To appear in: J. Chem. Phys. 149, 244119 (2018)

Quantitative comparison of adaptive sampling methods for protein dynamics

E. Hruska,[1,2] J.R. Abella,[3] F. Nüske,[1,4] L.E. Kavraki,[3] and C. Clementi[1,4,*]

[1] *Center for Theoretical Biological Physics, Rice University, Houston, TX, United States*
[2] *Department of Physics, Rice University, Houston, TX, United States*
[3] *Department of Computer Science, Rice University, Houston, TX, United States*
[4] *Department of Chemistry, Rice University, Houston, TX, United States*

(Dated: November 30, 2018)

Adaptive sampling methods, often used in combination with Markov state models (MSMs), are becoming increasingly popular for speeding up rare events in simulation such as molecular dynamics (MD) without biasing the system dynamics. Several adaptive sampling strategies have been proposed, but it is not clear which methods perform better for different physical systems. In this work, we present a systematic evaluation of selected adaptive sampling strategies on a wide selection of fast folding proteins. The adaptive sampling strategies were emulated using models constructed on already existing simulated MD trajectories. We provide theoretical limits for the sampling speedup and compare the performance of different strategies with and without using some a priori knowledge of the system. The results show that for different goals, different adaptive sampling strategies are optimal. In order to sample slow dynamical processes such as protein folding without a priori knowledge of the system, a strategy based on the identification of a set of metastable regions is consistently the most efficient, while a strategy based on the identification of microstates performs better if the goal is to explore newer regions of the conformational space. Interestingly, the maximum speedup achievable for the adaptive sampling of slow processes increases for proteins with longer folding times, encouraging the application of these methods for the characterization of slower processes, beyond the fast-folding proteins considered here.

I. INTRODUCTION

Molecular dynamics (MD) simulations have become indispensable for gaining insight into molecular systems at high spatial and temporal resolutions. However, a key limitation for MD with accurate all-atom force-fields remains the computational demand for simulating processes with long timescales. In particular, biologically relevant processes, such as protein folding and conformational changes, typically require simulation time in excess of milliseconds, while atomic-resolution MD trajectories can currently reach timescales on the order of microseconds on standard computational resources.

In the last decade, significant efforts have been devoted to alleviate the MD timescale problem. In general, such efforts can be divided into three broad categories. In the first category, the use of different computational resources has allowed the simulation of longer timescales, either by cumulating trajectories from massively-distributed computing [1, 2] or by the design of special-purpose hardware [3]. The second class of methods can be characterized by their ability to accelerate the occurrence of rare events in simulation, thereby reducing the actual computational time needed to observe biophysically relevant processes in practice. Examples in this class include accelerated MD [4], replica-exchange MD [5] and metadynamics [6]. While these methods improve the efficiency of sampling and can be used for thermodynamics studies, they alter the system's Hamiltonian, and can not be directly used to extract kinetic information from the simulation. Path reweighting methods [7–10] have been recently proposed to recover the kinetic information after altering the system's Hamiltonian.

The third class of methods can be grouped under the term *adaptive sampling* [11–21]. An analysis of the power and limitation of such methods is the subject of this work.

Instead of simulating long MD trajectories to observe rare events, adaptive sampling methods take a "divide and conquer" approach and attempt to iteratively combine many short MD simulations, distributing them in a way to escape from local free energy minima, and efficiently visit different regions of the conformational space of the systems of interest. At each iteration, all of the simulations that have been performed at that point are pooled and analyzed. New simulations are then initialized by using the information extracted from the analysis of the previous iterations. The main idea of adaptive sampling is that, by periodically analyzing the conformational space already explored, new simulations can be restarted in a way that may significantly enhance the probability of observing rare events. The choice of the strategy chosen to restart the trajectories is crucial to the success of the approach, and several different methods have been proposed [13–17, 22–24]. Recently, enhanced sampling in combination with adaptive sampling methods has also been proposed [25].

The popularity of adaptive sampling methods is due to the significant advances in the analysis of MD trajectories. In the last decade, different methods have been put forward to extract essential information from high dimensional MD data to a small number of reaction coordinates associated with the slow collective processes in the sys-

* cecilia@rice.edu

tem's dynamics [26, 27]. Such methods include Markov State Models (MSMs) [28–32], Diffusion Maps [33–36], likelihood based approaches [37], cut-based free energy profiles [38], or neural networks [39–41]. In particular, MSMs provide a good complement for adaptive sampling as they are designed to handle many short trajectories and do not require an equilibrium sampling to recover global thermodynamics and kinetic properties (such as metastable states, free energy barriers, and transitions between states), as long as the trajectories are in local equilibrium.

As mentioned above, different adaptive sampling methods can be characterized by how the information extracted from previously explored space is used to initiate new trajectories at each iteration. Although the power of adaptive sampling has been demonstrated by successful applications [42–44], there is no general consensus as to how to choose a particular method over another for a specific system. If the goal is to simulate a rare event such as protein folding, does a method based on eigenvalues outperform one based on counts? Could the same adaptive sampling method be then used for general exploration of conformational space? Additionally, previous studies [12–15] report efficiency gains with adaptive sampling between a factor 2 and a factor of 10. What characteristics of the system of interest can we use to predict that a particular adaptive sampling method will provide a better efficiency gain? Here we present a systematic study on a number of model systems to address these questions. In particular, we consider the efficiency of different adaptive sampling strategies for two different goals on a number of model systems: to speedup the simulation time needed to observe a specific rare event, such as the folding of a protein, or to speedup the exploration of large regions of the conformational space of the same protein.

In order to be able to benchmark and compare the results on different systems, we use previously generated extensive MD trajectories [45] to generate 8 discrete models for protein dynamics. The results of this analysis reveal that different strategies are needed for different goals. In particular, on-the-fly estimation of global equilibrium properties from non-equilibrium data is very important to speedup the folding of a protein, while the knowledge of equilibrium properties is not needed if the goal is the exploration of large regions of the conformational space (independently if folded or unfolded). This result suggests that different strategies may need to be combined in various stages of a specific application to both enhance the occurrence of a rare event and appropriately sample the different metastable states. Comparison of the results on different proteins suggest that the speedup that can be achieved by adaptive sampling is larger for slower processes, thus encouraging the application to more complex systems.

II. METHODS

A. Dataset of Simulations

We used previously existing long all-atom MD trajectories of 8 different small proteins[45], obtained on the Anton supercomputer, to generate discrete model systems, as discussed below. The dataset is summarized in Table I and contains proteins ranging from 10 to 80 residues, with different topologies (α-helices, β sheets, or a mix of both), simulated folding times ranging from 0.6 to 49 μs, and different timescale gaps between the folding process and other competing slow processes.

TABLE I: Previuosly simulated proteins used to generate discrete models in this study

Protein Name	PDB ID of Folded Structure	Size (# residues)	Folding Time (μs) from [45]
Chignolin	2RVD	10	0.6
Trp-cage	2JOF	20	14
BBA	1FME	28	18
WW Domain	2F21	35	21
Protein B	1PRB	47	3.9
Homeodomain	2P6J	52	3.1
α3D	2A3D	73	27
λ-repressor	1LMB	80	49

B. Construction of discrete protein models

To emulated adaptive sampling, an MSM was generated for each protein from the previously existing long all-atom MD trajectories, then synthetic microstate trajectories are generated by sampling the MSM transition matrix. An MSM models the system's dynamics by discretizing the explored conformational space into a finite number of states, and estimating their probability and the probability of transition between them [28]. The analysis is summarized by a transition matrix, T_{ij}, that indicates the probability that the system transitions from state i to state j within a chosen lag time τ. The discretization of the original conformational space into states (also called microstates) is usually performed by clustering the configurations sampled by MD trajectories using a distance metric that can separate slowly mixing configurations from rapidly interconverting ones [46, 47].

We have used standard procedures to perform these steps. In particular, for each protein, we used the Time-lagged Independent Component Analysis (TICA) [48, 49] combined with the commute map [46], to reduce the dimensionality of the system. As an input for TICA, each conformation was first featurized using all pairwise inter-residue distances (between the two closest heavy-atoms) and all dihedral angles along the protein chain. For smaller systems, the reciprocals of the inter-residue distances were also used as additional features. The Euclidean distance between the lower-dimensional points in the commute map space provides a good measure to obtain a kinetically meaningful state decomposition, and an associated MSM [46]. All conformations were then

partitioned with k-means clustering into 1000 or 2000 microstates, depending on the size of the protein. It was ensured that the slowest MSM eigenvector is the folding-unfolding process and all microstates are connected by removing disconnected microstates. Finally, the transition matrix for the MSM is computed using maximum-likelihood estimation with a detailed balance constraint. The lag time, τ, for the MSM was chosen based on the convergence of the implied timescales, and the Markovianity property of the MSM was tested by using the Chapman-Kolmogorov test [28]. All the analysis was performed using the PyEMMA Python package [50] and the exact parameters for the construction of discrete protein models for each protein are listed in the Supplementary material.

C. Simulating Trajectories using MSMs

Adaptive sampling involves iteratively running many short MD trajectories, and different adaptive sampling methods differ in how the new structures are chosen to initialize the next round of MD trajectories. We can simulate the adaptive sampling process of iteratively running an ensemble of n MD trajectories using the transition matrix from an MSM as follows. Note that the restart strategies here are concerned with selecting states visited among the discrete set of microstates in the MSM. In actual molecular dynamics simulations, continuous trajectories in a protein configurational space are used instead of the synthetic trajectories generated here by jumping between the different discrete states of an MSM in adaptive step 2. Therefore, in actual simulations the analysis (adaptive step 3 below) involves also the the discretization of all the available trajectories into a set of microstates.

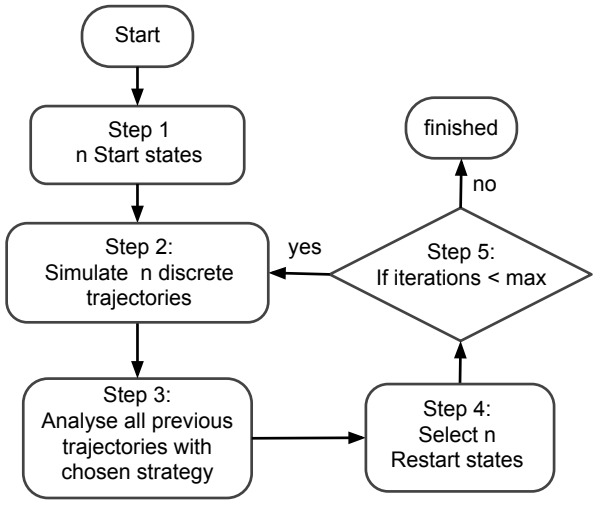

FIG. 1: Adaptive sampling strategy schema

- Adaptive step 1: Start with a randomly chosen unfolded state from the discrete set of microstates available for a given protein

- Adaptive step 2: Generate n independent discrete trajectories (of a fixed length) from the selected state(s) using the probabilities from the MSM transition matrix

- Adaptive step 3: Analyze the ensemble of trajectories generated

- Adaptive step 4: Select n microstates among the ones visited so far from which to start the next round of trajectories

- Adaptive step 5: Repeat adaptive steps 2,3 and 4 or finish after a certain number of iterations

We denote the adaptive steps 3 and 4 together as the *restart strategy* for an adaptive sampling method. Figure 1 is a graphical representation of the process. The different restart strategies that we use in this work are described in detail in the following section. When using continuous trajectories from actual molecular dynamics simulations, the analysis step 3 also includes the discretization of the continuous trajectories into discrete trajectories (for instance, by means of TICA and MSM). Thus, the restart strategy for adaptive sampling in actual molecular dynamics simulations must also select a set of individual conformations or frames from the selected microstates to initialize the next MD simulation. This can be done for example in a uniformly random fashion within the microstate or by selecting a representative conformation. The length of the trajectories in each iteration can be varied, but it needs to be larger than the lag time used to generate the MSM. Here we chose the length of each short MD trajectory the same as the lag time τ, that is, the analysis is performed after the discrete trajectories have been propagated by one step of length τ. At each iteration, for a given strategy, the n restarting points for the new trajectories are chosen independently of each other.

In order to study the speedup in folding, subsets of the discrete microstates for each protein are denoted as folded and unfolded states. Using the PDB files from Table I, the native contacts are extracted for the folded structure of each protein. A native contact is defined if the distance between the two closest heavy atoms in a pair of residues is 4Å or less. For each state in the MSM, we compute the median number of native contacts over all the conformations mapped to the state. States above a threshold value for the number of native contacts are assigned as a folded state. States below a threshold value for the number of native contacts are assigned as an unfolded state. The threshold values for individual proteins can be found in the Supplementary material.

D. Restart Strategies for Adaptive Sampling

For each protein model, we use the MSM analysis and adaptive sampling procedure detailed above with different restart strategies. We use a number of popular strategies that do not assume a priori knowledge of the system, such as the microstate counts, or strategies that assume some a priori knowledge of the system, such as the number of native contacts.

Here we describe all the restart strategies that we have used on all the different protein models. Several of these strategies require to set the value of some parameters, which are provided in the Supplementary material.

a. MD As a reference, we generate synthetic MD trajectories without any adaptive choice of the restart points. No analysis is performed after each iteration, and each trajectory is restarted from the same state where it ended in the previous iteration. That is, the restart state chosen for trajectory n_i at iteration t is simply the state of trajectory n_i at iteration $t-1$.

b. Microstate Counts ($1/C$) One intuitive and popular restart strategy consists in choosing the restart states based on how many times the previous trajectories have visited each state in the conformational space [13, 14, 17], in order to favor less populated states. In particular, a given state i is chosen as restart state with a probability inversely proportional to the number of times it has been visited.

c. Macrostate Counts ($1/C_M$) Another count-based method that has been used in different applications clusters all the visited microstates into fewer metastable macrostates on-the-fly. Usually, eigenvectors of a matrix summarizing the sampling performed are used for the clustering [15, 16]. Here we use the transitions between all the visited microstates to build an on-the-fly MSM, and the microstates are clustered into macrostates using PCCA+ [22]. The restart state is then chosen with the following procedure. A macrostate is first chosen with probability inversely proportional to the number of times the macrostate has been visited. Then a microstate within the chosen macrostate is chosen with probability inversely proportional to the number of times the microstate has been visited. We have tested four variations of this strategy. The first two variations (named $1/C^C_{M,1}$ and $1/C^C_{M,2}$) make use of the count matrix C_{ij} to directly estimate the on-the-fly MSM transition matrix. The count matrix C_{ij} contains the number of transitions that have been recorded in previous iterations from state i to state j. Every time a state is visited, the corresponding value in the count matrix is incremented by one. This count matrix is normalized such that each row sums to one and then used to estimate the on-the-fly MSM for the adaptive sampling strategies. The two variations differ as follows:

$1/C^C_{M,1}$: PCCA+ is used to cluster the microstates into 30 macrostates.

$1/C^C_{M,2}$: The number of macrostates generated by PCCA+ is based on the number of significant timescales using a 50% kinetic content cutoff [46].

The next two variations (named $1/C^K_{M,1}$ and $1/C^K_{M,2}$) are used to estimate the effect of using non-equilibrium trajectories for the adaptive sampling strategies. Since in most adaptive sampling methods many relatively short trajectories are used, the non-equilibrium sampling can introduce errors in the analysis of these trajectories. Recently, the Koopman reweighting method [51–56] has been introduced to correct for the non-equilibrium effects in estimating global equilibrium properties and can significantly reduce this error. In order to evaluate the effect of the non-equilibrium sampling error in the performance of the adaptive sampling strategy, we assume that the use of Koopman reweighting in the analysis of MD trajectories can provide an accurate estimate of the equilibrium transition probabilities between any pair of explored microstates. Thus, in the synthetic trajectories used here, at each iteration, we estimate an on-the-fly Koopman-corrected MSM by using the true transition probability between the explored microstates (properly renormalized) and discarding any transition to unexplored states. Two more variants are studied by applying this correction to the previous two:

$1/C^K_{M,1}$: PCCA+ is used to cluster the microstates into 30 macrostates, on the Koopman-corrected MSM

$1/C^K_{M,2}$: The number of macrostates generated by PCCA+ on the Koopman-corrected MSM is based on the number of significant timescales using a 50% kinetic content cutoff [46].

d. Q_f - Native Contacts If additional information is available on the system of interest, it can also be used to guide the sampling. For instance, it has been proposed [18] to select restarting structures for adaptive sampling based on the number of contacts likely made in the folded states based on an evolutionary coupling analysis. Alternatively, the FAST method [19] was proposed as a way to exploit a priori information, such as the distance to a target structure. Here, we consider the case where the folded structure is known, and the number of native contacts can be used as a reaction coordinate for the folding process. Out of the states already visited by the simulation, states with a higher median number of native contacts are chosen with higher probability than states with a lower number of native contacts. The probability of choosing a visited state i is proportional to $exp(-k * |Q_i - Q_{max}|)$, where Q_i is the number of native contacts in state i, Q_{max} is the total number of native contacts, and k is a parameter of the strategy (see Supplementary material).

e. $Q_{f,nn}$ - Native and Non-native contacts A variation of the previous strategy is to use two reaction coordinates in the case when the folded structure is known, keeping track of the number of both native and non-native contacts that are formed during the simulation. For each state in the MSM, we compute the median number of native and non-native contacts over all conformations mapped to each state. Out of the states already

visited by the simulation, states with a higher number of native contacts have a higher probability of being chosen as restarting points, as in the Q_f strategy described above. Additionally, states with a lower number of non-native contacts are chosen with a higher probability than states with a higher number of non-native contacts. The probability of choosing a visited state i is proportional to $exp(-d_i)$, where $d_i = \sqrt{k_1^2 * (Q_i - Q_{max})^2 + k_2^2 * N_i^2}$. Q_i is the number of native contacts in state i, Q_{max} is the total number of native contacts, N_i is the number of non-native contacts in state i, and k_1, k_2 are parameters of the strategy. One can think of d_i as a distance to the folded state in native/non-native contact space (scaled by k_1 and k_2). The two parameters k_1 and k_2 were optimized by a parameter sweep (see Supplementary material). In real simulations such an optimization of the parameters is not possible, but we perform it here to estimate the upper bound for the speed up.

f. p_{esc} - Optimal strategy for exploration We also test a strategy that is not feasible in practice but offers a baseline comparison as a theoretically optimal one. This strategy is built by using knowledge of the full transition matrix of the system, that is not a priori known in real applications (it is usually the goal of the sampling). For each visited microstate i, we compute the probability to transition to any microstate not yet explored using the true transition matrix:

$$p_{esc}[i] = \sum_{j \in unexplored} T[i,j]$$

The state with the highest p_{esc} value is chosen as the restart state. As stated above, this strategy is impossible to implement in practice for real protein simulations, but it is as a useful benchmark for comparing adaptive sampling strategies that aim to explore conformational space.

g. t_{opt} - Optimal strategy to speedup slow processes (protein folding) We also test another theoretically optimal strategy given perfect knowledge of the system dynamics as well as knowledge of the folded states. In a way that is similar to the definition of mean first passage time [57], for each state i in the MSM, we compute a value $t_{opt}[i]$ which estimates the minimal time to reach the folded state. We first define that for each folded state f, $t_{opt}[f] = 0$. Then we iteratively solve the following recurrence relation for each state i outside the folded region:

$$t_{opt}[i] = 1 + \sum_{j \in states} T[i,j] min(t_{opt}[i], t_{opt}[j])$$

The equation is solved iteratively until the relative change in t_{opt} drops below a cutoff. We then use it to define a benchmark restart strategy, by selecting the restart state among the ones explored that has the lowest t_{opt} value, representing the state that is the closest to the folded state. Note again that this strategy is impossible to implement in practice, but still is a useful benchmark for adaptive sampling strategies. With the t_{opt} benchmark the maximum achievable speedup with adaptive sampling for the folding of a protein can be evaluated.

III. RESULTS AND DISCUSSION

In order to quantify the performance of different adaptive sampling strategies, we considered two broad measures of efficiency. The first measure is the time it takes for a strategy to simulate a rare event, in terms of steps of synthetic trajectories. For the dataset used here, the rare event of interest is the folding process and, for all proteins considered, the slowest timescale (or rarest event) is the folding timescale. For each strategy, the average time measured for a given strategy to reach the folded state starting from an unfolded state is compared with the corresponding time in the absence of adaptive sampling (that is, for the MD strategy described above). The second measure is one that focuses on the exploration of the configurational space instead of a single rare event. For any given strategy, we measure the time needed to explore 95% of the states used to build the MSM and compare it with the corresponding MD time. For each protein, we evaluate these two measures for each of the adaptive sampling strategies described above. Each strategy is evaluated by using a different number of parallel trajectories n, ranging from 1 to 5000. The results reported are averaged over 100 independent runs per protein and per number of parallel trajectories.

A. Time to fold

Figure 2 shows the average folding time for each of the strategies for three different proteins using 100 parallel trajectories. First, we note that the popular microstate-based $1/C$ strategy does not always appear to speedup the folding time, while the macrostate-based methods do show significant improvement over MD. Interestingly, the benchmark strategy designed to maximize the probability to visit unexplored regions of the configurational space (p_{esc}, defined above) does not significantly speedup the sampling of the folding rare event. Both $1/C$ and p_{esc} are strategies designed for general exploration and not specifically for rare event sampling, and it is not surprising that these strategies do not perform well in accelerating folding events. Instead, the macrostate-based methods appear to successfully introduce a sampling bias toward states that are along the direction of the slowest timescale, as manifested in the significant speedup with respect to simple MD. These results are consistent over the set of model proteins studied.

Within the macrostate-based methods, the correction for non-equilibrium that can be achieved by Koopman reweighting ($1/C_M^K$) appears to further improve the sampling of the rare folding event over using a simple uncorrected count matrix ($1/C_M^C$) in the on-the-fly MSM

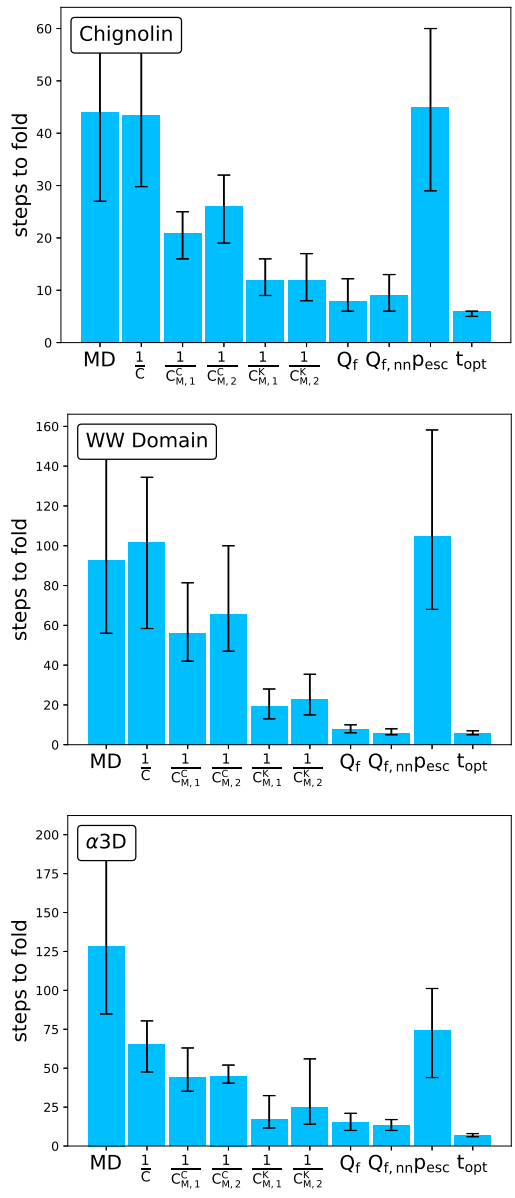

FIG. 2: Comparison of the number of steps of synthetic trajectories required to fold for the different adaptive sampling strategies for three different proteins using 100 parallel trajectories. The 20% and 80% percentiles are shown as the error bars. The results of the t-test between MD and the individual strategies for the different proteins are reported in the Supplementary material.

definition. The correction for non-equilibrium allows a more accurate estimation of the leading eigenvector of the true transition matrix over using the raw, on-the-fly count matrix. Thus, the resulting macrostates are more kinetically relevant. We also observe that the results obtained when the number of macrostates is defined by the kinetic content ($1/C_{M,2}$) do not differ significantly from what is obtained when a constant number of macrostates is used ($1/C_{M,1}$). In some instances, it appears that using kinetic content slightly hurts the performance of the adaptive sampling, which may be due to inaccurate estimation of the timescales in the early stages of the simulation.

Finally, we observe that incorporating a reaction coordinate, such as the number of native contacts in the strategies does indeed significantly improve the sampling of rare events. The improvement is, in fact, close to the theoretical maximum that is estimated by t_{opt}. We also observe that the addition of the number of non-native contacts as a second reaction coordinate does not always improve the performance of the algorithm. In particular, this is true for the smallest of the protein model considered (Chignolin), the kinetics of which does not exhibit any additional slow processes besides folding (see Fig. 2). For all the other protein models, the introduction of a second reaction coordinate very marginally improves the sampling of the folding process. These patterns are consistent across all the proteins and across the number of parallel trajectories used. The plots for all proteins (in addition to Fig. 2) can be found in the Supplementary material.

B. Time to explore 95% of states

Figure 3 shows the average time needed for the different adaptive sampling strategies to explore 95% of the microstates constituting the MSM, by using 100 parallel trajectories, for three different proteins. In this comparison we exclude the strategies designed to speedup the sampling of the folding process (such as the native contact based strategies as well as t_{opt}) because they are not designed for the purpose of general exploration. The comparison shows that, in general, the $1/C$ strategy explores the configurational space much more efficiently than plain MD. The speedup obtained by the $1/C$ strategy nears the theoretical maximum obtained by the optimal exploration strategy, p_{esc}. Within the macrostate-based strategies, there is more variance. The strategies using the regular count matrix ($1/C^C_M$) outperform the strategies that correct for non-equilibrium errors, ($1/C^K_M$). This is likely because the correction introduces a bias towards the sampling of slow processes rather than general exploration. The non-equilibrium error in the count matrix based strategies introduces randomness that helps the sampling of unexplored microstates. Additionally, for some proteins the optimization of the number of macrostates based on the kinetic content ($1/C_{M,2}$) does appear to provide an advantage over the use of a constant number of macrostates ($1/C_{M,1}$). The use of the kinetic content allows for a more accurate estimation of macrostate counts, which could help to focus the sampling bias towards regions that are less densely sampled. The patterns shown in Fig. 3 are consistent across all the proteins and across the number of parallel trajectories used. The corresponding plots for all the proteins can be

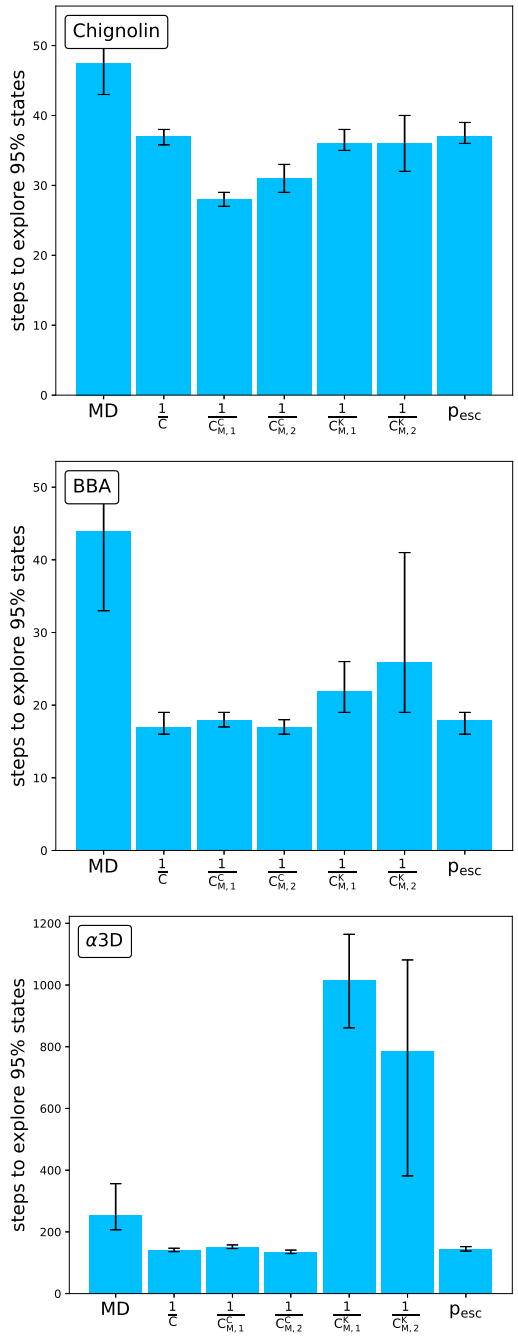

FIG. 3: Comparison of the number of steps required to explore 95% of the configurational space for different adaptive sampling strategies for three different protein models, by using 100 parallel trajectories. The 20% and 80% percentiles are shown as the error bars. The results of the t-test between MD and the individual strategies for the different proteins are reported in the Supplementary material.

found in the Supplementary material.

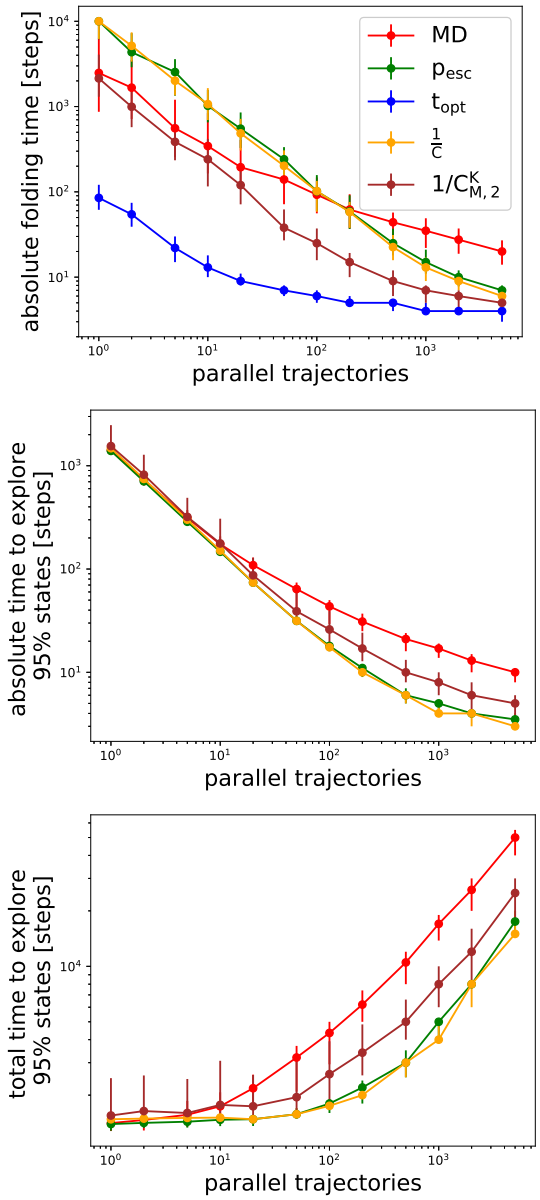

FIG. 4: Top: Scaling of the absolute folding time required to fold the protein model for the WW Domain for 5 different sampling strategies: plain MD, p_{esc}, t_{opt}, $1/C$ and $1/C^K_{M,2}$. Scaling of the absolute (middle) or cumulative (bottom) number of steps required to explore 95% of all microstates, for the protein model of BBA, for 3 different strategies: plain MD, p_{esc}, $1/C$ and $1/C^K_{M,2}$. The 20% and 80% percentiles are shown as error bars. Similar figures for the other protein models are reported in the Supplementary material.

C. Scaling

Adaptive sampling methods capitalize on the use of many relatively short parallel trajectories, usually deployed on massively parallel computers (MPC), to speed

up rare events or explore protein conformational spaces. In order to better understand the limits of scalability of adaptive sampling strategies, the measured absolute folding time for different parallelization is shown in Figure 4. The absolute folding time indicates the actual clock time required to record a folding event for a given protein on MPC with a given adaptive sampling strategy. The different strategies exhibit good scaling below a parallelization of around 100 and moderate scalability up to 1000 parallel trajectories. The scalability differs only slightly between different strategies, confirming that adaptive sampling generally scales well. Similar scaling is observed for all protein models. The time to explore 95% of microstates in Figure 4 scales to a higher parallelization than the time to fold the protein. In addition to Fig. 4, scaling plots for the other protein models are available in the Supplementary material.

D. Speedup for different proteins

The speedup in simulating the folding process achieved by using adaptive sampling in Figure 2 varies for different proteins as each protein has different dynamic properties. In order to better understand the factors determining the speedup reachable with adaptive sampling strategies, we compare different properties over the different protein models.

Figure 5 shows that, despite the small sample size and large error bars, there is a significant correlation between the theoretical maximum speedup in folding reachable with adaptive sampling (t_{opt}) and the folding time of a protein model (as measured by the mean first passage time). Similar correlations appear for the speedup achieved by using an adaptive sampling strategy based on the number of macrostates explored upon correction for non-equilibrium effect ($1/C_{M,2}$), and also when a reaction coordinate is used to guide the adaptive sampling (Q_f). That is, for slower folding proteins the efficiency of adaptive sampling strategies in accelerating the folding rare event increases. This result is very encouraging for the use of adaptive sampling strategies to sample slow processes, as adaptive sampling seems to perform better as the processes become slower. The large error bars are caused by the stochastic nature of the trajectories. No significant correlation is observed between the speedup achieved in folding and additional properties such as the size of the protein (Figures in Supplementary material).

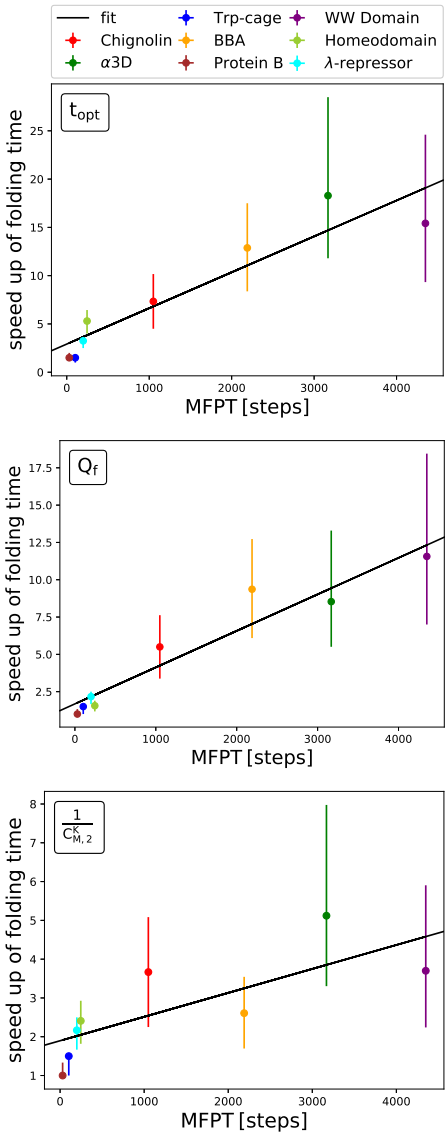

FIG. 5: The relationship between the speedup of the folding time with t_{opt}, Q_f or $1/C^K_{M,2}$ vs. mean first passage time for the 8 proteins. Results are reported for a parallelization of 100, and the 20% and 80% percentiles are shown as error bars. The speedup increases with longer MD folding time, the linear fit lines are drawn to guide the eye. The Pearson correlation coefficient is for t_{opt} 0.93, for Q_f 0.95 and for $1/C^K_{M,2}$ 0.82.

IV. CONCLUSION

We have presented a systematic analysis of the performance of different adaptive sampling strategies by using as test systems 8 different discrete protein models defined from long all-atom MD simulations. We have shown that different adaptive sampling strategies are optimal for different goals. In particular, if the goal of adaptive sampling is to speedup the simulation of a rare event (such as a protein folding process), it is important to be able to analyze the explored space on-the-fly and extract a few metastable states from which new simulations can be restarted. In the data analysis, it is also important to take into account the effect of non-equilibrium sampling. Indeed, our results show that a more accurate estimation of an equilibrium MSM from short non-equilibrium simulations, that can be obtained by using corrections based

on the estimation of the Koopman operator [51–56], significantly improve the adaptive sampling of a protein folding process with respect to a simple estimation of the MSM directly from non-equilibrium transition counts.

Different considerations are important if the goal of adaptive sampling is to speedup the exploration of any new regions of the configurational space of a protein system. In this case, it appears that the most efficient adaptive sampling strategy is based on the on-the-fly identification of a large number of kinetic microstates from the simulations already performed, and corrections for non-equilibrium effects do not appear relevant. These results suggest that different strategies could be used in different stages of investigation of a given biophysical process. For instance, the sampling of rare events could be optimized in a first stage to discover slow processes in a new system of interest, followed by a second stage where the different metastable regions in the conformational space can be better sampled by an adaptive sampling strategy optimized for fast exploration.

We have compared the speedup achieved with the different adaptive sampling strategies with theoretically optimal benchmark strategies for these two different goals, p_{esc} and t_{opt}, respectively. The gap between the speedup of the theoretically optimal strategies and the best performers among the strategies presented suggest that there could be even faster adaptive sampling methods and further investigation in this direction is underway.

We have also shown that, if there is a priori knowledge about the process under investigation, as for example a reaction coordinate, then adaptive sampling strategies for the sampling of rare events can be designed to achieve a speedup close to the theoretical maximum benchmark. In particular, we have shown that using the number of native (and non-native) contacts to guide the sampling, a significant improvement in the adaptive sampling of the folding process is obtained with respect to adaptive sampling strategies that do not use a priori knowledge of the system.

The adaptive sampling strategies reported here scale well with parallelization up to about 1000 for the investigated systems. This result generalizes what was reported in [12] for different proteins.

Although the best performing adaptive sampling strategies presented here show a robust speedup over plain MD over a number of different protein models, a significant variation in performance is observed. Interestingly, the speedup obtained with the best performing adaptive sampling strategies for the sampling of the folding process for different protein models correlates with the folding time as measured with plain MD simulations. Instead, the size of the proteins or the height of the folding free energy barrier for the different proteins do not appear to be a strong determinant for the speedup obtainable by adaptive sampling. A cautious extrapolation of the correlation between the adaptive sampling performance and the timescale of the folding rare event encourages the application of these methods for the characterization of slower processes, beyond the fast-folding proteins considered here. Due to the limited number of investigated proteins and the discrete nature of the models used, the upper limit of the speedup achievable with adaptive sampling methods for the sampling of rare events cannot be directly estimated from what is presented here.

V. SUPPLEMENTARY MATERIALS

See Supplementary material for complete results for all 8 proteins and the parameter for generating the MSM objects.

ACKNOWLEDGMENTS

We thank Frank Noé for stimulating discussions and support with the MSM analysis, and members of the Clementi group for valuable suggestions. We are indebted to D.E. Shaw Research for sharing the MD trajectories of the proteins used in this work. This work is supported in part by the National Science Foundation (CHE-1265929, CHE-1738990, and PHY-1427654 to C.C., and CCF-1423304 to L.K.), the Welch Foundation (C-1570 to C.C.), and Rice University funds. J.R.A. is supported by a training fellowship from the Gulf Coast Consortia on the Training Program in Biomedical Informatics, National Library of Medicine T15LM007093. F.N. is a postdoctoral researcher in the Rice University Academy of Fellows. This research used resources of the Oak Ridge Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC05-00OR22725, equipment that is supported in part by the Data Analysis and Visualization Cyberinfrastructure funded by NSF under Grant OCI 0959097, as well as on equipment that is supported by the Cyberinfrastructure for Computational Research funded by NSF under Grant CNS 0821727.

[1] M. Shirts and V. S. Pande, Science **290**, 1903 (2000).
[2] I. Buch, M. J. Harvey, T. Giorgino, D. P. Anderson, and G. De Fabritiis, J. Chem. Inf. Model **50**, 397 (2010).
[3] D. E. Shaw, J. Grossman, J. A. Bank, B. Batson, J. A. Butts, J. C. Chao, M. M. Deneroff, R. O. Dror, A. Even, C. H. Fenton, *et al.*, in *Proceedings of the international conference for high performance computing, networking, storage and analysis* (IEEE Press, 2014) pp. 41–53.
[4] D. Hamelberg, J. Mongan, and J. A. McCammon, J. Chem. Phys. **120**, 11919 (2004).

[5] Y. Sugita and Y. Okamoto, Chem. Phys. Lett. **314**, 141 (1999).
[6] A. Laio and F. L. Gervasio, Rep. Prog. Phys. **71**, 126601 (2008).
[7] D. M. Zuckerman and T. B. Woolf, Phys. Rev. E **63**, 016702 (2000).
[8] L. Donati, C. Hartmann, and B. G. Keller, J. Chem. Phys. **146**, 244112 (2017).
[9] C. Xing and I. Andricioaei, J. Chem. Phys. **124**, 034110 (2006).
[10] L. Donati and B. G. Keller, J. Chem. Phys. **149**, 072335 (2018).
[11] N. Singhal and V. S. Pande, J. Chem. Phys. **123**, 204909 (2005).
[12] G. R. Bowman, D. L. Ensign, and V. S. Pande, J. Chem. Theory Comput. **6**, 787 (2010).
[13] J. K. Weber and V. S. Pande, J. Chem. Theory Comput. **7**, 3405 (2011).
[14] S. Doerr and G. De Fabritiis, J. Chem. Theory Comput. **10**, 2064 (2014).
[15] J. Preto and C. Clementi, Phys. Chem. Chem. Phys. **16**, 19181 (2014).
[16] S. Doerr, M. Harvey, F. Noé, and G. De Fabritiis, J. Chem. Theory Comput. **12**, 1845 (2016).
[17] D. Lecina, J. F. Gilabert, and V. Guallar, Sci. Rep. **7**, 8466 (2017).
[18] Z. Shamsi, A. S. Moffett, and D. Shukla, Sci. Rep. **7**, 12700 (2017).
[19] M. I. Zimmerman and G. R. Bowman, J. Chem. Theory Comput. **11**, 5747 (2015).
[20] B. Trendelkamp-Schroer and F. Noé, Phys. Rev. X **6**, 011009 (2016).
[21] N. Plattner, S. Doerr, G. De Fabritiis, and F. Noé, Nat. Chem. **9**, 1005 (2017).
[22] S. Röblitz and M. Weber, Adv. Data. Anal. Classi. **7**, 147 (2013).
[23] A. Dickson and C. L. Brooks, J. Phys. Chem. B **118**, 3532 (2014).
[24] M. C. Zwier, J. L. Adelman, J. W. Kaus, A. J. Pratt, K. F. Wong, N. B. Rego, E. Suárez, S. Lettieri, D. W. Wang, M. Grabe, D. M. Zuckerman, and L. T. Chong, J. Chem. Theory Comput. **11**, 800 (2015).
[25] M. Biswas, B. Lickert, and G. Stock, J. Phys. Chem. B **122**, 5508 (2018).
[26] M. A. Rohrdanz, W. Zheng, and C. Clementi, Annu. Rev. Phys. Chem. **64**, 295 (2013).
[27] F. Noé and C. Clementi, Curr. Opin. Struct. Biol. **43**, 141 (2017).
[28] J.-H. Prinz, H. Wu, M. Sarich, B. Keller, M. Senne, M. Held, J. D. Chodera, C. Schütte, and F. Noé, J. Chem. Phys. **134**, 174105 (2011).
[29] B. E. Husic and V. S. Pande, J. Am. Chem. Soc. **140**, 2386 (2018).
[30] G. R. Bowman, V. Pande, and F. Noé, in *Advances in Experimental Medicine and Biology*, Vol. 797 (Springer, 2014).
[31] N.-V. Buchete and G. Hummer, J. Phys. Chem. B **112**, 6057 (2008).
[32] C. Schütte, A. Fischer, W. Huisinga, and P. Deuflhard, J. Comp. Phys. **151**, 146 (1999).
[33] R. R. Coifman, S. Lafon, A. B. Lee, M. Maggioni, B. Nadler, F. Warner, and S. W. Zucker, Proc. Natl. Acad. Sci. **102**, 7426 (2005).
[34] M. A. Rohrdanz, W. Zheng, M. Maggioni, and C. Clementi, J. Chem. Phys. **134**, 03B624 (2011).
[35] W. Zheng, B. Qi, M. A. Rohrdanz, A. Caflisch, A. R. Dinner, and C. Clementi, J. Phys. Chem. B **115**, 13065 (2011).
[36] L. Boninsegna, G. Gobbo, F. Noé, and C. Clementi, J. Chem. Theory Comput. **11**, 5947 (2015).
[37] B. Peters and B. L. Trout, J. Chem. Phys. **125**, 054108 (2006).
[38] S. V. Krivov and M. Karplus, Proc. Natl. Acad. Sci. **105**, 13841 (2008).
[39] A. Mardt, L. Pasquali, H. Wu, and F. Noé, Nat. Comm. **9** (2018).
[40] C. Wehmeyer and F. Noé, J. Chem. Phys. **148**, 241703 (2018).
[41] J. M. L. Ribeiro, P. Bravo, Y. Wang, and P. Tiwary, J. Chem. Phys. **149** (2018).
[42] M. Wieczorek, J. Sticht, S. Stolzenberg, S. Gnther, C. Wehmeyer, Z. El Habre, M. lvaro Benito, F. Noé, and C. Freund, Nat. Comm. **7** (2016).
[43] N. Plattner, S. Doerr, G. De Fabritiis, and F. Noé, Nat. Chem. **9**, 1005 (2017).
[44] K. Kohlhoff, D. Shukla, M. Lawrenz, G. Bowman, D. Konerding, D. Belov, R. Altman, and V. Pande, Nat. Chem. **6**, 15 (2014).
[45] K. Lindorff-Larsen, S. Piana, R. O. Dror, and D. E. Shaw, Science **334**, 517 (2011).
[46] F. Noé, R. Banisch, and C. Clementi, J. Chem. Theory Comput. **12**, 5620 (2016).
[47] F. Noé and C. Clementi, J. Chem. Theory Comput. **11**, 5002 (2015).
[48] G. Pérez-Hernández, F. Paul, T. Giorgino, G. De Fabritiis, and F. Noé, J. Chem. Phys. **139**, 07B604_1 (2013).
[49] C. R. Schwantes and V. S. Pande, J. Chem. Theory Comput. **9**, 2000 (2013).
[50] M. K. Scherer, B. Trendelkamp-Schroer, F. Paul, G. Perez-Hernandez, M. Hoffmann, N. Plattner, C. Wehmeyer, J.-H. Prinz, and F. Noé, J. Chem. Theory Comput. **11**, 5525 (2015).
[51] B. O. Koopman, Proc. Natl. Acad. Sci. **17**, 315 (1931).
[52] M. O. Williams, C. W. Rowley, and I. G. Kevrekidis, J. Comput. Dynam. **2**, 247 (2015).
[53] M. O. Williams, I. G. Kevrekidis, and C. W. Rowley, J. Nonlinear Sci. **25**, 1307 (2015).
[54] Q. Li, F. Dietrich, E. M. Bollt, and I. G. Kevrekidis, Chaos **27** (2017).
[55] H. Wu, F. Nüske, F. Paul, S. Klus, P. Koltai, and F. Noé, J. Chem. Phys. **146**, 154104 (2017).
[56] F. Nüske, H. Wu, J.-H. Prinz, C. Wehmeyer, C. Clementi, and F. Noé, J. Chem. Phys. **146**, 094104 (2017).
[57] P. G. Hoel, S. C. Port, and C. J. Stone, *Introduction to Stochastic Processes* (Waveland Press, 1972).
[58] D. Hamelberg, C. A. F. de Oliveira, and J. A. McCammon, J. Chem. Phys. **127**, 10B614 (2007).
[59] Y. Naritomi and S. Fuchigami, J. Chem. Phys. **134**, 02B617 (2011).