idea is similar in spirit to their method. In diffusive systems with many metastable minima, however, the ability of TPS to sample paths efficiently becomes compromised. String-based methods, on the other hand, have no difficulties in diffusive systems with many stable intermediates.[21,23,31,32] The milestoning and partial path transition interface sampling algorithms also exploit the microscopic dynamics along a pathway coordinate, although they are geared toward the construction of non-Markovian stochastic models.[10,33] Although the present framework based on the string method could also be extended to account for non-Markovian behavior, here we have chosen to map the microscopic dynamics onto a MM. Future application should clarify the circumstances under which this is valid. Nonetheless, it is worth noting that the formal underpinning of the string method with swarms-of-trajectories algorithm is provided by an overdamped Brownian dynamics propagation of the collective variables,[21] which is perhaps more consistent with the choice of a MM.

In all the MMs constructed in this work, several independent short trajectories were run from the centroid position in each region with random initial Gaussian velocities. Formally, p_{ij} can be defined as[7]

$$p_{ij}(\Delta t) = \frac{\int d\boldsymbol{r}(0) e^{-\beta U(\boldsymbol{r}(0))} h_i(\boldsymbol{r}(0)) h_j(\boldsymbol{r}(\Delta t))}{\int d\boldsymbol{r}\ \ e^{-\beta U(\boldsymbol{r})} h_i(\boldsymbol{r})}. \tag{9}$$

Strictly speaking, the integral in the numerator is taken over a set of initial configurations which are Boltzmann distributed. In practice, however, after a transient lag time during free simulations, trajectories can adequately sample the relevant parts of the underlying free energy surface in each of the regions. This is shown explicitly by the agreement between the f_i predicted from the MM and derived from numerical integration in Fig. 2(b). Short trajectories for MM construction were similarly initiated in previous work.[16,17]

Lastly, we have not addressed the question of how many states should be used in a given MM, i.e., no prescription was given for choosing the density of centroids along the string and within each basin. This will depend on the system under study. If states are spaced too closely, the dynamics may display some inertial effects and the model may be non-Markovian. On the other hand, if states are spaced too far

apart, constructing the MM will be computationally demanding and provide little savings in computational time when compared to a long brute force simulation. Recent developments, such as the automatic clustering methods of Chodera *et al.*,[30] which lump states together based on a kinetic metric, could be useful.

ACKNOWLEDGMENTS

We wish to acknowledge useful discussions with Deniz Sezer and Sanghyun Park and are also grateful to Janice Robertson and Wenxun Gan for comments on the manuscript. A.C.P. acknowledges NIH for a Kirchstein-NRSA postdoctoral fellowship (Grant No. 1F32GM083567-01). This work was supported by Grant No. MCB-0415784 from the National Science Foundation and by Grants No. GM62342 and CA93577 from the National Institute of Health.

APPENDIX: BAYESIAN ERROR ESTIMATES

When constructing a MM, the quantity which is directly measured in a simulation is the number of transitions $z_{ij}(\Delta t)$ from state i to state j in a time Δt. Defining the total number of transitions out of state i as $n_i(\Delta t)=\Sigma_{j=1}^{K} z_{ij}(\Delta t)$, then $z_{ij}(\Delta t)/n_i(\Delta t)$, strictly speaking, is a (maximum likelihood) estimate $p_{ij}^{\mathrm{est}}(\Delta t)$ of the true transition probability $p_{ij}(\Delta t)$ (for clarity and notational convenience, we drop the explicit dependence on lag time Δt in the remaining discussion).

To determine errors in p_{ij}^{est} from finite sampling, we use a Bayesian analysis method described in Hinrichs and Pande[24,25] and also used by Park *et al.*[34] For another Bayesian based MM error analysis method, see Sriraman *et al.*[13] A conventional error analysis would involve sampling different transition count matrices z_{ij} generated from many independent simulations or from subsets of a larger set of data and then calculating averages and standard deviations over all these possible realizations of z_{ij}.[35] Instead, a Bayesian analysis asks, given a particular set of transition counts observed during the simulation, what is the distribution of possible transition matrices p_{ij}^{est} that could arise from the data.

Here, the distribution of transition counts follows a multinomial distribution such that the probability of observing a particular row vector, $\boldsymbol{z_i}=(z_{i1},z_{i2},\ \ldots,z_{iK})$, of the matrix z_{ij}, given the set of transition probabilities, $\boldsymbol{p_i}=(p_{i1},p_{i2},\ \ldots,p_{iK})$, is

$$P(\boldsymbol{z_i}|\boldsymbol{p_i}) \propto p_{i1}^{z_{i1}} p_{i2}^{z_{i2}} \cdots p_{iK}^{z_{iK}}. \tag{A1}$$

To estimate errors, however, the distribution of interest is $P(\boldsymbol{p_i}|\boldsymbol{z_i})$, the probability of a set of transition probabilities given an observed set of transition counts. This posterior distribution can be found using Bayes' theorem,

$$P(\boldsymbol{p_i}|\boldsymbol{z_i}) \propto P(\boldsymbol{z_i}|\boldsymbol{p_i})P(\boldsymbol{p_i}), \tag{A2}$$

where $P(\boldsymbol{p_i})$ is a distribution over transition probability vectors reflecting a prior belief in their values before the observation of any data.

To proceed, we follow Hinrichs and Pande[25] and choose the prior to be a Dirichlet distribution. In this way, the posterior $P(\boldsymbol{p_i}|\boldsymbol{z_i})$ will also be a Dirichlet distribution (i.e., the Dirichlet distribution is the conjugate prior to the multinomial distribution). A Dirichlet distribution for an arbitrary vector $\boldsymbol{p}$ and a set of parameters $\boldsymbol{\alpha}$ is given by

$$P(\boldsymbol{p};\boldsymbol{\alpha}) = \frac{1}{B(\boldsymbol{\alpha})} \prod_{i=1}^{K} p_i^{\alpha_i - 1}, \tag{A3}$$

where the normalization constant $B(\boldsymbol{\alpha})$ can be expressed in terms of the gamma function Γ as $\Pi_{i=1}^{K}\Gamma(\alpha_i)/\Gamma(\Sigma_{i=1}^{K}\alpha_i)$. Here, we choose a uniform Dirichlet distribution for the prior $P(\boldsymbol{p_i})$, such that $\alpha_{ij}=1/K$. In general, the choice of prior will affect the estimation of errors. As the sampling of the number of transition counts z_{ij} increases, however, the error estimates will depend less and less on the exact choice of a prior distribution.

With these preliminaries aside, we can now sample directly from $P(\boldsymbol{p_i}|\boldsymbol{z_i})$ using Eq. (A2), to obtain possible realizations of the transition probability matrix given a set of observed transition counts. From these different realizations, distributions of quantities of interest such as τ and f_i can be ascertained. For the error estimates in this work, we took 100–1000 samples of p_{ij} for each value of interest. Reported error bars indicate 95% credibility intervals around medians. All routines for random number generation, sampling from a Dirichlet distribution and linear algebra, were taken from the GNU scientific libraries.[36] For more details on this Bayesian error analysis method, in particular, concerning ways to sample error more efficiently, refer to Refs. 24 and 25.

[1] B. Hille, *Ion Channels of Excitable Membranes*, 3rd ed. (Sinauer, Sunderland, MA, 2001).
[2] B. F. Volkman, D. Lipson, D. E. Wemmer, and D. Kern, Science **291**, 2429 (2001).
[3] D. Chandler, J. Chem. Phys. **68**, 2959 (1978).
[4] A. F. Voter, Phys. Rev. B **57**, R13985 (1998).
[5] T. S. van Erp, D. Moroni, and P. G. Bolhuis, J. Chem. Phys. **118**, 7762 (2003).
[6] G. Hummer and I. G. Kevrekidis, J. Chem. Phys. **118**, 10762 (2003).
[7] W. C. Swope, J. W. Pitera, and F. Suits, J. Phys. Chem. B **108**, 6571 (2004).
[8] W. C. Swope, J. W. Pitera, F. Suits, M. Pitman, M. Eleftheriou, B. G. Fitch, R. S. Germain, A. Rayshubski, T. J. C. Ward, Y. Zhestkov, and R. Zhou, J. Phys. Chem. B **108**, 6582 (2004).
[9] N. Singhal, C. D. Snow, and V. S. Pande, J. Chem. Phys. **121**, 415 (2004).
[10] A. K. Faradjian and R. Elber, J. Chem. Phys. **120**, 10880 (2004).
[11] N.-V. Buchete and G. Hummer, Phys. Rev. E **77**, 030902 (2008).
[12] G. Jayachandran, V. Vishal, and V. S. Pande, J. Chem. Phys. **124**, 164902 (2006).
[13] S. Sriraman, I. Kevrekidis, and G. Hummer, J. Phys. Chem. B **109**, 6479 (2005).
[14] P. M. Kasson and V. S. Pande, PLOS Comput. Biol. **3**, e220 (2007).
[15] D. Sezer, J. Freed, and B. Roux, J. Phys. Chem. B **112**, 5755 (2008).
[16] S. P. Elmer, S. Park, and V. S. Pande, J. Chem. Phys. **123**, 114902 (2005).
[17] S. P. Elmer, S. Park, and V. S. Pande, J. Chem. Phys. **123**, 114903 (2005).
[18] R. Du, V. S. Pande, A. Y. Grosberg, T. Tanaka, and E. S. Shakhnovich, J. Chem. Phys. **108**, 334 (1998).
[19] P. Geissler, C. Dellago, and D. Chandler, J. Phys. Chem. B **103**, 3706 (1999).
[20] P. G. Bolhuis, C. Dellago, D. Chandler, and P. Geissler, Annu. Rev. Phys. Chem. **53**, 291 (2002).
[21] A. Pan, D. Sezer, and B. Roux, J. Phys. Chem. B **112**, 3432 (2008).
[22] W. Ren, E. Vanden-Eijnden, P. Maragakis, and W. E., J. Chem. Phys. **123**, 134109 (2005).

[23] L. Maragliano, A. Fischer, E. Vanden-Eijnden, and G. Ciccotti, J. Chem. Phys. **125**, 024106 (2006).
[24] N. Singhal and V. S. Pande, J. Chem. Phys. **123**, 204909 (2005).
[25] N. S. Hinrichs and V. S. Pande, J. Chem. Phys. **126**, 244101 (2007).
[26] E. Vanden-Eijnden, *Transition Path Theory*, Computer Simulations in Condensed Matter: From Materials to Chemical Biology Vol. 2 (Springer, New York, 2006), p. 439.
[27] M. P. Allen and D. J. Tildesley, *Computer Simulation of Liquids* (Oxford University Press, New York, 1987).
[28] See http://www.science.uva.nl/~bolhuis/tps/content/exercise.pdf for more information about this potential energy surface.
[29] B. R. Brooks, R. E. Bruccoleri, B. D. Olafson, D. J. States, S. Swaminathan, and M. Karplus, J. Comput. Chem. **4**, 187 (1983).
[30] J. D. Chodera, N. Singhal, V. S. Pande, K. A. Dill, and W. C. Swope, J. Chem. Phys. **126**, 155101 (2007).
[31] W. E, W. Ren, and E. Vanden-Eijnden, J. Phys. Chem. B **109**, 6688 (2005).
[32] W. E, W. Ren, and E. Vanden-Eijnden, Phys. Rev. B **66**, 052301 (2002).
[33] D. Moroni, P. G. Bolhuis, and T. S. van Erp, J. Chem. Phys. **120**, 4055 (2004).
[34] S. Park, T. E. Klein, and V. S. Pande, Biophys. J. **93**, 4108 (2007).
[35] J. D. Chodera, W. C. Swope, J. W. Pitera, and K. A. Dill, Multiscale Model. Simul. **5**, 1214 (2006).
[36] M. Galassi, J. Davies, J. Theiler, B. Gough, G. Jungman, M. Booth, and F. Rossi, *GNU Scientific Library Reference Manual*, 2nd ed. (Network Theory, Bristol, UK, 2006), http://www.gnu.org/software/gsl/.