Free-energy landscapes and thermodynamic parameters of complex molecules from nonequilibrium simulation trajectories

Prem P. Chapagain,* Bernard S. Gerstman, Yuba R. Bhandari, and Dipak Rimal
Department of Physics, Florida International University, Miami, Florida 33199, USA
(Received 23 July 2010; revised manuscript received 20 December 2010; published 7 June 2011)

Thermodynamic parameters such as free energies and heat capacities are important quantities for understanding processes involving structural transitions in complex molecules such as proteins. Computational investigations provide simulated data that can be used for calculating thermodynamic parameters. However, calculations give accurate results only if the simulations sample all of configuration space with the appropriate temperature-dependent Boltzmann equilibrium probabilities. For many systems, truly comprehensive sampling of configuration space is not computationally feasible. We present an approximation technique for the calculations that will give accurate values for thermodynamic parameters when the data is incomplete. Our work is applicable to systems in which there are two distinct, important regions of configuration space that must be sampled. Importantly, the results are also valid when the system is more complex than two-state systems. Transition pathways that involve intermediate configurations between two stable regions are allowed in this treatment, and therefore the results are valid for multistate systems.

DOI: 10.1103/PhysRevE.83.061905 PACS number(s): 87.10.Rt, 87.14.et, 87.15.Cc, 05.70.Fh

I. INTRODUCTION

Thermodynamic parameters such as free energies and heat capacities are important quantities for understanding processes involving structural changes in complex molecules such as proteins. The large number of degrees of freedom and the many different substates in the energy landscape result in a rough energy landscape [1,2] that can be characterized as "complex" [3,4]. The conformational space of complex systems is enormously large and grows exponentially with the size of the system. An example is the increase in configurations of a peptide chain as a function of the number of amino acid residues in the chain [5]. Computational investigations provide simulated data that can be used for calculating thermodynamic parameters. The validity of these calculated results depends on the approach used to take the simulated data and calculate numerical values of the thermodynamic properties. Calculations give accurate results if all of the configuration space is sampled with the appropriate temperature-dependent Boltzmann equilibrium probabilities. For many systems, a truly comprehensive sampling of configuration space is not possible. A favorable situation for comprehensive sampling is at temperatures near the transition temperature between two regions. Under this condition, the probability is approximately equal for the system to be in either region of configuration space and both regions are adequately sampled. In contrast, for relatively high or low temperatures, one region of configuration space will have a low probability to be sampled. If computer power is unlimited, this would not be a problem: Eventually the computer simulation will leave the more probable region and transition to the other region. For a long enough simulation, even the less likely region would eventually be properly sampled. However, due to limitations in computer power, at simulation temperatures far from the transition temperature it may become unfeasible to allow the computer to run a simulation long enough for the system to make the multiple back-and-forth transitions necessary to sample configuration space with the correct equilibrium probabilities.

In order to understand the behavior of a complex system, various thermodynamic parameters are useful. To obtain these parameters in complex systems, a variety of computational simulation methods are used. All-atom molecular dynamics (MD) simulations provide the most detailed information about molecular processes but are limited because of the computational cost. Explicit solvent, all-atom MD simulations of proteins for up to a millisecond time scale have recently been performed on a special-purpose machine [6]. However, such simulations are still computationally costly for most proteins, and the MD simulations are restricted in the amount of configuration space that can be sampled to regions involving structural changes of only small segments of the entire molecule. Coarse-grained MD simulations and reduced-model Monte Carlo (MC) simulations provide less atomistic detail but have the benefit of being able to sample greater regions of the energy landscape for longer processes. Simulations of this type can investigate large-scale structural changes such as folding of a protein molecule.

Limitations in computer power mean that for glassy systems with rough landscapes, or large molecules with many degrees of freedom, even the MC and coarse-grained MD simulations are not able to sample all the regions of configuration-space without the help of clever techniques [7–11]. Enhanced sampling methods that facilitate equilibrium sampling of configuration space have been developed, such as multicanonical sampling [12,13], umbrella sampling [14,15], transition path sampling [16,17], and entropic sampling [18]. With proper weighting functions, these sampling methods can provide accurate estimates of equilibrium thermodynamic parameters such as free energy and heat capacity. Another method based on replica exchange (RE) [19] has emerged as a powerful sampling method. When applied to MC simulations it is known as the replica exchange Monte Carlo (REMC) or parallel tempering method, and in MD simulations it is known

*Corresponding author: chapagap@fiu.edu

as the replica exchange molecular dynamics (REMD) method [20]. The REMD method has proven especially valuable and is increasingly used to overcome sampling problems involving multiple minima and barrier crossing in conventional MD simulations [21,22]. The replica exchange method can also be applied in conjunction with other sampling methods for increased efficiency in equilibrium sampling for free-energy calculations [23]. However, in some situations, such as in simulations of large molecules in an explicit solvent, the RE method may not be efficient without carefully applied modifications [24,25]. Also, the RE method is not able to supply kinetic information, such as median first passage times (MFPTs).

Using many short simulation trajectories, instead of one long simulation, has proven very useful for efficient sampling. A powerful method based on such strategy is the Markov state model (MSM) [26], which has been applied to facilitate long time-scale MD simulations [27–29]. In this paper, we present a compensation method that allows simulated data obtained from many short, incomplete simulations to be used to accurately calculate thermodynamic parameters for any type of system with two distinct metastable configuration regions. This method conceptually resembles the sampling strategy of the MSM, but is applied to construct free-energy landscapes from MC trajectories.

Another method for estimating free-energy differences between macrostates is described by Jarzynski [30,31]. Jarzynski showed that in situations in which the work performed while a molecule transitions between two states can be determined, the free-energy difference between the two states can be obtained by averaging the exponentially weighted nonequilibrium work trajectories. Though not relevant to the investigations described here, this powerful method has proven particularly useful in force-extension or mechanical unfolding experiments as well as simulations [32–35], where work is done to perturb the system to transition from one state to other. In contrast, our method can be useful in situations where estimating the work done is not feasible.

We show how nonequilibrium data that do not comprehensively sample all of configuration space can be used in calculations to provide accurate thermodynamic and kinetic parameters using an approximation technique that is applicable to systems in which there are two distinct, important regions of configuration space that must be sampled. Transition pathways that involve intermediate configurations between two stable regions are allowed in this treatment, and therefore the results are valid for multistate systems if the intermediate state regions of configuration space are also appropriately sampled. The paper is organized as follows. In Sec. II we describe our compensation method of combining data from nonequilibrium studies to produce time series of energy and other structural data that can be used to calculate thermodynamic parameters. In Sec. III we describe the computer model we use to simulate protein dynamics and generate data. In Sec. IV we present results. Heat capacities and free energies are calculated and presented for two different proteins. One protein is small enough so that we can perform comprehensive sampling of all of configuration space at any temperature. The thermodynamic parameters calculated from these comprehensive equilibrium trajectories are treated as the "correct" reference values. We also use our method on an incomplete data set for the same protein to calculate thermodynamic parameters, and these results are compared to the correct reference values to show the accuracy of our compensation method. Next, in order to show the wide applicability of our method, we apply the compensation method to calculate thermodynamic parameters for a protein that is too large to permit comprehensive sampling of all of configuration space. We conclude with a summary of the results and their significance in Sec. V.

II. COMPENSATING FOR NONEQUILIBRIUM DATA

A. Generating time-series trajectories

The method is applicable to systems that have two distinct, important regions of configuration space. For proteins, one macrostate region is composed of compact states that are capable of performing the intended biological function. The states of the molecule in this region are collectively known as the "native (N) state" configurations. Nonfunctioning configurations, such as extended states of the molecule are called non-native, unfolded (U), or random coil states. Computational simulations are initialized to commence from either an N or U state. The configurations within a macrostate region make rapid transitions among each other and, therefore, to properly sample a region, it does not matter precisely which microstate (configuration) is used to initiate a simulation. The same is also true for other macrostate regions of configuration space, such as the folded region.

The probability to transition to the other region depends on various factors such as the height of the transition barriers and the temperature of the simulation. Though at high temperatures barriers are easy to cross, nevertheless, for a large, complex system the transition to the lower-energy N region may be unlikely because the high-energy, high-entropy U region may have an enormous number of substates and the transition barrier is never approached. For complex systems, the free-energy landscape $F = E - TS$ is a crucial parameter for understanding the behavior of the molecule.

If a simulation is run under appropriate conditions, such as near the transition temperature of the system, equilibrium trajectories of the behavior of the system are obtainable in which all regions of configuration space are appropriately sampled. An equilibrium trajectory may resemble the one in Fig. 1 in which we plot a time series of the energy of a protein system. The protein system is the GCN4-p1 leucine zipper dimer and the computer simulations used a MC algorithm. Later, we will describe in more detail both the protein and the simulation technique. To generate Fig. 1, the system is initialized to be in a high-energy (~ -20 kcal/mol) U state. The temperature used in the simulation was near the system's transition temperature and therefore the system makes multiple transitions from U configurations (-80 kcal/mol$< E_U < 0$) to the biologically functional N configurations (-160 kcal/mol$< E_N < -120$ kcal/mol). It can also be seen that the system samples intermediate states in the region between -120 and -80 kcal/mol, showing that this is a multistate system. For the simulation temperature used in Fig. 1, the system spends a substantial fraction of the time in both U and N configurations. All of configuration space is

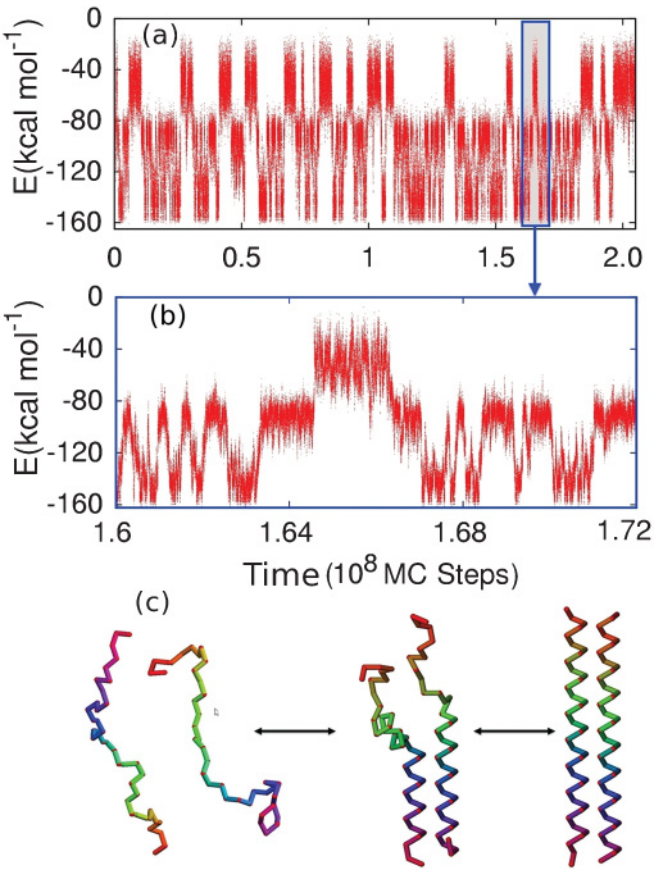

FIG. 1. (Color online) (a) Energy time series for folding and unfolding dynamics of the leucine zipper dimer. This simulation was run at a temperature that allows multiple transitions between low-energy native states and high-energy unfolded states. (b) Magnified view of a section in the MC time series which clearly shows the multistate nature of the transition. (c) Transitions occur between the high-energy disorganized unfolded state, a midenergy transition state, and a low-energy organized native-state dimer.

adequately sampled, and this trajectory can be used to calculate accurate values for important thermodynamic parameters such as heat capacity and free energy.

Figure 1 displays the time series of an energy trajectory that includes multiple folding and unfolding transitions. However, short simulations can also be used for calculating kinetic and thermodynamic parameters, as is done in the MSM [26]. In MSM, multiple short trajectories are created during MD simulations by initiating new runs at random times along a simulation that started from an initial state such as U or N. Trajectories are used in calculating parameters such as the mean folding time or median first passage time (MFPT), and p-fold only if they end up in the desired macrostate region of configuration space (U or N) within the assigned cutoff time. This method efficiently samples folding transitions. In our method, we start our simulations from either U or N and stop a simulation when the molecule reaches the other macrostate, or when the cutoff time is reached. This can be viewed as a modified subset of MSM that allows combining trajectories for free-energy calculations.

Rather than using one long trajectory, we combine a series of shorter simulations to create a time series such as the one shown in Fig. 1. A group of 100 simulation runs can all be started in the same high-energy U configuration to investigate the dynamics of the folding process. The difference between simulations is that each will be supplied with a different sequence of random numbers that are used to simulate the behavior of the system and each simulation will generate a different trajectory through configuration space. Each simulation is run independently of the others, and each simulation is stopped when it makes the folding transition to an N configuration. Likewise, to investigate unfolding, 100 simulations can all be started from the same low-energy folded N configuration, but each with a different sequence of random numbers. Each of these simulations will run independently, and each one separately stopped when it makes the unfolding transition to a U configuration. If the energy time series of the 100 folding simulations from U→N are appended to the 100 unfolding simulations from N→U, the result will be a long time-series trajectory of the type displayed in Fig. 1(a). This method of using multiple short simulation runs to generate a long time-series trajectory has several advantages [17,26] over running a single, very long run with the hope of obtaining multiple folding and unfolding transitions. Kinetic properties such as the characteristic time for folding require averaging over many U→N folding attempts. Multiple independent folding runs allow the averaging to be carried out more easily than disentangling the information from a single long run with many transitions. Also, occasionally problems occur during a simulation that are artifacts of the computer model and not physically relevant. An example of this is when a peptide chain becomes stuck in a computer configuration from which a real protein could escape. If this occurs during a single long simulation that is intended to replicate many folding and unfolding events, the information obtained does not represent real dynamics. In contrast, if a single short simulation gets stuck in an unphysical configurational trap, it only affects a small fraction of the total simulation time and has an insignificant effect on the results. We show that the remaining simulations can be used for the next stage of analysis. Because our trajectories through configuration space are determined by random numbers, each time we start another simulation, the time sequence of steps through configuration space is completely independent of any other simulation. This allows a realistic sampling of configurational space. For these reasons, we create time-series trajectories by combining multiple short runs.

B. Density of states and average folding time

The aim of computational results is to obtain accurate values for quantities such as the true average U→N folding time $\bar{\tau}_f$, representing the theoretical average that would be obtained if it was possible to include all trajectories through the entire energy landscape that take the chain from an unfolded configuration to the folded, native state. Similarly, the true average N→U unfolding time $\bar{\tau}_u$ is defined for all trajectories that take the chain from the folded state to an unfolded configuration. Both $\bar{\tau}_f(T)$ and $\bar{\tau}_u(T)$ are temperature dependent. This can be seen by use of the master equation [36],

$$\frac{dP_f}{dt} = -\frac{dP_u}{dt} = k_f P_u - k_u P_f, \tag{1}$$

where P_f is the probability to be in the folded, native-state region of configuration space, P_u is the probability to be in an unfolded configuration, k_f is the average rate for folding, and k_u is the average rate of unfolding. In equilibrium at any temperature, $k_f P_u - k_u P_f = 0$ and $k_f/k_u = P_f/P_u$. Since P_f and P_u are each temperature dependent, then so are k_f and k_u, and likewise $\bar{\tau}_f(T) = 1/k_f(T)$ and $\bar{\tau}_u(T) = 1/k_u(T)$ are also temperature dependent.

The parameters $\bar{\tau}_f$ and $\bar{\tau}_u$ can be theoretically calculated if the configurational landscape is known in complete detail. In order to relate the underlying topology of the landscape to observable kinetics, ideally we would like to know the folding time as a function of temperature for each possible unfolded configuration, $\tau_f = \tau_f(\vec{x}_u, T)$. Here, $\vec{x}_u$ is a multidimensional vector representing the structural configuration of an unfolded chain. Because of the complex nature of the energy landscape, different unfolded configurations $\vec{x}_u$ may have the same τ_f, i.e., the mapping of the set of $\vec{x}_u$ onto the set of τ_f is not a one-to-one mapping. The distribution in folding times can be represented as a density of states over all configurations, each with a specific but not unique folding time, $D(\vec{x}_u(\tau_f), T)$. At a given temperature T, all unfolded configurations with a folding time between a specific τ_f and $\tau_f + d\tau_f$ contribute to a density of states $D(\tau_f, T)$:

$$D(\tau_f; T) = \int_{\vec{x}_u} D(\vec{x}_u(\tau_f'); T)\delta(\tau_f' - \tau_f)\, d\vec{x}_u. \tag{2}$$

This leads to an expression relating $D(\vec{x}_u(\tau_f); T)$ to the experimentally observable $\bar{\tau}_f$. At a given temperature,

$$\begin{aligned}\bar{\tau}_f(T) &= \int_0^\infty \tau_f D(\tau_f; T)\, d\tau_f \\ &= \int_0^\infty \int_{\vec{x}_u} \tau_f D(\vec{x}_u(\tau_f'); T)\delta(\tau_f' - \tau_f)\, d\vec{x}_u\, d\tau_f. \end{aligned}\tag{3}$$

A similar expression can be written for the theoretical value $\bar{\tau}_u$ representing the transition from N→U,

$$\begin{aligned}\bar{\tau}_u(T) &= \int_0^\infty \tau_u D(\tau_u; T)\, d\tau_u \\ &= \int_0^\infty \int_{\vec{x}_f} \tau_u D(\vec{x}_f(\tau_u'); T)\delta(\tau_u' - \tau_u)\, d\vec{x}_f\, d\tau_u. \end{aligned}\tag{4}$$

The correct value of $\bar{\tau}_f$ can be obtained from computational simulations if an infinite number of different simulations are performed in order to exhaustively sample configuration-space to reproduce $D(\vec{x}_u(\tau_f); T)$ and each simulation is allowed to run as long as necessary to successfully complete the structural transition. Practical considerations limit the amount of computations that can be performed. In our computer investigations, at each temperature we start N_0 simulations ($N_0 = 100$) in a native configuration, and an equal number in an unfolded, random coil configuration. If started in an unfolded configuration, a successful folding simulation run is terminated when the chain first reaches a folded native-state configuration, and its τ_f is recorded. The results of the simulations provide a computationally determined numerical estimate for the theoretical $\bar{\tau}_f$ of Eq. (3), $\bar{\tau}_f = \frac{1}{N_0}\sum_{i=1}^{N_0} \tau_{fi}$. Though we only discuss the details of the analysis for the folding process, the same method can be applied at lower temperatures to the unfolding process to determine the average unfolding time $\bar{\tau}_u = \frac{1}{N_0}\sum_{i=1}^{N_0} \tau_{ui}$.

C. Nonequilibrium simulations: Problems due to computational limitations

An equilibrium time-series trajectory of folding and unfolding transitions can be generated by appending the time series of all the folding runs to the time series from all the unfolding runs. If all N_0 folding and N_0 unfolding runs are successful, the total length of the full equilibrium trajectory is

$$\tau = \sum_{i=1}^{N_0} \tau_{fi} + \sum_{i=1}^{N_0} \tau_{ui} = (\bar{\tau}_f + \bar{\tau}_u)N_0. \tag{5}$$

Unfortunately, at many temperatures of interest, it is not possible to create such an equilibrium trajectory. At high temperatures, some of the simulations may take such a long time to fold that it is not computationally feasible to follow the folding to completion. At low temperatures, the same situation can occur for unfolding runs. To avoid such endless simulations in computational investigations, a simulation will be terminated if the simulation is unsuccessful in reaching the other condition after a user-defined cutoff, or end time, τ_e. The proper way to compensate for these unsuccessful runs is the topic of this paper.

D. Survival probability

At high temperatures, all N_0 unfolding simulations that start in the native state will be successful, but only a fraction n/N_0 of folding simulations will be successful in a time less than τ_e. Under these conditions, it is still possible to estimate $\bar{\tau}_f$ from survival probability [37–39]. The survival probability is the fraction of simulations that are unsuccessful in folding and remain in the unfolded state [40]. The survival probability to remain unfolded after a simulation time t is denoted by $p(t) = 1 - n(t)/N_0$, where n is the number of successfully folded simulation runs.

For the special case of a process that can be described by single exponential kinetics, the survival function can be expressed as $p(t) = 1 - n(t)/N_0 = \exp(-t/\bar{\tau}_f)$, which allows $\bar{\tau}_f$ to be straightforwardly calculated from the simulation results by counting the number of successful runs n after any cutoff end time, $t = \tau_e$.

E. Compensating for incomplete sampling

Under temperature conditions in which comprehensive sampling occurs and all folding and all unfolding runs are successful, Eq. (5) shows that an equilibrium time-series trajectory can be composed by using an equal number of folding and unfolding simulations. We now show that the total length of an equilibrium trajectory can be expressed more generally in terms of the number of folding versus the number of unfolding trajectories that should be combined. When conditions are used that prevent comprehensive sampling and 100% success for all runs, these numbers may be different.

At high temperature, all N_0 unfolding simulations are successful in making the transition, but many folding runs are terminated at the artificial cutoff end time τ_e without folding. If

we combine N_0 unfolding simulations with the N_0 attempted folding simulations in which only n are successful, the total length of the nonequilibrium trajectory is a function of the artificial cutoff end time τ_e,

$$\begin{aligned}\tau_{\rm ne} &= \sum_{i=1}^{n}\tau_{fi} + \sum_{n+1}^{N_0}\tau_e + \sum_{i=1}^{N_0}\tau_{ui} \\ &= n\bar{\tau}_{ff} + (N_0 - n)\tau_e + N_0\bar{\tau}_u, \end{aligned} \tag{6}$$

instead of Eq. (5). Here, τ_{ne} refers to the length of a nonequilibrium trajectory that contains $N - n$ simulations that were not able to successfully complete their task within τ_e, and $\bar{\tau}_{ff}$ is the average folding time for the n successful folding simulations. In analogy with Eq. (3), $\bar{\tau}_{ff}$ can be expressed as $\bar{\tau}_{ff}(T) = \int_0^{\tau_e}\tau_f D(\tau_f;T)d\tau_f$. For the nonequilibrium (not all successful) time series, the total time spent by all N_0 simulations that are attempting to fold under unfavorable conditions is $\tau_{f,\rm ne} = n\bar{\tau}_{ff} + (N_0 - n)\tau_e$.

A nonequilibrium trajectory of total length τ_{ne} is not an equilibrium trajectory because the unfolded state is undersampled. Each of the $N_0 - n$ folding simulations that were unsuccessful in folding was still in the process of exploring the unfolded part of the energy landscape when it was artificially terminated at τ_e. For each unsuccessful folding simulation, the unfolded region of the landscape was undersampled by a time $\Delta\tau_{fi} = \tau_{fi} - \tau_e$, where τ_{fi} is the time at which the run would have folded if it had been allowed to continue for as long as necessary. Each of the $N_0 - n$ unsuccessful folding runs contribute to this undersampling, and the total deficit of time spent in the unfolded region compared to equilibrium sampling is

$$\Delta\tau_f = \sum_{i=n+1}^{N_0}\Delta\tau_{fi} = \sum_{i=n+1}^{N_0}(\tau_{fi} - \tau_e). \tag{7}$$

In order to compensate for the undersampling of the unfolded state, $\Delta\tau_f$ steps in the unfolded region of configuration space should be added to the nonequilibrium length $\tau_{f,ne}$ that is obtained directly from the simulations. Determining $\Delta\tau_f$ is the problem addressed in the paper. The reason for the complication is that for the $N_0 - n$ unsuccessful folding runs that are terminated at τ_e, we do not know their τ_{fi} to insert in Eq. (7). We now describe a practical method to estimate and compensate for this deficit $\Delta\tau_f$.

We define ΔN_f to be the unknown number of additional folding simulations that are needed to supply $\Delta\tau_f$. This rephrases the problem so that the question switches from estimating $\Delta\tau_f$ to how to determine an appropriate number for ΔN_f. Once ΔN_f is determined, the total length of the equilibrium trajectory will become

$$\tau = \tau_{\rm ne} + \Delta\tau_f = \left(\sum_{i=1}^{n}\tau_{fi} + \sum_{i=n+1}^{N_0}\tau_e + \sum_{i=1}^{N_0}\tau_{ui}\right) + \sum_{j=1}^{\Delta N_f}\tau_{fj}. \tag{8}$$

Because some folding runs are prematurely terminated after τ_e, Eq. (8) shows that for every N_0 unfolding trajectories, we must have $N_f = N_0 + \Delta N_f$ folding trajectories. The total number of unfolding and folding simulation runs that constitute the full equilibrium time-series trajectory is given by $N_{\rm eq} = N_0(u) + N_0(f) + \Delta N_f = 2N_0 + \Delta N_f$.

Since we are specifically interested in conditions in which the true $\bar{\tau}_f$ of Eq. (3) is not known, from a practical standpoint the only information available to estimate ΔN_f is the fraction of runs that successfully folded within the cutoff time τ_e, which is n/N_0. The simplest correction would be to assume that the $N_0 - n$ folding runs that did not fold would have necessitated much longer than τ_e to successfully fold. If true, then we should include not only the time series of these unsuccessful runs but also use the same τ_e to run the same number of new runs in addition, so that $\Delta N_f = N_0 - n$. This value of ΔN_f can be reexpressed in terms of the known survival function $p(\tau_e) = 1 - n(\tau_e)/N_0$, which gives $\Delta N_f = pN_0$.

However, we can expect the same probability of success for these additional folding runs, which means that a fraction $p(\tau_e)$ of these ΔN_f additional runs will also not be successful, and the unfolded region of configuration space will still be undersampled, though now by a smaller amount. We must add additional simulations to compensate for the remaining undersampling. Since the remaining undersampling is due to pN_0 simulations, the additional simulations require a correction that is second order with respect to the survival function p: we add an additional $p(pN_0) = p^2N_0$ runs. We continue with this iterative process, which leads to an equation that allows us to determine the number ΔN_f of additional folding simulations (not all successful) that we need to combine with the original N_0 (not all successful) folding simulations and the N_0 original (all successful) unfolding simulations so that both regions of the energy landscape are correctly sampled with the proper equilibrium ratio of time:

$$\begin{aligned} N_f &= N_0 + \Delta N_f = N_0 + p(\tau_e)N_0 + p(\tau_e)\,[p(\tau_e)N_0] + \cdots \\ &= N_0\{1 + p(\tau_e) + [p(\tau_e)]^2 + \cdots\} = N_0\frac{1}{1 - p(\tau_e)} \\ &= N_0\left(\frac{N_0}{n}\right). \end{aligned} \tag{9}$$

Equation (9) directly gives the number ΔN_f of additional folding simulations that are necessary to run in order to create an equilibrium trajectory:

$$\Delta N_f = N_f - N_0 = N_0\left(\frac{N_0}{n}\right) - N_0 = N_0\left(\frac{N_0}{n} - 1\right). \tag{10}$$

As a check on Eqs. (9) and (10), if we are at a temperature in which all N_0 folding simulations are successful, then $n = N_0$ and we get the correct results that $\Delta N_f = 0$, which means that an equilibrium trajectory requires an equal number of folding and unfolding simulations, $N_f = N_u = N_0$. At the other extreme, when $n \ll N_0$, we get the expected result that $\Delta N_f \gg N_0$.

Equations (9) and (10) give expressions for the number of additional folding runs that are necessary to create an equilibrium trajectory. The procedure can be made simpler to implement by noticing that Eq. (9) also gives the ratio

of unfolding versus folding runs $N_u/N_f = N_0/N_f$, that is required to create an equilibrium trajectory of folding and unfolding simulations that will have the correct temperature-dependent ratio of $\bar{\tau}_u/\bar{\tau}_f$ of Eq. (5). Since proper sampling requires the correct ratio of time spent in each region, if we maintain the same ratio of unfolding to folding simulations, $N_u/N_f = N_0/N_f$, but use a smaller number of each, we will continue to have an equilibrium trajectory. Equation (9) shows this ratio has a value of

$$\frac{N_u}{N_f} = \frac{N_0}{N_0\left(\frac{N_0}{n}\right)} = \frac{n}{N_0}. \tag{11}$$

We can keep this ratio the same by using all original $N_f = N_0$ folding simulations (not all successful), and only $N_u = n$ unfolding simulations (all successful). This means that the final $N_0 - n$ successful unfolding runs are discarded. This is preferred because we use runs that are all completed in the first set of simulations containing N_0 folding runs and N_0 unfolding runs, and it is not necessary to run additional simulations after this first set is completed. The same procedure can be used at low temperatures at which all the folding simulations are successful but not all of the unfolding simulations.

III. COMPUTER MODEL

Equation (11) is a simple expression that allows computer simulations of transitions of a system to be combined so that accurate values of thermodynamic parameters can be calculated. We now briefly describe the specific computational model that we employed to generate the results presented in the next section on two different protein systems. The model for one-chain simulations is described in more detail in Ref. [41] and for two-chain simulations in Refs. [42] and [43].

The computational model [44,45] uses an underlying cubic lattice, and includes separate degrees of freedom for an amino acid's backbone and sidechain. The location of an amino acid residue is defined by the position of its backbone. In addition, another lattice site is assigned for the sidechain. The orientation of a sidechain with respect to its backbone can vary, but is always constrained so that it gives left-handed chirality to the C_α, as is true with real amino acids. The lattice representation of the peptide bond allows protein secondary, tertiary, and quaternary structures to be effectively represented [46–48]. The volume of the simulation box was big enough to allow the chains the freedom to flip, bend, or rotate, either individually or as a dimer.

The computer simulations were performed by employing a MC algorithm which simulates the dynamics of the system by changing the internal configuration of each chain, as well as the relative separation and orientation between the two chains. A Metropolis test is incorporated to assure that the various configurations appear with the correct thermodynamic Boltzmann probability. Changes in the internal configuration of a chain are attempted through a combination of moves involving individual residues and multiresidue moves that allow all of configuration space to be accessed. For a two-chain system, translations and rotations of an entire chain that change the relative position and orientation of the chains with respect to each other are implemented based on Brownian motion theory. In the simulations, time is counted in MC steps. Each MC step includes a variety of moves involving individual amino acids, groups of amino acids, or an entire chain.

For any configuration of the two chains that occurs in a simulation, the energy of the system is calculated using a Hamiltonian of the form [41,43]

$$H = \sum_i a_i E_i. \tag{12}$$

There are several different energy terms in the Hamiltonian. Each term contributes to the total energy of the chain by switching the corresponding a_i from 0 to 1. Depending on their user-defined properties, two sidechains can interact through one of three terms: E(hydrophobic-hydrophobic), E(hydrophobic-hydrophilic), or E(hydrophilic-hydrophilic). Two backbones interact through E(backbone-backbone) if their active sites approach within a distance of 4 or less. This interaction represents the combined effects of hydrogen bonding, dipole interactions, and van der Waals interactions. If a residue finds itself in a secondary structural configuration that is the same as its user-defined propensity, the energy of the chain is lowered by a term representing the individual amino acid's local preference, E_L. If two amino acids in a row are in the same configuration preferred by the user-defined propensity, a cooperative or medium range energy E_M also lowers the energy of the chain.

All information for the initial configurations of the peptide chains, as well as the strengths of various interactions appearing in the Hamiltonian, and other parameters, are supplied by the user in an input file. For each configuration in a simulation, the interactions that contribute are determined based upon distances between amino acids and other criteria that are expressed through the a_i and explained in Refs. [42,43].

There are a variety of parameters that provide information on the configuration of the system $\vec{x}$. The total energy of the chains, including each chain's internal energy as well as interchain interactions, is denoted as E, and an example of a time series of E is plotted in Fig. 1. To show clearly that we are applying our compensation method to a system with intermediate transition states, Fig. 1(b) is an expanded view of a small section of Fig. 1(a). Figure 1(b) shows that the system includes intermediate-energy states. A picture of an intermediate state for the leucine zipper dimer is shown in Fig. 1(c).

For the leucine zipper, the intermediate states are defined as having three intact native interchain contacts. These three interchain contacts occur between the trigger regions of the two chains. In other complex systems, it may not be easy to define what is meant by an intermediate state. Fortunately, a precise definition of "intermediate state" is not important as long as the configurations in this region are properly sampled so that we can calculate accurate values for kinetic and thermodynamic parameters. To accomplish this sampling of intermediate states, we use an especially strict definition of "folded" state. A folding simulation that starts in a random coil configuration is considered to have successfully folded only if all native interhelical contacts have been made. This ensures that the intermediate states are sampled during a folding simulation. In addition, to ensure that intermediate states are sampled during an unfolding simulation, we define success for an unfolding simulation only when all native interhelical

contacts have been broken. Therefore, interhelical states which have some, but not all, native contacts made are sampled both during folding runs and unfolding runs.

In addition to energy time series, the computational simulations also produce time series of structural parameters. The secondary structure of each individual chain is monitored through the parameter q that supplies information on how many amino acids are in an α-helical native secondary structure. For the two-chain leucine zipper, the maximum value of q is 74 (37 from each chain). Native interchain interactions along the interface between the chains are monitored by Q, with $Q = 9$ as the maximum that occurs in the fully formed native dimer of Fig. 1(c). If the two chains interact, but are not properly aligned, the value can be $Q = 0$. The separation between the two chains is measured by d_{cm}, which is the distance between the centers of mass of the two chains. There are many non-native configurations of the chain that can have the same d_{cm}, and therefore d_{cm} is most valuable when used with other structural parameters and the chains' energy. The results from using a combination of parameters to investigate the dimerization process are presented.

In the next section we present results of calculations for heat capacity and free energy. The temperature profile of the heat capacity of a system that undergoes a structural transition can provide deep insight into the underlying dynamics. The heat capacity at constant volume can be calculated by using the fluctuations in the energy, $C_v = (\overline{E^2} - \bar{E}^2)/kT^2$. The most accurate way to calculate C_v at any T is to perform simulations at that specific T. The energy time series can be used to obtain both $\bar{E}$ and $\overline{E^2}$ for use in the heat capacity equation. Calculating C_v at many different temperatures requires time-consuming simulations at different temperatures. A computationally quicker approach for determining C_v at many different T is known as the histogram method [49,50], which allows the calculation of thermal averages for a range of temperatures from the trajectory of a single simulation at a single temperature. Free energy can be calculated for a specific configuration using the expression $F(x) = -kT \ln P(x)$, where $P(x)$ is the probability during a simulation that a chain will have a specific value of a structural characteristic x, such as E, q, or Q. $P(x)$ is calculated using $P(x) = m'(x)/m$, where m is the length of a simulation, and $m'(x)$ is the number of frames during the simulation in which a structural characteristic has a specific value x.

IV. RESULTS

We first apply the prescription of Eq. (11) to a system that is small enough so that we can also perform a comprehensive sampling of all of configuration space at any temperature to validate the method. The thermodynamic parameters calculated from the comprehensive equilibrium trajectories are treated as the "correct" reference values. We then rerun the simulations using τ_e that is small enough so that all simulations are not successful. We show that if not all simulations are successful, using all N_0 folding runs and all N_0 unfolding runs to calculate thermodynamic parameters gives highly inaccurate results. We then apply the compensation technique of Eq. (11) to the short τ_e nonequilibrium, incomplete data set for the same protein and then recalculate thermodynamic parameters. The results of these compensated calculations are compared to the correct reference values to show the accuracy of the estimation method. At the end of this section, in order to show the wide applicability of our method, we apply the compensation method to calculate thermodynamic parameters for a protein that is too large to permit comprehensive sampling of all of configuration space.

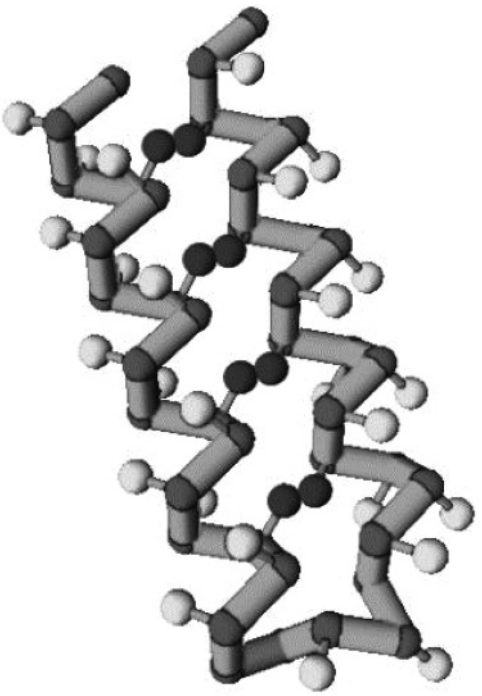

FIG. 2. Native structure of the two-helix bundle showing each helix and the sidechain interactions (dark circles) between the two helices.

The small system that can be investigated exhaustively is a single protein whose native state is a two-helix bundle. The native state is shown in Fig. 2 and is defined so that all four interhelical contacts are made. This can only occur when most, or all, of the amino acids in the helical sections have assumed a helical configuration. Because this protein is relatively small and has a simple native-state configuration, folding and unfolding transitions occur quickly relative to computational time scales. Therefore, over a large range of temperatures, equilibrium trajectories can be constructed in which all folding transitions are successful and all unfolding transitions are successful. In Fig. 3(a), at each simulation temperature the open circle is the heat capacity of the protein calculated from an energy time series that was composed of 100 successful folding simulations combined with 100 successful unfolding simulations. These open circles are treated as the "correct" reference values for comparison. For guidance, we also include a continuous curve in which the heat capacity at all temperatures is calculated using the histogram method from the simulations performed at a single temperature near the transition temperature $T_c \sim 352$ K.

In order to have 100% success for both folding and unfolding at all temperatures in Fig. 3(a), the open circle reference values were calculated from simulations in which we set $\tau_e = 60$ million ($60\mathrm{M} = 6\times10^7$) MC steps. Since all

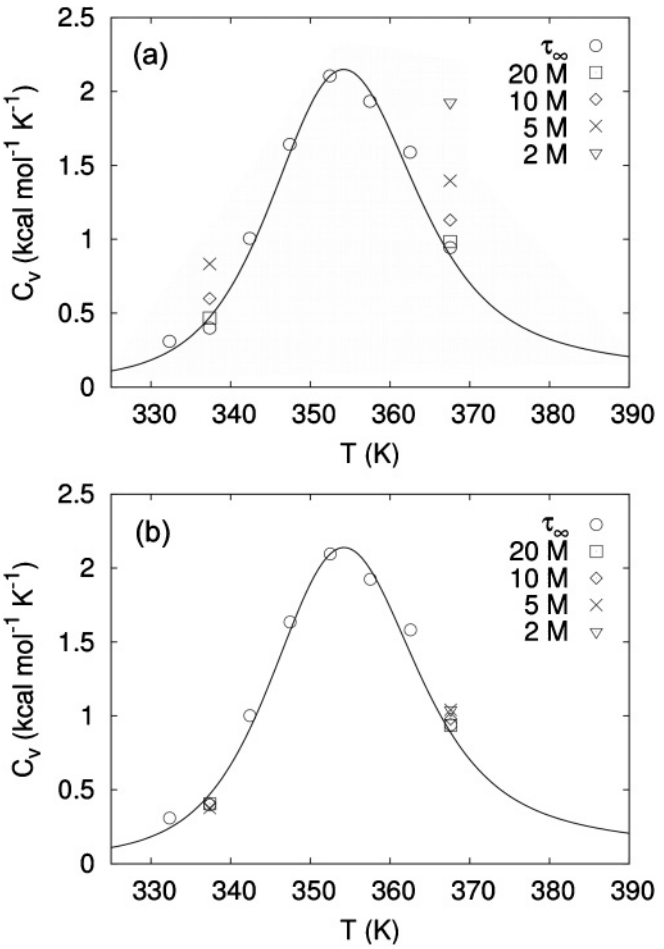

FIG. 3. Calculated heat capacity as a function of temperature for the two-helix bundle. The open circle (○) τ_e=∞ data points are the correct values to be used as reference points. The solid curve is the heat capacity calculated from single-temperature ($T \sim 352$ K) equilibrium simulations using the histogram technique [54] and helps guide the eye. (a) Uncompensated calculations are shown at T=337 and 367 K for different τ_e, given in units of M=10^6 MC steps. For smaller τ_e, the success rate for simulations decreases (Table I) and the uncompensated C_v deviate further from the reference values, exemplified by the 2M value at 367 K. (b) Heat capacities from the same incomplete sampling simulations used in (a) but calculated using the compensation method of Eq. (11). The compensation method produces results that are very close to the correct value (τ_e=∞) for all values of τ_e, including τ_e that are so short that configuration space is poorly sampled and the success rate for transitions is low.

simulations at all temperatures were successful within this time, we can treat 60M MC steps as $\tau_e = \infty$. Near the transition temperature, $T_c \sim 352$ K, the folding and unfolding processes are easy and all simulations are successful in less than 7×10^6 (7M) MC steps. More specifically, the MFPT is the time by which half of the simulations are successful. For example, in this work we are using 100 simulations that attempt to fold at each temperature, as well as 100 simulations that attempt to unfold. The MFPT for folding, τ_F(MFPT), is the time required for the 50th slowest simulation to first fold. At $T_c = 352$ K, the MFPT for folding is $\tau_F(\text{MFPT}) = 2.06 \times 10^6$ MC steps and the unfolding $\tau_U(\text{MFPT}) = 1.97 \times 10^6$ MC steps.

TABLE I. Probability of success for folding and unfolding as a function of cutoff time τ_e for two different simulation temperatures. The transition temperature for the two-helix bundle is $T_c = 352$ K. The cutoff time is given in units of million (M = 10^6) MC steps.

$T = 337$ K		
τ_e (10^6 MC steps)	% folded	% unfolded
∞ (>60M)	100	100
20	100	70
10	100	53
5	100	37
$T = 367$ K		
τ_e (10^6 MC steps)	% folded	% unfolded
∞ (>60M)	100	100
20	90	100
10	73	100
5	48	100
2	23	100

However, at temperatures well above T_c, the folding transition is difficult and folding times can lengthen, though for this simple exemplar system they remain below 60M MC steps. Likewise, at temperatures well below T_c, unfolding is difficult and unfolding times become large. It is at these temperatures, far from T_c, that for more complicated systems it can become computationally unfeasible to obtain equilibrium simulations for both folding and unfolding. These are the conditions at which our compensation technique is most valuable.

For the simple system of the two-helix bundle, we examine two such temperatures, $T = 337\,\text{K} < T_c$, in which unfolding requires much longer times than folding [$\tau_F(\text{MFPT}) = 0.49\times10^6$ MC steps, $\tau_U(\text{MFPT}) = 6.70\times10^6$ MC steps], and $T = 367\,\text{K} > T_c$, in which folding requires much longer time than unfolding [$\tau_F(\text{MFPT}) = 5.41\times10^6$ MC steps, $\tau_U(\text{MFPT}) = 0.53\times10^6$ MC steps]. As can be seen in Table I, for T = 337 K, as we lower τ_e, a larger fraction of the unfolding runs are unsuccessful, and likewise for the decreasing probability of success for the folding runs at T=367 K. At these temperatures with small τ_e, if all 100 folding simulations are combined with all 100 unfolding simulations, there is undersampling of one of the regions in configuration space and the combined time series is not an equilibrium representation at that temperature. As Fig. 3(a) shows with these smaller values of τ_e, using all simulations of folding and unfolding regardless of whether they were successful, the heat capacity that is calculated using this uncompensated approach is not accurate when compared to the reference value ($\tau_e = \infty$) at the corresponding temperature. This problem is exemplified by the 2M value at 367 K.

In Fig. 3(b), we implement Eq. (11) to compensate for unsuccessful runs. At $T = 337$ K, for $\tau_e = 5$M, all 100 folding runs were successful but only 37 of the 100 unfolding runs were successful. Following the prescription of Eq. (11), we combined 37 folding simulations with all 100 of the unfolding runs. The value of this approach is dramatic. Using all 100 folding and unfolding runs for $\tau_e = 5$M simulations at T =

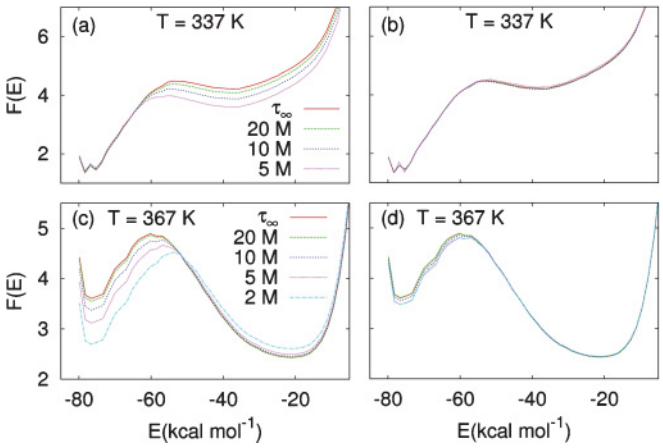

FIG. 4. (Color online) Free-energy curves $F(E)$ for the two-helix bundle as a function of energy at the same temperatures that are focused on for Fig. 3, 337 K ($<T_c$) and 367 K ($>T_c$). Different $F(E)$ curves are calculated using simulations that imposed different cutoff times τ_e. (a) and (c): Uncompensated results showing large deviations from the correct $\tau_e = \infty$ reference curve. (b) and (d): Compensated calculations produce results that collapse for all τ_e onto the reference curve $\tau_e = \infty$.

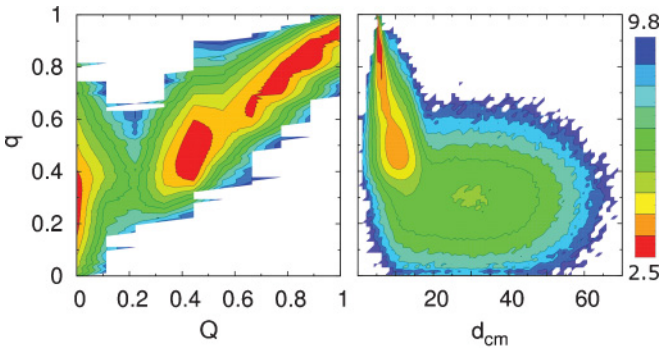

FIG. 5. (Color online) Free-energy landscapes for the leucine zipper dimer of Fig. 1. Because of the complexity of the system, only nonequilibrium, incomplete simulation trajectories are available. The calculation of F employs the compensation technique of Eq. (11). Red represent low F and blue represent high F (scale shown in kcal mol^{-1}). In both (a) and (b), $F(q,x)$ is calculated for different values of the secondary structural parameter, q which represents the fraction of amino acids that are in the native helical state. The other structural parameter x is (a) the fraction of native tertiary contacts Q that are intact. The region with $q \sim 1$, $Q \sim 1$ corresponds to the native-state basin, whereas $q \sim 0$ and $Q \sim 0$ correspond to the unfolded, random coil basin. (b) In addition to q, the other structural parameter is the center-of-mass separation $d_{\rm cm}$. The region with $q \sim 1$, $d_{\rm cm} \sim 6$ corresponds to the native-state basin.

337 K, Fig. 3(a) shows that the calculated heat capacity is far from the reference results. In contrast, Fig. 3(b) shows that implementation of Eq. (11) brings the heat capacity calculation very close to the reference value. An even more extreme example of the value of the method is demonstrated by the $\tau_e = 2$M point at $T = 367$ K in which only 23% of the folding runs were successful, as could occur for a system with a very complex energy landscape. Figure 3(a) shows that using all folding runs and all unfolding runs for calculating C_v gives a result that is very far from the reference value. This incorrect value is so high that it would broaden the peak of the C_v curve dramatically and lead to incorrect estimates of T_c. Figure 3(b) shows that when Eq. (11) is implemented, the calculated C_v value for the τ_e=2M point at T=367 K dramatically collapses to the correct reference value. In addition, Fig. 3(b) also shows that the calculated heat capacities that are the same for all τ_e, and the compensation method produces accurate results even when the undersampling is large.

We examined another important thermodynamic parameter, free energy F of the two-helix bundle. In Fig. 4, we plot F as a function of E for the same two temperatures that we focused on for Fig. 3, 337 and 367 K. In Figs. 4(a) and 4(c), we plot $F(E)$ using all 100 folding simulations and all 100 unfolding simulations. As with the heat capacity graph of Fig. 3(a), for smaller τ_e the calculation of $F(E)$ deviates far from the correct reference value given by the $\tau_e = \infty$ curve. The results are so inaccurate that it can be hard to even distinguish which state is stable at 367 K. In Figs. 4(b) and 4(d), we plot $F(E)$ that is recalculated using the compensation method. As can be seen, as with Fig. 3(b), the compensation method correctly collapses the calculated $F(E)$ for all τ_e onto the reference curve $\tau_e = \infty$.

The compensation method used in Figs. 3 and 4 for the relatively simple two-helix bundle system can be applied also to more complex multistate systems involving metastable transition states. We performed folding and unfolding simulations of the GCN4-p1 leucine zipper dimer [51–53] that is displayed in Fig. 1. For the energy parameters and amino acid sequence used in these simulations, there was no temperature at which there is 100% success for both folding and unfolding runs within $\tau_e = 100$M. In Fig. 5 we plot the free-energy landscape as a function of various structural parameters at $T = 347$ K. At this temperature, which is near the transition temperature, 100% of the folding runs were successful, but only 87% of the unfolding runs were successful, and therefore we applied our compensation method to get the proper configuration-space sampling ratio. Figures 5(a) and 5(b) are projections of the free-energy landscape onto different planes of the multidimensional configuration space. In both Figs. 5(a) and 5(b), $F(q,x)$ is calculated for different values of the secondary structural parameter q. We plot F in terms of the fraction of amino acids that are in the native helical state as compared to the maximum of $q = 74$. The other structural parameter x in Fig. 5(a) is the fraction of native tertiary contacts Q, with $Q = 1$ corresponding to all nine contacts. The region with $q \sim 1$, $Q \sim 1$ corresponds to the native-state basin, whereas the region with $q \sim 0$ and $Q \sim 0$ corresponds to the unfolded, random coil basin. In Fig. 5(b), in addition to q, the other structural parameter is the center-of-mass separation $d_{\rm cm}$. In both Figs. 5(a) and 5(b), transition states that are populated are shown. This shows the value of our compensation method when calculating thermodynamic properties such as free energy, because Figs. 5(a) and 5(b) reveal likely routes from the unfolded region to the native-state configuration.

V. CONCLUSION

We describe a method that allows computational simulations to be used to calculate accurate values of thermodynamic parameters for systems that are complex enough that computer

simulations cannot properly sample the configurational landscape. Our compensation method can be applied to any system that has transitions between two stable regions of configuration space, and is valid when there are intermediate transition states that are appropriately sampled. We present results for two different biological physics systems, a relatively simple two-helix bundle protein, and a more complicated leucine zipper dimer.

In order to accurately calculate thermodynamic parameters such as heat capacity and free energy, simulations must explore configuration-space regions with a fractional time given by the Boltzmann probability. For systems with many different types of interactions and many degrees of freedom, this may be computationally unfeasible at many temperatures because of the complexity of the energy landscape. As shown in Figs. 3 and 4, thermodynamic parameters calculated from these incomplete, nonequilibrium simulations will give highly inaccurate results if compensational methods are not used. The results can be so far off from the correct values that little insight can be obtained about the dynamics of the system. Our method gives a straightforward technique for compensating for the nonequilibrium sampling and allows nonequilibrium simulations to be combined to allow accurate calculations of thermodynamic parameters.

[1] H. Nymeyer, A. E. Garcia, and J. N. Onuchic, Proc. Natl. Acad. Sci. USA **95**, 5921 (1998).
[2] J. N. Onuchic, Z. Luthey-Schulten, and P. G. Wolynes, Annu. Rev. Phys. Chem. **48**, 545 (1997).
[3] H. Frauenfelder, S. G. Sligar, and P. G. Wolynes, Science (New York) **254**, 1598 (1991).
[4] H. Frauenfelder, Proc. Natl. Acad. Sci. USA **99**, Suppl. 1, 2479 (2002).
[5] P. P. Chapagain *et al*., J. Chem. Phys. **127**, 075103 (2007).
[6] D. E. Shaw *et al*., Science (New York) **330**, 341 (2010).
[7] B. J. Berne and J. E. Straub, Curr. Opin. Struct. Biol. **7**, 181 (1997), and references therein.
[8] R. H. Swendsen, J. S. Wang, and A. M. Ferrenberg, in *The Monte Carlo Method in Condensed Matter Physics*, edited by K. Binder (Springer, Berlin, 1992), p. 75.
[9] G. G. Maisuradze, A. Liwo, and H. A. Scheraga, Phys. Rev. Lett. **102**, 238102 (2009).
[10] H. Lei and Y. Duan, Curr. Opin. Struct. Biol. **17**, 187 (2007).
[11] A. Roitberg and R. Elber, J. Chem. Phys. **95**, 9277 (1991).
[12] B. A. Berg and T. Neuhaus, Phys. Rev. Lett. **68**, 9 (1992).
[13] F. Yasar *et al*., J. Comput. Chem. **23**, 1127 (2002).
[14] J. P. Valleau and D. N. Card, J. Chem. Phys. **57**, 5457 (1972).
[15] G. M. Torrie and J. P. Valleau, Chem. Phys. Lett. **28**, 578 (1974).
[16] P. G. Bolhuis *et al*., Annu. Rev. Phys. Chem. **53**, 291 (2002).
[17] G. Hummer, J. Chem. Phys. **120**, 516 (2004).
[18] J. Lee, Phys. Rev. Lett. **71**, 211 (1993).
[19] R. H. Swendsen and J. S. Wang, Phys. Rev. Lett. **57**, 2607 (1986).
[20] Y. Sugita and Y. Okamoto, Chem. Phys. Lett. **314**, 141 (1999).
[21] Y. M. Rhee and V. S. Pande, Biophys. J. **84**, 775 (2003).
[22] H. A. Scheraga, M. Khalili, and A. Liwo, Annu. Rev. Phys. Chem. **58**, 57 (2007).
[23] M. G. Wolf *et al*., J. Phys. Chem. B **112**, 13493 (2008).
[24] A. J. Ballard and C. Jarzynski, Proc. Natl. Acad. Sci. USA **106**, 12224 (2009).
[25] P. Liu *et al*., Proc. Natl. Acad. Sci. USA **102**, 13749 (2005).
[26] N. Singhal, C. D. Snow, and V. S. Pande, J. Chem. Phys. **121**, 415 (2004).
[27] N. W. Kelley *et al*., J. Chem. Phys. **129**, 214707 (2008).
[28] P. M. Kasson and V. S. Pande, Biophys. J. **95**, L48 (2008).
[29] G. Jayachandran, V. Vishal, and V. S. Pande, J. Chem. Phys. **124**, 164902 (2006).
[30] C. Jarzynski, Phys. Rev. Lett. **78**, 2690 (1997).
[31] C. Jarzynski, Phys. Rev. E **56**, 5018 (1997).
[32] M. T. Woodside, C. Garcia-Garcia, and S. M. Block, Curr. Opin. Chem. Biol. **12**, 640 (2008).
[33] N. C. Harris and C. H. Kiang, Phys. Rev. E **79**, 041912 (2009).
[34] D. K. West, P. D. Olmsted, and E. Paci, J. Chem. Phys. **125**, 204910 (2006).
[35] F. Liu, H. Tong, and Z. C. Ou-Yang, Biophys J. **90**, 1895 (2006).
[36] R. Zwanzig, Proc. Natl. Acad. Sci. USA **94**, 148 (1997).
[37] C. Hyeon, G. Morrison, and D. Thirumalai, Proc. Natl. Acad. Sci. USA **105**, 9604 (2008).
[38] O. K. Dudko *et al*., Biophys. J. **92**, 4188 (2007).
[39] C. Hyeon *et al*., Proc. Natl. Acad. Sci. USA **106**, 20288 (2009).
[40] P. P. Chapagain and B. S. Gerstman, Biopolymers **81**, 167 (2006).
[41] P. P. Chapagain and B. S. Gerstman, J. Chem. Phys. **120**, 2475 (2004).
[42] Y. Liu *et al*., J. Chem. Phys. **128**, 045106 (2008).
[43] Y. Liu, P. P. Chapagain, and B. S. Gerstman, J. Phys. Chem. B **114**, 796 (2010).
[44] J. Skolnick and A. Kolinski, Science (New York) **250**, 1121 (1990).
[45] P. Chapagain and B. Gerstman, J. Chem. Phys. **119**, 1174 (2003).
[46] J. Skolnick and A. Kolinski, J. Mol. Biol. **221**, 499 (1991).
[47] A. Kolinski and J. Skolnick, Proteins **18**, 338 (1994).
[48] A. Kolinski and J. Skolnick, Proteins **18**, 353 (1994).
[49] A. M. Ferrenberg and R. H. Swendsen, Phys. Rev. Lett. **61**, 2635 (1988).
[50] A. M. Ferrenberg and R. H. Swendsen, Phys. Rev. Lett. **63**, 1195 (1989).
[51] R. A. Kammerer *et al*., Proc. Natl. Acad. Sci. USA **95**, 13419 (1998).
[52] R. A. Kammerer *et al*., J. Biol. Chem. **276**, 13685 (2001).
[53] P. P. Chapagain, Y. Liu, and B. S. Gerstman, J. Chem. Phys. **129**, 175103 (2008).
[54] N. D. Socci and J. N. Onuchic, J. Chem. Phys. **103**, 4732 (1995).