Trajectory stratification of stochastic dynamics

Jeremy O. B. Tempkin,[1, 2] Brian Van Koten,[3] Jonathan C. Mattingly,[4] Aaron R. Dinner,[1, 2] and Jonathan Weare[1, 3]

[1]*James Franck Institute, The University of Chicago, Chicago, Illinois 60637, USA*
[2]*Department of Chemistry, The University of Chicago, Chicago, Illinois 60637, USA*
[3]*Department of Statistics, The University of Chicago, Chicago, Illinois 60637, USA*
[4]*Departments of Mathematics and Statistical Science,*
Duke University, Durham, North Carolina 27708, USA
(Dated: November 1, 2016)

We present a general mathematical framework for trajectory stratification for simulating rare events. Trajectory stratification involves decomposing trajectories of the underlying process into fragments limited to restricted regions of state space (strata), computing averages over the distributions of the trajectory fragments within the strata with minimal communication between them, and combining those averages with appropriate weights to yield averages with respect to the original underlying process. Our framework reveals the full generality and flexibility of trajectory stratification, and it illuminates a common mathematical structure shared by existing algorithms for sampling rare events. We demonstrate the power of the framework by defining strata in terms of both points in time and path-dependent variables for efficiently estimating averages that were not previously tractable.

I. INTRODUCTION

Computer simulation is a powerful tool for the study of physical processes. However, for many systems, the most interesting events are those that occur most infrequently and are therefore very difficult to observe by direct numerical integration of the equations governing the dynamics. For example, in chemistry, the conformational changes responsible for the function of many molecules and, in climate science, extreme events like severe droughts and violent hurricanes, occur on timescales orders of magnitude longer than the timestep for numerical integration. This basic observation has motivated the development of numerous techniques aimed at enhancing the sampling of rare events of interest without sacrificing statistical fidelity (see [1] for an account within the context of molecular simulation).

In this article, we depart from standard enhanced sampling approaches and develop a general mathematical and computational framework for the estimation of statistical averages involving rare trajectories of stochastic processes. Our approach can be viewed as a form of stratified sampling, long a cornerstone of experimental design in statistics (e.g., [2]). In stratified sampling, a population is divided into subgroups (strata), averages within those strata are computed separately, and then averages over the entire state space are assembled as weighted sums of the strata averages. Stratification also has a long history in computer simulations of condensed-phase systems as umbrella sampling (US) [1, 3–6]. The key idea behind any stratified sampling strategy is that, when the strata are chosen appropriately, their statistics can be obtained accurately with relatively low effort and combined to estimate the average of interest with (much) less overall effort than directly sampling the stochastic process to the same statistical precision. Here we show that the trajectories of an arbitrary discrete-time Markov process (including many dynamics with memory, so long as they

can be written as a suitable mapping) can also be stratified: they can be decomposed into fragments restricted to regions of trajectory space (strata), averages over the distributions of trajectory fragments within the strata can be computed with limited communication between them, and those averages can be combined in a weighted fashion to yield a very broad range of statistics that characterize the dynamics.

These basic features are at the core of the existing nonequilibrium umbrella sampling (NEUS) method [7–9], which forms the starting point for our development. NEUS was originally introduced to estimate stationary averages with respect to a given, possibly irreversible, stochastic process [7]. Starting in [8, 9] it was observed that the general NEUS approach was applicable to certain dynamical averages as well. At its most basic level, NEUS relies on duplication of states in rarely visited regions of space and subsequent forward evolution of the duplicated states. In this way it is similar to a long list of so-called "trajectory splitting" techniques [10–18] that are also able to compute averages of dynamic quantities. Like NEUS, splitting techniques also often involve a decomposition of state space into regions. Unlike NEUS however, in most splitting techniques bias is removed through the use of a separate weight factor for each individual sample (rather than for an entire region), and the computational effort expended in each region is not controlled exactly. What makes the NEUS method unique among splitting techniques is that it is also a trajectory stratification strategy.

Our goal in this article is to provide a clear and general mathematical framework for trajectory stratification that builds upon the NEUS method. In the process we clearly delineate the range of statistics that can be estimated by NEUS, including more general quantities than previously computed. Our analysis of the underlying mathematical structure of US [19, 20] has already facilitated the derivation of a central limit theorem for US and a detailed

understanding of its error properties. Here, our framework reveals unanticipated connections between the equilibrium and nonequilibrium US methods and places the nonequilibrium algorithm within the well studied family of stochastic approximation methods [21]. The analysis leads to a practical scheme that departs dramatically from currently available alternatives. We demonstrate the use of trajectory stratification to compute a hitting time distribution as well as to compute the expectation of a path-dependent functional that gives the relative normalization constants for two arbitrary, user-specified unnormallized probability densities.

II. A UNIFIED FRAMEWORK

In this section we present a framework that reveals the unified structure underlying umbrella sampling in both the equilibrium and nonequilibrium case. In Section II A, we review the equilibrium approach [19, 20] to introduce terminology and the central eigenproblem in a context where the analogies to traditional umbrella sampling descriptions [1, 3–6] are readily apparent. In Section II B, we present the nonequilibrium version of the algorithm and show how this interpretation results in a flexible scheme for computing dynamic averages. As for its equilibrium counterpart, an eigenproblem lies at the core of the nonequilibrium method. This eigenproblem however, involves a matrix that depends on the desired eigenvector, introducing the need for a self-consistent iteration. In Section III, we give a precise description of the fixed-point problem solved by this iteration and show that the algorithm is an example of a stochastic approximation strategy [21]. In Section IV we specialize our development to the context of steady-state averages that motivated the original development of NEUS [7].

A. Averages with respect to a specified density

Our presentation in this section follows [19]. We view umbrella sampling as a method to compute averages of the form

$$\int_{x \in \mathbb{R}^d} f(x)\pi(dx), \tag{1}$$

where π is a known probability distribution and d is the dimension of the underlying system (e.g., the total number of position coordinates for all atoms in a molecular system). For example, π might be the canonical distribution, $\pi(dx) \propto e^{-\beta V(x)}dx$ where V is a potential energy function, β is an inverse temperature, and f might be 1 on some set A and 0 elsewhere. In this case, $-\beta^{-1} \log \int f(x)\pi(dx)$ can be regarded as the free energy of the set A.

Note that in our notation π is a probability measure on $\mathbb{R}^d$ and dx is an infinitesimal volume element in $\mathbb{R}^d$. If the distribution π has a density function $p(x)$ then $\pi(A) = \int_{x \in A} p(x)dx$ and, in particular, $\pi(dx) = p(x)dx$. This more general notation is useful when we move to our description of the nonequilibrium umbrella sampling scheme. As an aid to the reader, we choose to introduce it in the simpler setting of this section.

Consistent with traditional implementations of US [1, 4], we divide the computation of the average in (1) into a series of averages over local subsets of space. To describe that process mathematically we define a set of n probability distributions

$$\pi_j(dx) = \frac{\psi_j(x)\pi(dx)}{\int_{y \in \mathbb{R}^d} \psi_j(y)\pi(dy)}, \tag{2}$$

where the ψ_j are non-negative user defined functions satisfying $\sum_{j=1}^n \psi_j(x) = 1$ for all x (this last requirement is relaxed in [19]). For example, one might choose $\psi_j(x) = \mathbf{1}_{A_j}(x)/\sum_{\ell=1}^n \mathbf{1}_{A_\ell}(x)$, where the A_j are a collection of sets covering the space to be sampled, and, for any set A_j, the function $\mathbf{1}_{A_j}(x)$ is 1 if $x \in A_j$ and 0 otherwise. Like this one, typical choices of ψ_j concentrate (or restrict) probability under π_j in smaller regions of space (relative to π itself) with the goal of eliminating or reducing barriers to efficient sampling associated with π.

To see that averages with respect to π can be reconstructed using averages with respect to the π_j, note that we can decompose the expectation in (1) as a weighted sum of expectations over the local distributions defined in (2). That is

$$\int_{x \in \mathbb{R}^d} f(x)\pi(dx) = \sum_{j=1}^n z_j \langle f \rangle_j, \tag{3}$$

where we have defined the quantities

$$z_j = \int_{x \in \mathbb{R}^d} \psi_j(x)\pi(dx) \tag{4}$$

and

$$\langle f \rangle_j = \int_{x \in \mathbb{R}^d} f(x)\pi_j(dx). \tag{5}$$

Here z_j is the statistical weight associated with each distribution π_j and $\langle f \rangle_j$ are the averages of the observable f against π_j. From (3) we see that if we can sample from the π_j and compute the z_j then we can compute averages with respect to π. Since π_j is known explicitly in this case, it can be sampled by standard means (e.g., Langevin dynamics or Metropolis Monte Carlo [1]).

Our key observation underpinning the equilibrium umbrella sampling method is that the z_j themselves are functions of averages with respect to the local distributions π_j:

$$z_j = \sum_{i=1}^n z_i F_{ij} \quad \text{and} \quad \sum_{j=1}^n z_j = 1, \tag{6}$$

where

$$F_{ij} = \int_{x \in \mathbb{R}^d} \psi_j(x)\pi_i(dx). \tag{7}$$

The matrix F is stochastic (i.e., has non-negative entries with rows that sum to 1) and (6), which is written in matrix-vector form as

$$z^{\mathrm{T}}F = z^{\mathrm{T}} \quad \text{and} \quad \sum_{j=1}^{n} z_j = 1, \qquad (8)$$

is an eigenproblem that can be solved easily for the vector z.

We now have a stratification scheme for computing the target average in (1) by sampling from the distributions π_j. Operationally, the main steps are as follows.

1. Assemble F defined in (7) (or the alternative in Appendix A below) and $\langle f \rangle_j$ defined in (5) by sampling from π_j defined in (2).

2. Solve eigenvector equation (8) for z defined in (4).

3. Compute the desired expectation via (3).

The efficiency of this *equilibrium* US scheme has been analyzed in detail elsewhere [19, 20]. Roughly, the benefit of US is due to the facts that averages with respect to the π_j are often sufficient to solve for all desired quantities, and one can choose ψ_j so that averages with respect to the π_j converge much more quickly than averages with respect to π itself. It is this basic philosophy that we extend in Section II B to the computation of dynamic averages.

B. Averages with respect to a given Markov process

The mathematical description of the nonequilibrium umbrella sampling scheme that follows reveals how the stratification strategy developed for the equilibrium case in Section II A can be extended to compute nearly arbitrary dynamic statistics. Our interest in this section is computing averages over trajectories of some specified Markov process, $X^{(t)}$. This process can be time-inhomogenous, i.e., given the value of $X^{(t)}$, the distribution of $X^{(t+1)}$ can depend on the value of t. We compute averages of trajectories evolved up to a first exit time of the process, $(t, X^{(t)})$, from a user specified set of times and positions, D—i.e., trajectories terminate when they first leave the set D. We consider averages over trajectories of $X^{(t)}$ run until time

$$\tau = \min\{t > 0 : (t, X^{(t)}) \notin D\} \qquad (9)$$

for a set $D \in \mathbb{N} \times \mathbb{R}^d$. The averages are of the form

$$\mathbf{E}\left[\sum_{t=0}^{\tau-1} f(t, X^{(t)})\right]. \qquad (10)$$

We note that the average in (10) is not completely general, in order to streamline the developments below. Without any modification, we can compute averages similar to (10) but with the argument $(t, X^{(t)})$ in

the definitions of τ and f replaced by $(t, X^{(t-1)}, X^{(t)})$. On the other hand, expectations with $(t, X^{(t)})$ replaced by $(t, X^{(t-m)}, \ldots, X^{(t-1)}, X^{(t)})$ for $m \geq 2$ cannot be obtained immediately. These and many more general expectations can, however, be accommodated by applying the algorithm to an enlarged process (e.g., $(t, X^{(t-m)}, \ldots, X^{(t-1)}, X^{(t)})$) at the cost of storing copies (as described below) of the enlarged process. For many expectations, this cost is quite manageable. Finally, we require that $\mathbf{E}[\tau] < \infty$. The limit $\tau \to \infty$ will be considered in Section IV.

Below we show that expectations of time-dependent functions can be decomposed as a weighted sum of expectations computed over restricted subsets of the full space and, in turn, how the statistical weights can be computed as expectations over these subsets, mirroring the basic structure of the equilibrium scheme described in Section II A. However, as we discuss in Section III, the algorithm for computing these local expectations departs significantly from the equilibrium case because their form is not known *a priori* in the nonequilibrium setting.

1. The index process

The US scheme in Section II A used the basis functions ψ_j to stratify the sampling of the distribution π by decomposing averages with respect to π into averages with respect to the more easily sampled π_j. To arrive at an analogous partitioning of state space for the nonequilibrium case, we first need to introduce an *index process* $J^{(t)}$ that takes values in $\{1, 2, \ldots, n\}$ and (roughly) labels the point $(t, X^{(t)})$ in time and space, $\mathbb{N} \times \mathbb{R}^d$. Our objective is to generate fragments of trajectories of $X^{(t)}$ consistent with specific values of $J^{(t)}$ thereby breaking the coupled process $(X^{(t)}, J^{(t)})$ into separate regions corresponding to a given value of $J^{(t)}$.

The idea of discretizing a process $X^{(t)}$ according to the value of some user-specified index process is not new in computational statistical mechanics. For example, in our notation, given a partition of state space $A_1, A_2, \ldots, A_n$, the Milestoning procedure [22] and some Markov State Modeling procedures [23] correspond to an index process that marks the pairs of sets (A_i, A_j) for $i \neq j$ between which $X^{(t)}$ last transitioned. In the Milestoning method, the pairs of sets are considered unordered, so that a transition from A_j to A_i immediately following a transition from A_i to A_j does not correspond to a change in $J^{(t)}$ and $J^{(t)}$ can assume $n = (m \text{ choose } 2)$ distinct values. The original presentation of NEUS on the other hand corresponds to a process $J^{(t)}$ which marks the index of the set A_j containing $X^{(t)}$. For accurate results, the Milestoning procedure requires that the index process $J^{(t)}$ itself be Markovian. Even under the best circumstances, that assumption is only expected to hold approximately. It is not required by the NEUS algorithm. Our presentation below reveals the full flexibility in the choice of $J^{(t)}$

within NEUS. That flexibility is essential in the generalized setting of this article.

In the developments below we require that $J^{(t)}$ is chosen so that the *joint process* $(X^{(t)}, J^{(t)})$ is Markovian. This assumption allows that trajectories can be continued beyond a single transition event (before τ) without additional information about the history of $X^{(t)}$ or $J^{(t)}$. We do not assume that $J^{(t)}$ alone is Markovian and in general it will not be. Our assumption implies no practical restriction on the underlying Markov process $X^{(t)}$. When $X^{(t)}$ is non-Markovian, additional variables can often be appended to $X^{(t)}$ to yield a new Markov process to which the developments below can be applied. A version of this idea is applied in Section V C where we append a variable representing a nonequilibrium work to an underlying Markov process.

2. The eigenproblem

Given a specific choice of index process $J^{(t)}$, the nonequilibrium umbrella sampling algorithm stratifies trajectories of $X^{(t)}$ according to their corresponding values of $J^{(t)}$. That is, for each possible value of the index process, NEUS generates segments of trajectories of $X^{(t)}$ between the times that $J^{(t)}$ transitions to and from $J = i$. To make this idea more precise, we need to carefully describe the distribution sampled by these trajectory fragments:

$$\pi_j(t, dx) = \frac{\mathbf{P}\left[t < \tau, X^{(t)} \in dx, J^{(t)} = j\right]}{z_j}, \qquad (11)$$

where

$$z_j = \sum_{t=0}^{\infty} \mathbf{P}\left[t < \tau, J^{(t)} = j\right]. \qquad (12)$$

For each j, π_j is the distribution of time and position pairs $(t, X^{(t)})$ conditioned on $J^{(t)} = j$ and $t < \tau$. We call the π_j *restricted distributions*. We have reused the notations π_j and z_j from our account of the equilibrium umbrella sampling scheme to emphasize the analogous roles played by those objects in both sections. Note that here we are treating time as an additional random variable. Also note that in these definitions as well as in the formulas below, $\mathbf{P}$ and $\mathbf{E}$ represent probabilities and expectations with respect to the original, unbiased $X^{(t)}$ and $J^{(t)}$. We assume that $z_j > 0$ for all j since we can remove the index j from consideration if $z_j = 0$. The z_j are all finite because $\sum_{j=1}^{n} z_j = \mathbf{E}\left[\tau\right]$, which we assume is finite.

Observe that

$$\mathbf{E}\left[\sum_{t=0}^{\tau-1} f(t, X^{(t)})\right] = \sum_{t=0}^{\infty} \mathbf{E}\left[f\left(t, X^{(t)}\right), t < \tau\right]$$

$$= \sum_{j=1}^{n} \sum_{t=0}^{\infty} \int_{x \in \mathbb{R}^d} f(t, x)$$

$$\times \mathbf{P}\left[t < \tau, X^{(t)} \in dx, J^{(t)} = j\right]$$

$$= \sum_{j=1}^{n} z_j \langle f \rangle_j,$$

where

$$\langle f \rangle_j = \sum_{t=0}^{\infty} \int_{x \in \mathbb{R}^d} f(t, x) \pi_j(t, dx). \qquad (13)$$

Thus we have a decomposition of (10) analogous to the decomposition of (1) in (3). Also as in the equilibrium case, the z_j can be computed from averages with respect to the π_j. To see this, observe that for any t we can write

$$\sum_{i=1}^{n} \mathbf{P}\left[t + 1 < \tau, J^{(t+1)} = j, J^{(t)} = i\right]$$

$$= \mathbf{P}\left[t + 1 < \tau, J^{(t+1)} = j\right]. \qquad (14)$$

Summing this expression over t we obtain

$$\sum_{i=1}^{n} \sum_{t=0}^{\infty} \mathbf{P}\left[t + 1 < \tau, J^{(t+1)} = j, J^{(t)} = i\right]$$

$$= \sum_{t=0}^{\infty} \mathbf{P}\left[t < \tau, J^{(t)} = j\right] - \mathbf{P}\left[J^{(0)} = j\right]. \qquad (15)$$

These expressions are all bounded by $\mathbf{E}\left[\tau\right]$ and are therefore finite. Expression (15) can be rewritten as an affine eigenequation:

$$z^{\mathrm{T}} G + a^{\mathrm{T}} = z^{\mathrm{T}}, \qquad (16)$$

where z is defined in (12),

$$G_{ij} = \frac{\sum_{t=0}^{\infty} \mathbf{P}\left[t + 1 < \tau, J^{(t+1)} = j, J^{(t)} = i\right]}{z_i}, \qquad (17)$$

and

$$a_j = \mathbf{P}\left[J^{(0)} = j\right]. \qquad (18)$$

Equation (16) is the analog of (8) in Section II A. Here, the matrix element G_{ij} stores the expected number of transitions between the values of $J^{(t)}$, normalized by the expected number of time steps with $J = i$. Note that the matrix G is substochastic; that is, it has non-negative entries and rows that sum to a number less than or equal to one.

To complete the analogy with the umbrella sampling schemes described in Section II A, we need to show that the elements of the matrix G are expressible as expectations over the π_j. Indeed,

$$
G_{ij} = \frac{1}{z_i} \int_{x \in \mathbb{R}^d} \sum_{t=0}^{\infty} \mathbf{P}_{t,x,i} \left[t+1 < \tau,\ J^{(t+1)} = j \right] \quad (19)
$$

$$
\times\ \mathbf{P} \left[t < \tau,\ X^{(t)} \in dx,\ J^{(t)} = i \right]
$$

$$
= \sum_{t=0}^{\infty} \int_{x \in \mathbb{R}^d} \mathbf{P}_{t,x,i} \left[t+1 < \tau,\ J^{(t+1)} = j \right] \pi_i(t, dx)
$$
$$
(20)
$$

where $\mathbf{P}_{t,x,i}$ is used to denote probabilities with respect to X initialized at time and position (t,x) and conditioned on $J^{(t)} = i$ and $t < \tau$. Note that in the first line we have appealed to the Markovian assumption on $(X^{(t)}, J^{(t)})$. Had we instead assumed that $J^{(t)}$ alone was Markovian, we could have ignored the x dependence in (20).

Just as for the umbrella sampling algorithm described in Section II A, we arrive at a procedure for computing (10) via stratification:

1. Assemble G_{ij} defined in (17) and $\langle f \rangle_j$ defined in (13) by sampling from the π_j defined in (11).

2. Solve the affine eigenvector equation (16) for z defined in (12).

3. Compute

$$
\mathbf{E} \left[\sum_{t=0}^{\tau-1} f(t, X^{(t)}) \right] = \sum_{j=1}^{n} z_j \langle f \rangle_j
$$

as in (13).

Relative to the scheme in Section II A, sampling the restricted distributions π_j requires a more complicated procedure. This is the subject of Section III.

Rapid convergence of the scheme rests on the choice of $J^{(t)}$. Roughly, one should choose the index process so that the variation in estimates of the required averages with respect to the π_j (e.g., estimates of the G_{ij}) are small. In practice, this requires that transitions between values of $J^{(t)}$ are frequent, which is the analog of selecting the biases in equilibrium US to limit the range of the free energy over each subset of state space (see [19, 20]). We defer further specification of $J^{(t)}$ to Section V, where we describe this and other important implementation details in the context of particular applications.

III. A GENERAL NEUS FIXED-POINT ITERATION

In this section we present a detailed algorithm for computing (10) by the stratification approach outlined in Section II B. To accomplish this one must be able to generate samples from the restricted distributions $\pi_j(t, dx)$.

In NEUS, the restricted distributions are sampled by introducing a set of Markov processes $Y_j^{(r)}$ whose trajectories consist of sequential segments of the process $(t, X^{(t)})$ such that $J = j$. We use the time variable r to distinguish between the time associated with the underlying process $(t, X^{(t)})$ and the time r associated with the process $Y_j^{(r)}$. The trajectories of $Y_j^{(r)}$ are generated by integrating the underlying process $(X^{(t)}, J^{(t)})$ forward in time until a segment leaves the state $J = j$ or reaches the stopping time τ at which point the dynamics are stopped and restarted at a random time s and position y drawn from a distribution $\bar{\pi}_j(s, dy)$. For $Y_j^{(r)}$ to preserve the restricted distribution $\pi_j(t, dx)$, $\bar{\pi}_j(s, dy)$ must be the distributions of times s and positions y at which the process $(X^{(t)}, J^{(t)})$ transitions from a state $J^{(s-1)} = i$ with $i \neq j$ to state $J^{(s)} = j$ (see Appendix B). We call these distributions the *flux distributions*. The $Y_j^{(r)}$ process is illustrated in Figure 1 for a particular choice of index process.

In general, the flux distributions $\bar{\pi}_j(s, dy)$ for which $Y_j^{(r)}$ preserves the restricted distribution $\pi_j(t, dx)$ are not known *a priori* and must be computed approximately. In Section III A, we define the flux distributions carefully and express them as a weighted sum of *conditional* flux distributions $\gamma_{ij}(s, dy)$ which are a set of time and position distributions conditioned on the process $(X^{(t)}, J^{(t)})$ switching from a specific $J = i$ to $J = j$. In order to construct an estimate of $\bar{\pi}_j(s, dy)$ from estimates of the $\gamma_{ij}(s, dy)$ it is necessary to compute the flux $z_i G_{ij}$ of the process $(X^{(t)}, J^{(t)})$ from each restriction with $J = i$, $i \neq j$. The elements of the desired matrix G (and the corresponding weights z computed from (16)) are computed from expectations over the restricted distribution $\pi_j(t, dx)$ and sampled by integrating the process $Y_j^{(r)}$. This relationship between the weights z_j and the flux distributions $\bar{\pi}_j(s, dy)$ motivates a self-consistent iteration in which we solve for the flux distributions $\bar{\pi}_j(s, dy)$ and the matrix G, yielding successively better approximations of the flux distributions and the matrix G simultaneously. In Section III B, we derive a fixed-point equation that expresses the relationship between the flux distributions and the transition matrix G that motivates this self-consistent iteration. In Section III C, we describe the complete NEUS algorithm in detail and interpret it as a stochastic approximation algorithm [21] for solving the fixed-point equation derived in Section III B.

A. The flux distributions

Before deriving the fixed-point problem and the corresponding stochastic approximation algorithm, we define the flux distributions $\bar{\pi}_j(s, dy)$ precisely. First, define the time of the ℓth change in the value of $J^{(t)}$ for a given re-
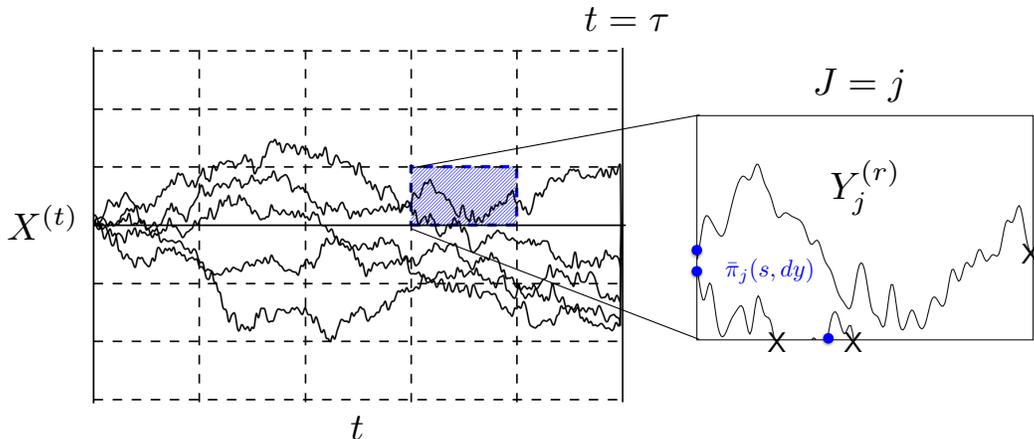
FIG. 1. Illustration of the stratification of a process $(X^{(t)}, J^{(t)})$ (black lines, left panel) via the scheme outlined in Section II B. The restricted distributions corresponding to each value of the index process $J^{(t)}$ are outlined as discrete regions of the $(t, X^{(t)})$ space (left panel, black dashed lines). In this depiction, the value of $J^{(t)}$ corresponds to the current cell containing $(t, X^{(t)})$ within a rectangular grid of time and position. Each of the restricted distributions $\pi_j(t, dx)$ are sampled by integrating a locally restricted dynamics $Y_j^{(r)}$ (right panel, black lines). The $Y_j^{(r)}$ process is generated by integrating sequential fragments of the unbiased process $(X^{(t)}, J^{(t)})$ corresponding to a particular fixed value of $J = j$ (left panel). As each fragment transitions from $J = i$ to $J = j$ with $j \neq i$, the dynamics are stopped and restarted at a time and point (s, y) (left panel, blue dots) drawn from the flux distribution $\bar{\pi}_j(s, dy)$.

alization of the coupled process $(X^{(t)}, J^{(t)})$ as

$$S^{(\ell)} = \min\{s > S^{(\ell-1)} : J^{(s)} \neq J^{(S^{(\ell-1)})}\} \qquad (21)$$

for $\ell > 0$, and $S^{(0)} = 0$. Note that for $i \neq j$,

$$z_i G_{ij} = \sum_{\ell=0}^{\infty} \mathbf{P}\left[S^{(\ell+1)} < \tau, \, J^{(S^{(\ell+1)})} = j, \, J^{(S^{(\ell)})} = i\right] \qquad (22)$$

is the net probability flux between values of the process $J^{(t)}$ before time τ.

Now let $\bar{\pi}_j(s, dy)$ be the distribution of time and position pairs $(S^{(\ell)}, X^{(S^{(\ell)})})$ conditioned on $J^{(S^{(\ell)})} = j$. Specifically, define

$$\bar{\pi}_j(s, dy) \propto \sum_{\ell=0}^{\infty} \mathbf{P}\left[S^{(\ell)} = s, \, s < \tau, \, X^{(s)} \in dy, \, J^{(s)} = j\right]. \qquad (23)$$

Instead of working directly with the flux distributions, we find it convenient to express both the fixed-point problem and the algorithm in terms of the probability distribution of time and position pairs $(t, X^{(t)})$ conditioned on observing a transition from $J = i$ to $J = j$ at time t, i.e., in terms of

$$\gamma_{ij}(s, dy) \propto \int_{x \in \mathbb{R}^d} \mathbf{P}_{s-1,x,i}[s < \tau, \, X^{(s)} \in dy, \, J^{(s)} = j]$$
$$\times \pi_i(s-1, dx) \quad (24)$$

which is defined only for $s > 0$. To simplify notation, we let γ denote the set of all distributions γ_{ij}. The following simple but crucial identity relates γ to the flux

distributions $\bar{\pi}_j$:

$$\bar{\pi}_j(s, dy)$$
$$= \frac{1}{z_j(1 - G_{jj})} \begin{cases} \sum_{i \neq j} z_i \, G_{ij} \, \gamma_{ij}(s, dy), & \text{if } s > 0 \\ a_j \, \mathbf{P}[X^{(0)} \in dy \mid J^{(0)} = j] & \text{if } s = 0. \end{cases} \qquad (25)$$

The normalization is the total flux into $J = j$, $z_j(1 - G_{jj}) = \sum_{\ell=0}^{\infty} \mathbf{P}[J(S^{(\ell)}), S^{(\ell)} < \tau]$. The $s > 0$ term is the contribution from transitions into state $J = j$ from the neighboring state $J = i$, and the $s = 0$ term accounts for the initial $t = 0$ contribution of the underlying process when $J = j$. We emphasize that both the fixed-point problem and the iteration that we define below could be expressed in terms of the flux distributions $\bar{\pi}_j$ instead of γ. We choose to express them in terms of γ because the resulting formalism more naturally captures the implementation of the method used to generate our numerical results in Section V.

B. The fixed-point problem

We now derive the fixed-point problem. Our goal is to find an expression of the form

$$(\mathcal{G}(G, \gamma), \Gamma(G, \gamma)) = (G, \gamma) \qquad (26)$$

that characterizes the desired matrix G and collection of probability measures γ as the fixed-point of a pair of

maps $\mathcal{G}(\tilde{G}, \tilde{\gamma})$ and $\Gamma(\tilde{G}, \tilde{\gamma})$ that take as arguments approximations $\tilde{G}$ of G and $\tilde{\gamma}$ of γ and return, respectively, a new substochastic matrix and a new collection of probability measures.

To this end, we define functions mapping $\tilde{G}$ and $\tilde{\gamma}$ to approximations of the flux distributions $\bar{\bar{\pi}}_j$ and the restricted distributions π_j. We denote these functions by the corresponding capital letters $\bar{\bar{\Pi}}_j$ and Π_j. Based on (25) and (16), we define

$$\bar{\bar{\Pi}}_j(s, dy; \tilde{G}, \tilde{\gamma})$$
$$= \frac{1}{\tilde{z}_j\,(1 - \tilde{G}_{jj})} \begin{cases} \sum_{i \neq j} \tilde{z}_i \tilde{G}_{ij}\,\tilde{\gamma}_{ij}(s, dy) & \text{if } s > 0, \\ a_j\,\mathbf{P}[X^{(0)} \in dy \mid J^{(0)} = j] & \text{if } s = 0, \end{cases}$$
$$\tag{27}$$

where $\tilde{z}$ solves the equation $\tilde{z}^{\mathrm{T}} = \tilde{z}^{\mathrm{T}}\tilde{G} + a^{\mathrm{T}}$. To motivate the definition above, we observe that for the exact values G and γ, $\bar{\bar{\pi}}_j(s, dy) = \bar{\bar{\Pi}}_j(s, dy; G, \gamma)$ by (25). Moreover, given $\tilde{G}$ and samples from $\tilde{\gamma}$, one can generate samples from $\bar{\bar{\Pi}}_j(s, dy; \tilde{G}, \tilde{\gamma})$; see Section III C. This is crucial in developing a practical algorithm to solve the fixed-point problem.

Similarly, we define

$$\Pi_j(t, dx; \tilde{G}, \tilde{\gamma})$$
$$\propto \sum_{s=0}^{t} \int_{y \in \mathbb{R}^d} \mathbf{P}_{s,y,j}\left[t < \sigma(s),\, t < \tau,\, X^{(t)} \in dx \right] \tag{28}$$
$$\times \bar{\bar{\Pi}}_j(s, dy; \tilde{G}, \tilde{\gamma}),$$

where

$$\sigma(s) = \min\{r > s : J(r) \neq J(s)\}.$$

Definition (28) is motivated by the same two considerations as (27): First, $\Pi_j(s, dy; G, \gamma) = \pi_j(s, dy)$; this follows directly from the definition of Π_j, using that $\bar{\bar{\Pi}}_j(s, dy; G, \gamma) = \bar{\bar{\pi}}_j(s, dy)$. Second, the distribution $\Pi_j(t, dx; \tilde{G}, \tilde{\gamma})$ is invariant for a process similar to the process $Y_j^{(r)}$ mentioned above, except with times and positions drawn from $\bar{\bar{\Pi}}_j(s, dy; \tilde{G}, \tilde{\gamma})$ instead of $\bar{\bar{\pi}}_j(s, dy)$ on leaving the state $J = j$; see Section III C. Thus, given $\tilde{G}$ and samples from $\tilde{\gamma}$, one can generate samples from $\Pi_j(t, dx; \tilde{G}, \tilde{\gamma})$.

At this point we are ready to define the functions $\mathcal{G}$ and Γ appearing in (26) above. For a substochastic matrix $\tilde{G}$ and a collection of probability distributions $\tilde{\gamma} = \{\tilde{\gamma}_{ij}\}$, define the substochastic matrix

$$\mathcal{G}_{ij}(\tilde{G}, \tilde{\gamma}) = \sum_{t=0}^{\infty} \int_{x \in \mathbb{R}^d} \mathbf{P}_{t,x,i}[t + 1 < \tau,\, J^{(t+1)} = j] \tag{29}$$
$$\times \Pi_i(t, dx; \tilde{G}, \tilde{\gamma})$$

and the collection of probability distributions

$$\Gamma_{ij}(s, dy; \tilde{G}, \tilde{\gamma})$$
$$\propto \int_{x \in \mathbb{R}^d} \mathbf{P}_{s-1,x,i}[s < \tau,\, X^{(s)} \in dy,\, J^{(s)} = j] \tag{30}$$
$$\times \Pi_i(s - 1, dx; \tilde{G}, \tilde{\gamma}).$$

Because $\Pi_j(G, \gamma) = \pi_j$, expressions (20) and (24) imply that $\mathcal{G}(G, \gamma) = G$ and $\Gamma_{ij}(G, \gamma) = \gamma_{ij}$, establishing our fixed-point relation (26).

Having fully specified the fixed-point problem, we can now consider iterative methods for its solution. One approach would be to fix some $\varepsilon \in (0, 1]$ and compute the deterministic fixed-point iteration

$$\tilde{G}(m + 1) = \tilde{G}(m) + \varepsilon\left(\mathcal{G}(\tilde{G}(m), \tilde{\gamma}(m)) - \tilde{G}(m)\right), \text{ and}$$

$$\tilde{\gamma}(m + 1) = \tilde{\gamma}(m) + \varepsilon\left(\Gamma(\tilde{G}(m), \tilde{\gamma}(m)) - \tilde{\gamma}(m)\right), \tag{31}$$

given initial guesses $\tilde{G}(0)$ and $\tilde{\gamma}(0)$ for G and γ, respectively. One would choose $\varepsilon = 1$ in this deterministic iteration; we consider arbitrary $\varepsilon \in (0, 1]$ to motivate the stochastic approximation algorithm developed in Section III C.)

In practice, computing (31) requires computing averages with respect to $\Pi_j(\tilde{G}(m), \tilde{\gamma}(m))$ exactly and we can only compute a process sampling $\Pi_j(\tilde{G}(m), \tilde{\gamma}(m))$. Therefore, in Section III C, we explain how to solve the fixed-point problem by a stochastic approximation algorithm similar to (31) but based on random approximations of $\mathcal{G}$ and Γ.

C. A stochastic approximation

In this section, we present the full NEUS algorithm and we interpret it as a stochastic approximation algorithm analogous to the deterministic fixed-point iteration (31). In NEUS, as in the fixed-point iteration, we generate a sequence of approximations $\tilde{G}(m)$ and $\tilde{\gamma}(m)$, converging to G and γ, respectively. During the mth iteration of the NEUS algorithm, we update the current approximations $\tilde{G}(m)$ and $\tilde{\gamma}(m)$ based on statistics gathered from trajectories of a Markov process $Y_j^{(r)}(\tilde{G}(m), \tilde{\gamma}(m)) = \left(T_j^{(r)}, X_j^{(r)}\right)$ whose law depends on $\tilde{G}(m)$ and $\tilde{\gamma}(m)$. Recall that τ is the first time at which the process $(t, X^{(t)})$ is not contained in the set $D \in \mathbb{N} \times \mathbb{R}^d$. We define a single step in the evolution of $Y_j^{(r)}(\tilde{G}(m), \tilde{\gamma}(m))$ as follows:

1. Evolve the process $\left(X^{(t)}, J^{(t)}\right)$ one step from time $T_j^{(r)}$ and initial position $(X_j^{(r)}, j)$ to generate $\left(\tilde{X}_j^{(r+1)}, \tilde{J}_j^{(r+1)}\right)$ and time $\tilde{T}_j^{(r+1)} = T_j^{(r)} + 1$. Set $\tilde{Y}_j^{(r+1)} = (\tilde{T}_j^{(r+1)}, \tilde{X}_j^{(r+1)})$.

2. If $\tilde{J}_j^{(r+1)} = j$ and $\tilde{Y}_j^{(r+1)} \in D$, set $Y_j^{(r+1)}(\tilde{G}(m), \tilde{\gamma}(m)) = \tilde{Y}_j^{(r+1)}$.

3. Otherwise, if $\tilde{J}_j^{(r+1)} \neq j$ or $\tilde{Y}_j^{(r+1)} \notin D$, draw $Y_j^{(r+1)}(\tilde{G}(m), \tilde{\gamma}(m))$ from $\bar{\bar{\Pi}}_j(s, dy; \tilde{G}(m), \tilde{\gamma}(m))$.

The process $Y_j^{(r)}(\tilde{G}, \tilde{\gamma})$ preserves the distribution $\Pi_j(t, dx; \tilde{G}, \tilde{\gamma})$ in (28); see Appendix B for proof. For the following, we assume that $Y_j^{(r)}(\tilde{G}, \tilde{\gamma})$ is ergodic with respect to $\Pi_j(t, dx; \tilde{G}, \tilde{\gamma})$. The implementation details for drawing $Y_j^{(r+1)}(\tilde{G}(m), \tilde{\gamma}(m))$ from $\bar{\Pi}_j(s, dy; \tilde{G}(m), \tilde{\gamma}(m))$ will be discussed in more detail in Section V.

We now state the NEUS algorithm. To simplify the expressions below, we sometimes omit the iteration number m. The algorithm proceeds as follows:

1. Choose initial approximations $\tilde{G}(0)$ and $\tilde{\gamma}(0)$ of G and γ, respectively. Fix the number of steps, K, of the processes $Y_j^{(r)}(\tilde{G}(m), \tilde{\gamma}(m))$, and the maximum number of new points, L, included in the update to the empirical approximations of the distributions $\tilde{\gamma}_{ij}(m)$.

2. If $m > 0$, for each $j = 1, 2, \ldots, n$ evolve the process $Y_j^{(r)}(\tilde{G}(m), \tilde{\gamma}(m))$ for K time steps, starting from initial condition

$$Y_j^{(0)}(m) = Y_j^{(K)}(m-1).$$

Label the output $Y_j^{(r)}(m)$ for $r = 1, 2, \ldots, K$. Similarly, record the steps $\tilde{Y}_j^{(r)}$ and $\tilde{J}_j^{(r)}$ proposed during the first stage of the algorithm for simulating $Y_j^{(r)}(\tilde{G}(m), \tilde{\gamma}(m))$, and label them $\tilde{Y}_j^{(r)}(m)$ and $\tilde{J}_j^{(r)}(m)$, respectively. For $m = 0$, $Y_j^{(0)}(0)$ is drawn from $\bar{\Pi}_j(s, dy; \tilde{G}(0), \tilde{\gamma}(0))$.

3. Let

$$M_{ij}(m) = \sum_{r=1}^{K} \mathbf{1}_{\{j\}}(\tilde{J}_i^{(r)}) \, \mathbf{1}_D(\tilde{Y}_i^{(r)})$$

be the number of i to j transitions of the index process observed while generating $Y_j^{(r)}(m)$ and let $\left\{\tilde{X}_{ij}^{(\ell)}\right\}_{\ell=1}^{M_{ij}(m)}$ and $\left\{\tilde{T}_{ij}^{(\ell)}\right\}_{\ell=1}^{M_{ij}(m)}$ be the positions $\tilde{X}_i^{(r)}$ and times $\tilde{T}_i^{(r)}$ for which $\tilde{J}_i^{(r)} = j$ and $\tilde{Y}_i^{(r)} \in D$.

4. Compute

$$\hat{G}_{ij}(m) = \frac{M_{ij}(m)}{K}, \tag{32}$$

$$\hat{\gamma}_{ij}(s, dy; m) = \begin{cases} \frac{1}{L \wedge M_{ij}(m)} \sum_{\ell=1}^{L \wedge M_{ij}(m)} \mathbf{1}_{\tilde{T}_{ij}^{(\ell)}}(s) \, \delta_{\tilde{X}_{ij}^{(\ell)}}(dy) & \text{if } M_{ij}(m) > 0, \\ 0 & \text{if } M_{ij}(m) = 0, \end{cases} \tag{33}$$

and

$$\langle \hat{f} \rangle_i(m) = \frac{1}{K} \sum_{r=0}^{K-1} f\left(Y_i^{(r)}(m)\right),$$

where $L \wedge M_{ij}(m) = \min\{L, M_{ij}\}$. In Equation (33), δ_x represents the Dirac delta function centered at position x.

5. Replace the deterministic iteration (31) by the approximation

$$\tilde{G}_{ij}(m+1) = \tilde{G}_{ij}(m) + \varepsilon_m \left(\hat{G}_{ij}(m) - \tilde{G}_{ij}(m)\right) \tag{34}$$

and

$$\begin{aligned} \tilde{\gamma}_{ij}(m+1) = \tilde{\gamma}_{ij}(m) \\ + \varepsilon_m \left(\hat{\gamma}_{ij}(m) - \tilde{\gamma}_{ij}(m)\right) \left(\frac{\mathbf{1}_{\{M_{ij}(m)>0\}}}{I_{ij}(m)}\right) \end{aligned} \tag{35}$$

where

$$I_{ij}(m) = \frac{1}{m+1} \sum_{\ell=0}^{m} \mathbf{1}_{\{M_{ij}(\ell)>0\}}$$

and $\varepsilon_m > 0$ satisfies

$$\sum_{m=1}^{\infty} \varepsilon_m = \infty \quad \text{and} \quad \sum_{m=1}^{\infty} \varepsilon_m^2 < \infty. \tag{36}$$

6. Update the expectations

$$\langle \tilde{f} \rangle_i(m+1) = \langle \tilde{f} \rangle_i(m) + \varepsilon_m \left(\langle \hat{f} \rangle_i(m) - \langle \tilde{f} \rangle_i(m)\right). \tag{37}$$

7. Once the desired level of convergence has been reached, compute

$$\mathbf{E}\left[\sum_{t=0}^{\tau-1} f(t, X^{(t)})\right] \approx \sum_{j=1}^{n} \tilde{z}_j(m) \, \langle \tilde{f} \rangle_i(m),$$

where the vector $\tilde{z}(m)$ solves $\tilde{z}^{\mathrm{T}}(m) = \tilde{z}^{\mathrm{T}}(m) \, \tilde{G}(m) + a^{\mathrm{T}}$.

We now interpret NEUS as a stochastic approximation algorithm analogous to the deterministic fixed-point iteration (31). First, we observe that $\hat{G}(m)$ approximates $\mathcal{G}(\tilde{G}(m), \tilde{\gamma}(m))$ in the following sense. Suppose we were to compute a sequence

$$\hat{G}(n), \hat{G}(n+1), \ldots, G(n+k)$$

as in NEUS, except holding the values of $\tilde{G}(n)$ and $\tilde{\gamma}(n)$ fixed and computing $G(n+i)$ from a trajectory of $Y_i^{(t)}(\tilde{G}(n), \tilde{\gamma}(n))$. We would then have

$$\lim_{k \to \infty} \frac{1}{k} \sum_{i=0}^{k-1} G(n+i) = \mathcal{G}(\tilde{G}(n), \tilde{\gamma}(n)) \qquad (38)$$

with probability one, since the left hand side is a trajectory average of $Y_i^{(t)}(\tilde{G}(m), \tilde{\gamma}(m))$, and assuming that $Y_i^{(t)}(\tilde{G}(m), \tilde{\gamma}(m))$ is ergodic for $\Pi_i(\tilde{G}(m), \tilde{\gamma}(m))$, its limit is $\mathcal{G}(\tilde{G}(m), \tilde{\gamma}(m))$ by (29). The distribution $\gamma_{ij}(m)$ approximates $\Gamma_{ij}(\tilde{G}(m), \tilde{\gamma}(m))$ in a similar sense. Therefore, the NEUS iteration (34) is a version of the deterministic fixed-point iteration (31) but with a shrinking sequence ε_m instead of a fixed ε and with random approximations instead of the exact values of $\mathcal{G}$ and Γ. The conditions (36) on the sequence ε_m are common to most stochastic approximation algorithms [21]; they are necessary to ensure convergence of the iteration when $\mathcal{G}$ and Γ can only be approximated up to random errors.

We remark that in practice the empirical measures $\tilde{\gamma}(m)$ are stored as lists of time and position pairs. The update in (34) allows the number of pairs stored in these lists to grow with each iteration. This can lead to impractical memory requirements for the method. We therefore limit the size of each list $\tilde{\gamma}_{ij}(m)$ to a fixed maximum value by implementing a selection step in which the points that have been stored for the most iterations are removed to make room for the points in the updates of $\tilde{\gamma}_{ij}(m)$ when this maximum is exceeded. Also, in our numerical experiments in Section V, we use $\varepsilon_m = 1/(m+1)$ in which case,

$$\tilde{G}_{ij}(m) = \frac{1}{m+1} \sum_{\ell=0}^{m} \hat{G}_{ij}(\ell)$$

and

$$\tilde{\gamma}_{ij}(m) = \frac{1}{\sum_{\ell=0}^{m} \mathbf{1}_{\{M_{ij}(\ell)>0\}}} \sum_{\ell=0}^{m} \hat{\gamma}_{ij}(\ell).$$

This and other details of our implementation are explained in Section V.

IV. ERGODIC AVERAGES

In this section we consider the calculation of ergodic averages with respect to a general (not necessarily time-homogenous) Markov process. The system might not be stationary, and an "ergodic average" should be understood here as an infinite time average. We will assume that these time averages converge, which, for our purposes, is operationally the same as assuming that the system is stationary and ergodic. We also describe the simplifications that occur when the target Markov process is time-homogenous as in the original NEUS algorithm. We illustrate how quantities such as reaction rates can be computed within that context.

For a general Markov process we require that

$$\lim_{\tau \to \infty} \frac{1}{\tau} \sum_{t=0}^{\tau-1} \mathbf{P}\left[X^{(t)} \in dx, \ J^{(t)} = i\right]$$

exists as a probability distribution on $\mathbb{R}^d \times \{1, 2, \ldots, n\}$ and let

$$\pi(dx) = \lim_{\tau \to \infty} \frac{1}{\tau} \sum_{t=0}^{\tau-1} \mathbf{P}\left[X^{(t)} \in dx\right].$$

This general ergodicity requirement allows processes $X^{(t)}$ with periodicities or time dependent forcing.

Our goal is to compute ergodic averages of the form

$$\lim_{\tau \to \infty} \frac{1}{\tau} \sum_{t=0}^{\tau-1} \mathbf{E}\left[f(X^{(t)})\right] = \int_{x \in \mathbb{R}^d} f(x)\pi(dx).$$

To that end, in the notation of Section II B we fix a deterministic time horizon $\tau > 0$. If we divide both sides of (16) by τ and take the limit $\tau \to \infty$, we obtain the equation

$$z^{\mathrm{T}} G = z^{\mathrm{T}} \qquad (39)$$

where now

$$z_j = \lim_{\tau \to \infty} \frac{1}{\tau} \sum_{t=0}^{\tau-1} \mathbf{P}\left[J^{(t)} = j\right]. \qquad (40)$$

and

$$G_{ij} = \lim_{\tau \to \infty} \frac{\sum_{t=0}^{\tau-2} \mathbf{P}\left[J^{(t+1)} = j, \ J^{(t)} = i\right]}{\sum_{t=0}^{\tau-1} \mathbf{P}\left[J^{(t)} = i\right]}. \qquad (41)$$

Note that the matrix G is now stochastic and that $\sum_{j=1}^{n} z_j = 1$. We can rewrite the ergodic average of f as

$$\int_{x \in \mathbb{R}^d} f(x)\pi(dx) = \sum_{j=1}^{n} z_j \langle f \rangle_j,$$

where

$$\langle f \rangle_j = \int_{x \in \mathbb{R}^d} f(x)\pi_j(dx) \qquad (42)$$

and we represent the large τ limit of the position marginal distribution of π_j defined in (11) as

$$\pi_j(dx) = \lim_{\tau \to \infty} \sum_{t=0}^{\tau-1} \pi_j(t, dx). \qquad (43)$$

These formulas indicate that the only modification of the algorithm in Section III that is required to compute a long-time average is to set $\tau = \infty$ in the definition of the processes $Y_i(\tilde{G}, \tilde{\gamma})$, to set $a = 0$ in (27), and let $\tilde{z}$ solve $\tilde{z}^{\mathrm{T}} = \tilde{z}^{\mathrm{T}}\tilde{G}$ with $\sum_{j=1}^{n} \tilde{z}_j = 1$. In other words, the

algorithm seamlessly transitions from solving the initial value problem to solving the infinite time problem as τ becomes large.

When the joint process $(X^{(t)}, J^{(t)})$ is time-homogenous and stationary and our goal is to compute the average of a position dependent observable $f(x)$ with respect to the stationary distribution π of $X^{(t)}$, the above relations can be further simplified. In this case,

$$\pi_j(dx) = \frac{1}{z_j} \lim_{t \to \infty} \mathbf{P} \left[X^{(t)} \in dx, \, J^{(t)} = j \right],$$

where z_j defined in (40) becomes

$$z_j = \lim_{t \to \infty} \mathbf{P} \left[J^{(t)} = j \right].$$

The matrix G in (41) can now be written

$$G_{ij} = \lim_{t \to \infty} \mathbf{P} \left[J^{(t+1)} = j \mid J^{(t)} = i \right]$$

and the vector $\langle f \rangle_j$ defined in (42) becomes

$$\langle f \rangle_j = \int_{x \in \mathbb{R}^d} f(x) \pi_j(dx).$$

These simplifications lead to a version of the original NEUS method [7] that employs a direct method for solving for the weights similar to the scheme in [9].

In [9] and [8] the basic NEUS approach was extended to the estimation of transition rates between sets for a stationary Markov process. Implicit in this extension was the observation that any algorithm that can efficiently compute averages with respect to the stationary distribution of a time-homogenous Markov process can be applied to computing dynamic averages more generally by an enlargement of the state space, i.e., by applying the scheme to computing stationary averages for a higher dimensional time-homogenous Markov process. This idea is also central to Exact Milestoning [24], which extends the original Milestoning procedure [22] to compute steady-state averages with respect to a time-homogenous Markov process and is very similar in structure to steady-state versions of NEUS. For an alternative implementation of NEUS inspired by Exact Milestoning but presented in the more general context of Sections II B and III, see Appendix C.

V. NUMERICAL EXAMPLES

Here we illustrate the flexibility of the generalized algorithm with respect to both the means of restricting the trajectories (the choice of the $J^{(t)}$ process) and the averages that can be calculated. Specifically, in Section V A we discuss our choice of the $J^{(t)}$ process. In Section V B

we show how finite-time hitting probabilities can be calculated by discretizing the state space according to both time and space. In Section V C we show how free energies can be obtained by discretizing the state space according to time and the irreversible work.

A. One choice of the $J^{(t)}$ process

Rapid convergence of the scheme outlined in Section III rests on the choice of $J^{(t)}$. Perhaps the most intuitive choice is

$$J^{(t)} = \sum_{j=1}^n j \mathbf{1}_{A_j}(t, X^{(t)}) \tag{44}$$

where the subsets $A_1, A_2, \ldots, A_n$ partition $\mathbb{N} \times \mathbb{R}^d$. Indeed, earlier steady-state NEUS implementations [7–9, 25, 26] employed an analogous rule using a partition of the space variable (the time variable was not stored or partitioned). However, even with an optimal choice of the subsets $A_1, A_2, \ldots, A_n$, (44) has an important disadvantage: in many situations, $X^{(t)}$ frequently recrosses the boundary between neighboring subsets A_i and A_j, which slows convergence. Fortunately, there are many alternative choices of $J^{(t)}$ that approximate the choice in (44) while mitigating this issue. We give one simple and intuitive alternative which we use in the numerical examples that follow.

Let ψ_j be a set of non-negative functions on $\mathbb{N} \times \mathbb{R}^d$ for which $\sum_{j=1}^n \psi_j = 1$. The ψ_j are generalizations of the functions $\mathbf{1}_{A_j}$ in that they serve to restrict trajectories to regions of state space. In practice, given a partition of space $A_1, A_2, \ldots, A_n$, the ψ_j can be chosen to be smoothed approximations of the functions $\mathbf{1}_{A_j}$. Given a trajectory of $X^{(t)}$, the rule defining $J^{(t)}$ is as follows. Initially, choose $J^{(0)} \in \{1, 2, \ldots, n\}$ with probabilities proportional to $\{\psi_1(0, X^{(0)}), \psi_2(0, X^{(0)}), \ldots, \psi_n(0, X^{(0)})\}$. At later times $J^{(t)}$ evolves according to the rule

1. If $\psi_{J^{(t-1)}}(t, X^{(t)}) > 0$ then $J^{(t)} = J^{(t-1)}$.

2. Otherwise sample $J^{(t)}$ independently from $\{1, 2, \ldots, n\}$ according to probabilities $\{\psi_1(t, X^{(t)}), \psi_2(t, X^{(t)}), \ldots, \psi_n(t, X^{(t)})\}$.

While transitions out of $J^{(t)} = i$ occur when $X^{(t)}$ leaves the support of ψ_i, transitions back into $J^{(t)} = i$ can only occur outside of the support of ψ_j. Thus, this transition rule allows one to separate in space the values of $X^{(t)}$ at which $J^{(t)}$ transitions away from i from those where $J^{(t)}$ transitions into i, mitigating the recrossing issues mentioned above.

In our examples, we discretize time and only one additional "collective variable" (a dihedral angle in Section V B and the nonequilibrium work in Section V C). Here we denote the collective variable by ϕ, and we discretize it within some interval of values $[a, b]$ (though it may take values outside this interval). In both examples $[a, b]$ is evenly discretized into a set of points $\{a + k(b-a)/m_\phi\}_{k=0}^{m_\phi}$ for some integer m_ϕ. Letting ϕ_j be any of the points in that discretization, we set

$$\psi_j(t,x) \propto \begin{cases} \left[1 - \frac{1}{\Delta_\phi}|\phi(x)\phi_j|\right] \mathbf{1}_{[a,b]} & \text{if } |\phi(x) - \phi_j| \le \Delta_\phi \text{ and } t \in [t^j_{start}, t^j_{end}) \\ 0 & \text{otherwise} \end{cases} \quad (45)$$

where Δ_ϕ is some fixed value controlling the width of the support of ψ_j, and the indicator $\mathbf{1}_{[a,b]}$ restricts the terminal functions. Recall that the ψ_j are required to sum to 1. We choose t^j_{start} and t^j_{end} to equally divide the interval $[0, \tau)$, where, in our examples, τ is a fixed time horizon. The function ψ_j is largest when $t \in [t^j_{start}, t^j_{end})$ and $\phi(x) = \phi_j$. The supports of the various ψ_j correspond to products of overlapping intervals in the ϕ variable, but non-overlapping intervals in time. The fact that ψ_j depends on time is essential in our examples.

B. Finite-time hitting probability

In this section we compute the probability, $P_{BA}(\tau_{\max})$, of hitting a set B before a separate set A and before a fixed time $\tau_{\max} > 0$ given that the system is at a point $X^{(0)} \notin A \cup B$ at time $t = 0$. In the case where $X^{(0)}$ and B are separated by a large free energy barrier while $X^{(0)}$ and A are not, computing $P_{BA}(\tau_{\max})$ can be challenging since trajectories that contribute to $P_{BA}(\tau_{\max})$ are rare in direct simulations. To compute $P_{BA}(\tau_{\max})$ via the scheme in Section III C, we let the stopping time τ be the minimum of $\tau_{\max}$ and the first time, t, at which $X^{(t-1)}$ is in either A or B, i.e., $\tau - 1 = \min\{\tau_A, \tau_B, \tau_{\max} - 1\}$ where τ_A and τ_B are the first times that $X^{(t)}$ enters the sets A and B respectively. Strictly speaking, to write τ in the form in (9), we need to replace $(t, X^{(t)})$ in that equation by $(t, X^{(t-1)}, X^{(t)})$. The set D corresponding to our choice of τ is then $D = \{(t, x, y) : t < \tau_{\max}, x \notin (A \cup B)\}$. As we have already mentioned, this can be done without further modification of the scheme. Then $f(t, X^{(t)})$ in (10) is

$$f(t, X^{(t)}) = \mathbf{1}_B(X^{(t)}). \quad (46)$$

The system that we simulate is the alanine dipeptide $(CH_3\text{-}CONH\text{-}C^\alpha H(C^\beta H_3)\text{-}CONH\text{-}CH_3)$ in vacuum modeled by the CHARMM 22 force field [27]. We use the default Langevin integrator [28] implemented in LAMMPS [29], with a temperature of 310 K, a timestep of 1 fs and a damping coefficient of $30\,\text{ps}^{-1}$. The SHAKE algorithm is used to constrain all bonds to hydrogens [30]. We consider the system to be in set A if $-150° < \phi < -100°$ and in set B if $30° < \phi < 100°$ (Figure 2). We discretize time into intervals of $t_{end} - t_{start} = 10^3$ time steps with a terminal time of $\tau_{\max} = 10^4$ time steps. We use the rule outlined in Section V A for the evolution of $J^{(t)}$ with the ψ_j of the form in (V A). The ϕ_j in (V A) are chosen from the set $\{-100°, -74°, -48°, -22°, 4°, 30°\}$ with $[a, b] = [-100°, 30°]$ and $\Delta_\phi = 20°$.

We generate the initial point $X^{(0)}$ by running an unbiased simulation at 310 K and choosing a single point $X^{(0)}$ between the sets A and B. The vector a defined in (18) is

$$a_j = \frac{\psi_j(0, X^{(0)})}{\sum_{i=1}^n \psi_i(0, X^{(0)})}. \quad (47)$$

Note that the initial condition at $J^{(0)}$ can be drawn from an ensemble of configurations with minimal changes to the algorithm, but we restrict our attention to the initial condition consisting of a single point. To evaluate the performance of the algorithm in Section III C, we choose two points from our direct simulation, one at $\phi = -58.0°$ and one at $\phi = -91.0°$. The former is chosen to allow the NEUS results to be compared with results from unbiased direct simulations, while the latter provides a more challenging test because P_{BA} becomes small when $X^{(0)}$ is close to A.

We set $K = 1000$ and $L = 1$ and perform a total of 10^4 iterations (about 5.9 μs of dynamics) of the scheme in Section III C for each starting point. Each step of the process $Y_j^{(r)}(\tilde{G}(m), \tilde{\gamma}(m))$ corresponds to 10 time steps of the physical model. The $\tilde{\gamma}_{ij}$ are represented as lists of time and position pairs with associated weights. We cap the maximum size of those lists at 250 entries. If $\tilde{\gamma}_{ij}$ reaches this maximum size, each new entry overwrites the oldest previous entry. Because the lists are empty at the start of the calculation, we restrict sampling in Step 2 of the algorithm in Section III C to regions with at least one stored entry point ("progressive initialization" in [26]). When required, a sample (S, Y) is drawn from $\bar{\Pi}_j(s, dy; \tilde{G}(m), \tilde{\gamma}(m))$ by the following. With probability $a_j/z_j(1 - G_{jj})$, set $S = 0$ and select Y from $\mathbf{P}[X^{(0)} \in dy | J^{(0)} = j]$, or with the remaining probability select an index I proportional to the flux $z_i G_{ij}$ and then select (S, Y) from the list of weighted samples comprising $\tilde{\gamma}_{Ij}(m)$. For each j we compute $\langle f \rangle_j = P_{BA}^j = M_{jB}/(mK)$ where M_{jB} is the total number of transition events of $X_j^{(r)}$ into B observed after m iterations (mK is the total computational effort in state j after m iterations). The estimate of $P_{BA}(\tau_{\max})$ after m iterations is then computed as $P_{BA}(\tau_{\max}) = \sum_{j=1}^n P_{BA}^j \tilde{z}_j(m)$.

To assess the efficiency of the trajectory stratification, we also estimate $P_{BA}(\tau_{\max})$ by integrating an ensemble of $n = 10^6$ unbiased dynamics trajectories for $\tau_{\max}$ time steps from the initial point $X^{(0)}$. In this case, $P_{BA}(\tau_{\max}) \approx N_B/N$, where N_B is the number of trajectories that hit set B before set A. To assess the accuracy of the NEUS result, we perform 10 independent NEUS
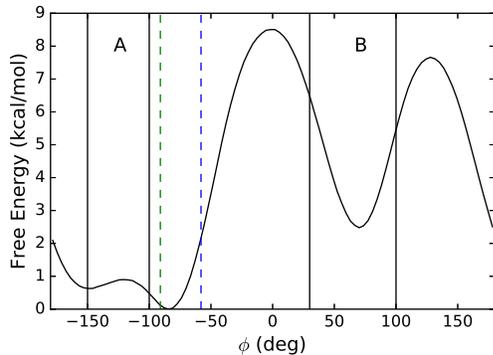
FIG. 2. Free energy (black curve) of the alanine dipeptide projected onto the ϕ dihedral angle, with sets A and B indicated. The initial positions of $X^{(0)}$ at $\phi = -58.0°$ (blue) and $\phi = -91.0°$ (green) are shown as vertical dashed lines. The free energy is computed from the method presented in Section II A as implemented in [19]

.

calculations. In each NEUS simulation, we estimate the value of P_{BA} as the average over the final 1000 iterations of each simulation and compute the mean of this estimate over 10 independent NEUS simulations. We obtain $P_{BA}(\tau_{\max}) \approx 4.33 \times 10^{-4}$ from NEUS and $P_{BA}(\tau_{\max}) \approx 4.12 \times 10^{-4}$ from direct simulation for the starting point at $\phi = -58.0°$ (Figure 3). In this case, the NEUS result is within the 95% confidence interval $[3.72 \times 10^{-4}, 4.52 \times 10^{-4}]$ (estimated as $\pm 1.96\sqrt{p(1-p)/n}$, where p is the estimate of P_{BA} from the direct simulation) for the direct simulation estimate given the number of samples. We obtain $P_{BA}(\tau_{\max}) \approx 2.94 \times 10^{-8}$ from NEUS for the starting point at $\phi = -91.0°$, consistent with the fact that none of the unbiased trajectories reached B before A in this case. From the same data (for either NEUS or direct simulation), one can easily assemble estimates of $P_{BA}(t)$ for any $t \leq \tau_{\max}$ by counting only those transitions into B that occur before t time steps. Up to a normalization, $P_{BA}(t)$ is the cumulative distribution function for the time that it takes $X^{(t)}$ to enter B conditioned on not entering A. Estimates of this cumulative distribution function compiled from the NEUS and direct simulation data are plotted in Figure 4. The NEUS results show excellent agreement with the results from the direct simulation.

Spatiotemporal plots of the weights computed from the converged NEUS calculations and the direct simulations are shown in Figure 5. For both starting points, the stratification scheme is able to efficiently sample events with weights spanning 12 orders of magnitude. When $X^{(0)}$ is close to the boundary of set A, accurate estimation of the very small probability $P_{BA}(\tau_{\max})$ depends sensitively on the ability to realize a set of very rare trajectories, ruling out the use of direct simulation.
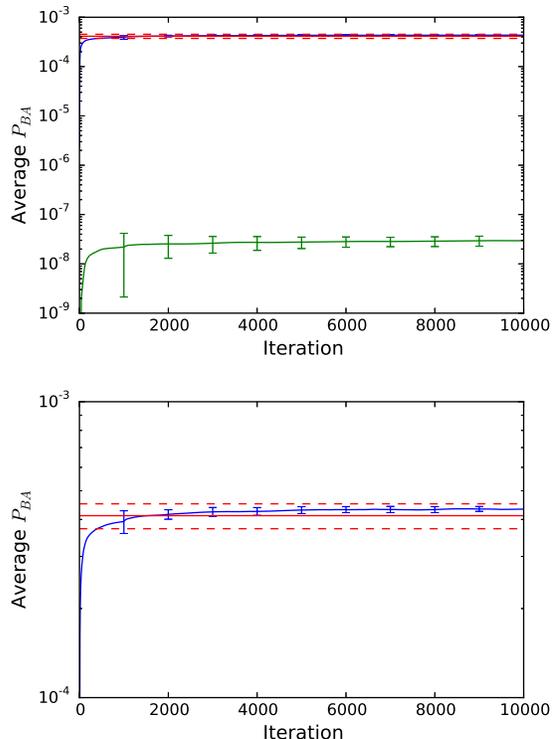

FIG. 3. Running estimate of P_{BA} from NEUS for dynamics starting at $\phi = -58.0°$ (blue, upper curve; error bars are computed every 1000 iterations and indicate $\pm 2.262 s/\sqrt{n}$ where s is the standard error estimated from $n = 10$ independent NEUS simulations) compared to the final result from direct simulation (red solid line; dashed lines indicate $\pm 1.96\sqrt{p(1-p)/n}$, where $n = 10^6$ is the number of physically weighted trajectories generated and p is the estimate of P_{BA} from the direct simulation). Also shown is the estimate from NEUS for dynamics starting from $\phi = -91.0°$ (green, lower curve; error bars computed similarly as the blue curve). The estimate at each iteration is computed as the average of the previous 1000 iterations. Lower panel is a magnification of the upper panel.

C. Free Energy Differences via the Jarzynski Equation

In this section, we show how a specific choice of the $J^{(t)}$ process enables us to stratify a path-dependent variable, specifically, the accumulated work appearing in the Jarzynski equation [6, 31]. For a statistical model defined by a density proportional to $\exp[-V(x)]$ (e.g., $V(x)$ is a potential function or a log-likelihood), the normalization constant is $Q = \int e^{-V(x)} dx$. In fields ranging from statistics to chemistry, a ratio of normalization constants is often used to compare models [32, 33]. Subject to certain conditions [31, 34], the Jarzynski equation relates the ratio of normalization constants to an average over paths of a time-dependent process, $X^{(t)}$:

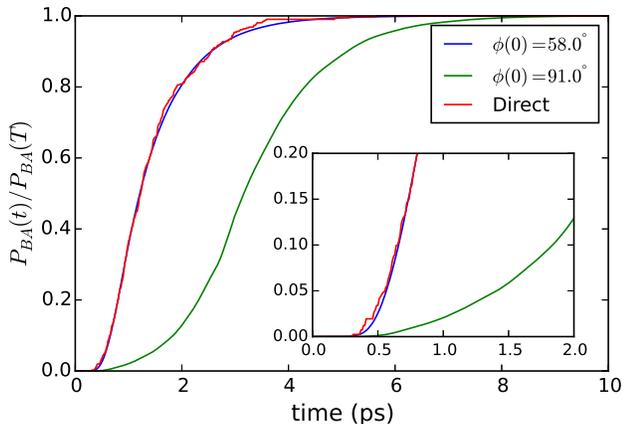$$\frac{Q_t}{Q_0} = \mathbf{E}\left[\exp(-W^{(t)})\right] \qquad (48)$$

FIG. 4. Estimate of the cumulative distribution function of the time to enter set B conditioned on not entering A from NEUS for the dynamics starting at $\phi = -58.0°$ (blue) and $\phi = -91.0°$ (green) compared to the result from the direct simulation (red). (Inset) The early time portion is shown. The estimate from each NEUS simulation at each time is computed as an average over the last 1000 iterations of the calculation and then averaged over 10 independent NEUS simulaitons.

where

$$W^{(t)} = \sum_{\ell=0}^{t-1} V(\ell+1, X^{(\ell)}) - V(\ell, X^{(\ell)}), \qquad W^{(0)} = 0$$

and we refer to $\Delta F = -\log(Q_t/Q_0)$ as the free energy difference. For example, for a small time discretization parameter, dt, a suitable choice of dynamics is

$$X^{(t+1)} = X^{(t)} - \frac{\partial V(t+1, X^{(t)})}{\partial x} dt + \sqrt{2\, dt}\, \xi_t \quad (49)$$

where ξ_t is a standard Gaussian random variable and $X^{(0)}$ is drawn from $p_0 \propto \exp[-V(0, x)]$.

Formula (48) suggests a numerical procedure for estimating free energy differences in which one simulates many trajectories of $X^{(t)}$, evaluates the work $W^{(t)}$ for each, and then uses this sample to compute the expectation on the right hand side of (48) approximately. This approach has been particularly useful in the context of single-molecule laboratory experiments [35, 36]. A well-known weakness of this strategy in the fast-switching (small t) regime is large statistical errors result from the fact that low-work trajectories contribute significantly to the expectation but are infrequently sampled [35, 37–40].

The quantity that we seek to compute is the free energy difference between a particle in a double-well potential that is additionally harmonically restrained with spring constant $k = 20$ near $x = -1$ and a particle in the same potential restrained near $x = 1$. The model is adapted from the one presented in [32]. Setting $\tau = 1001$, for $t < \tau$ we define

$$V(t, x) = 5\left(x^2 - 1\right)^2 + 3x + k\left(x - (2t\, dt - 1)\right)^2 \quad (50)$$

where $dt = 0.001$. We show $V(0, x)$, $V(\tau - 1, x)$, and $V(x; k = 0)$ in Figure 6. The process $X^{(t)}$ evolves according to (49).

The reader may be concerned that the expectation in (48) is not immediately of the general form in (10) suitable for an application of NEUS. We will apply NEUS as described in Section II B to the augmented process $Z^{(t)} = (X^{(t)}, W^{(t)})$. To compute the expectation of the left hand side of (48) via NEUS, we compute the expectation in (10) with

$$f(t, Z^{(t)}) = \begin{cases} \exp(-W^{(t)}) & \text{if } t = \tau - 1 \\ 0 & \text{if } t \neq \tau - 1. \end{cases} \quad (51)$$

The index process $J^{(t)}$ will mark transitions between regions of the time t and accumulated work $W^{(t)}$ variables. We discretize the work space in overlapping subsets using the pyramid form in (V A). We use 86 subsets with centers evenly spaced on the interval $[-25.0, 30.0]$ with a width of $\Delta_\phi = 0.6$. We discretize time into 5 discrete nonoverlapping subsets every 200 time steps for a total of 430 subsets. We again cap the maximum size of the list representation of $\{\tilde{\gamma}_{ij}\}$ at 250 entries using the same scheme as in Section V B.

For both NEUS and direct simulations, we prepare an ensemble of 1000 starting states $X^{(0)}$ by performing an unbiased simulation with fixed potential $V(0, x)$ for 10^6 steps, saving every 1000 steps. The direct fast-switching simulations start from each of these points and comprise 1000 steps of integration forward in time; each trajectory contributes equally to the left hand side of (48). For the NEUS simulations, the vector a is constructed as in (47), and trajectories are initialized at $J^{(0)}$ by drawing uniformly from this ensemble. We set $K = 1000$ and $L = 1$, and we perform 250 iterations. Each step in K corresponds to a single step of (49). As in Section V B, we sample only in the restricted distributions where there is at least one point stored in $\tilde{\gamma}$ from which to restart the dynamics.

The estimated ΔF produced from data generated in the last 50 iterations of NEUS is 5.87 (the units are chosen to absorb temperature factors above), which is in excellent agreement with the reference value of 5.94, in contrast to the estimate from direct simulation (Figure 7). The left panel of Figure 8 shows the weights along the time and work axes. In the right panel of Figure 8 we plot histogram approximations of the density $P_W(w)$ of $W^{(\tau-1)}$ along with the weighted density proportional to $P_W(w)\exp(-w)$. The separation of the peaks of this distribution highlight how NEUS is able to effectively sample the low work tails that contribute significantly to the expectation in the Jarzynski relation in (48) but are rarely accessed by the switching procedure in the unbiased simulations.
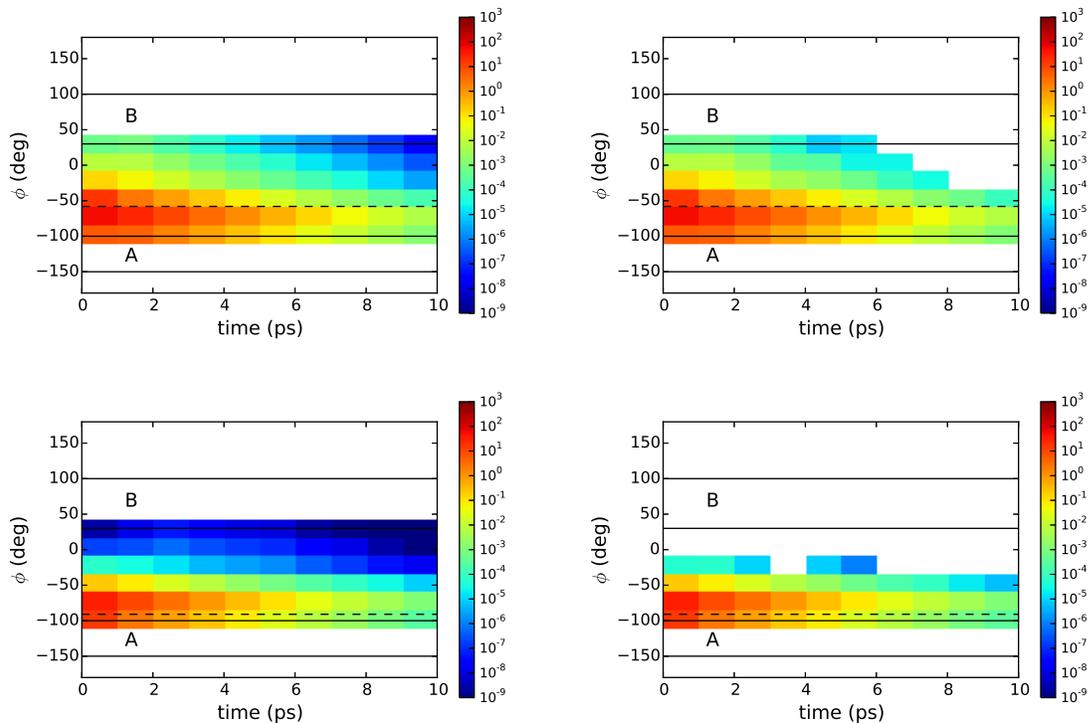
FIG. 5. Estimates of the subset weights from NEUS (left) and direct simulations (right). Upper panels show the dynamics starting from $\phi = -58.0°$ (dashed line) and lower panels show the dynamics starting from $\phi = -91.0°$ (dashed line). White space represents subsets which were not sampled.
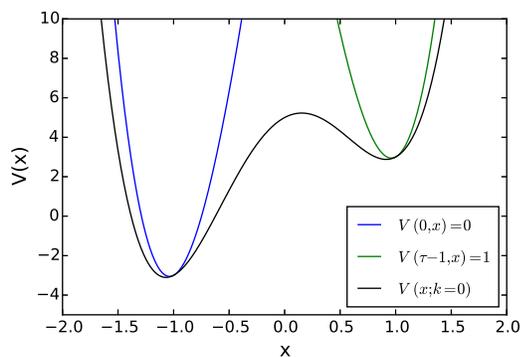

FIG. 6. $V(0, x)$ (blue) and $V(\tau-1, x)$ (green) for the switching process used to compute Jarzynski's equality. For reference, the potential with $k = 0$ (black) is also shown.
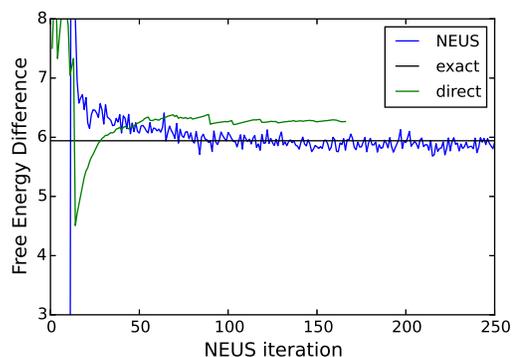

FIG. 7. Estimate of the free energy computed from NEUS (blue) and from conventional fast-switching simulations (green). The value computed from numerically integrating the potentials is shown as a black line. For the direct fast-switching simulations, we scale the number of repetitions to the number of NEUS iterations that are equivalent in computational effort.

VI. CONCLUSIONS

We describe a trajectory stratification framework for the estimation of expectations with respect to arbitrary Markov processes. The basis for this framework is the nonequilibrium umbrella sampling method (NEUS) originally introduced to compute steady state averages. Our development highlights the structural similarities between the nonequilibrium and equilibrium US algorithms

and places the NEUS method within the general context of stochastic approximation. These connections have practical implications for further optimizing the procedure and point the way to a more in depth convergence analysis that will be the subject of future work.
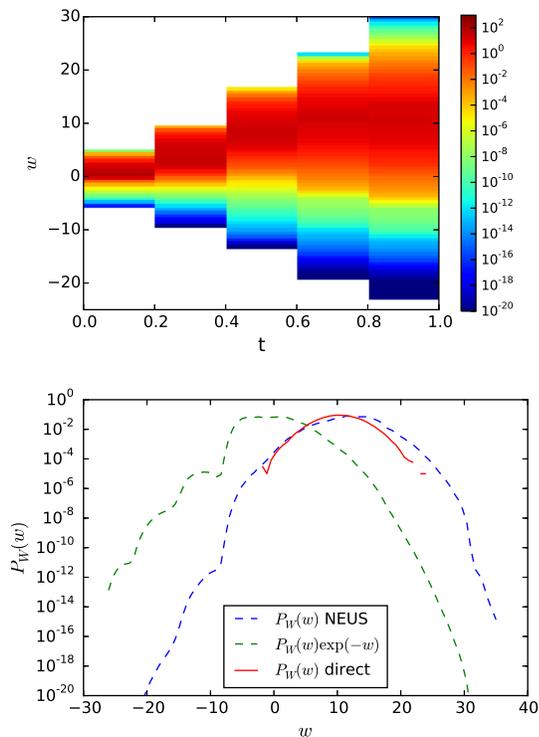
arriving observations (i.e., with respect to the posterior distribution). In high-dimensional settings (e.g., weather forecasting) the only practical alternatives are limited to providing information about only the mode of the posterior distribution (i.e., variational methods) or involve uncontrolled and often unjustified approximations (i.e., Kalman-type schemes). The approach that we present here opens the door to efficient data assimilation, machine learning, and, more generally, new forms of analysis of complex dynamics.

FIG. 8. Sampling the work with NEUS. (top) The estimate of the dynamic weights, z_j, from the final iteration of the NEUS calculation. White space represents subsets that are not visited in the NEUS calculation. (bottom) The probability density $P_W(w)$ of the accumulated work $W^{(\tau-1)}$ estimated from NEUS (blue dashed line), from direct integration (red solid line) and the exponentially scaled probability density proportional to $P_W(w)\exp(-w)$ estimated from the NEUS calculation (green dashed line).

Our development reveals that the basic trajectory stratification approach can be useful well beyond the estimation of stationary averages for time-homogenous Markov processes. This flexibility is demonstrated in two examples, both involving an expectation over trajectories of finite duration. In the first example, we show that the probability of first hitting a set within a finite time can be efficiently computed via stratification even when the dynamics start close to a competing absorbing state. In our second example, we use NEUS to stratify a process according to a path-dependent variable, the accumulated work in a nonequilibrium process appearing in the Jarzynski equation. The result is a novel and effective scheme for estimating free energy differences by enhancing sampling of the tails of the accumulated work distribution.

Our general framework also suggests new and exciting applications of trajectory stratification. For example, with little modification, these methods can be applied to sequential data assimilation applications where the goal is to approximate averages with respect to the conditional distribution of a hidden signal $X^{(t)}$ given sequentially

Appendix A: An alternative F

Here we present an alternative construction of the stochastic matrix F (Section II A) that more closely aligns with the nonequilibrium version of the algorithm presented in Section II B. Suppose that one has available a transition distribution $p(dy \,|\, x)$ for a Markov chain that preserves (or nearly preserves) the target density, π, in the sense that

$$\pi(dy) = \int_{x\in\mathbb{R}^d} p(dy \,|\, x)\pi(dx).$$

For example, $p(dy \,|\, x)$ might be the transition density for a number of steps of a Langevin dynamics integrator. We can again express the z_i as the solution to an eigenproblem (8) where now

$$F_{ij} = \sum_{i=1}^{n} \int_{y\in\mathbb{R}^d} \int_{x\in\mathbb{R}^d} \psi_j(y)p(dy \,|\, x)\pi_i(dx). \qquad \text{(A1)}$$

Note that when $\psi_i(x) = \mathbf{1}_{A_i}$ for some partition of space $\{A_i\}$, and $p(dy \,|\, x)$ is reversible with respect to π, the entry F_{ij} can be estimated by evolving samples according to $p(dy \,|\, x)$, rejecting any proposed samples that lie outside of A_i (so that π_i is preserved), and then counting the number of times the chain attempts transitions from set A_i to set A_j. For a closely related approach to approximating certain nonequilibrium quantities see [41].

Appendix B: Invariance of Π_i for $Y_i^{(t)}$

Lemma 1. *The measure $\Pi_i(t, dx; \tilde{G}, \tilde{\gamma})$ defined in (27) is invariant for the process $Y_i^{(t)}(\tilde{G}, \tilde{\gamma})$ defined in Section III C.*

Proof. To simplify notation, we write $\Pi_j(s, dy)$ for $\Pi_j(s, dy; \tilde{G}, \tilde{\gamma})$, and we do the same for $\bar{\Pi}_j$ and $Y_j^{(t)}$. The

kernel of the process $Y_j^{(t)}$ is

$$Q(t, x; s, dy) = \mathbf{P}_{t,x,j}^{\mathrm{acc}}(dy)\mathbf{1}_{\{t+1\}}(s) + \mathbf{P}_{t,x,j}^{\mathrm{rej}}\bar{\Pi}_j(s, dy),$$

where

$$\mathbf{P}_{t,x,j}^{\mathrm{acc}}(dy) = \mathbf{P}_{t,x,j}\left[t+1 < \tau,\, X^{(t+1)} \in dy,\, J^{(t+1)} = j\right],$$

and

$$\mathbf{P}_{t,x,j}^{\mathrm{rej}} = \mathbf{P}_{t,x,j}\left[t+1 = \tau \text{ or } J^{(t+1)} \neq j\right].$$

We will assume at first that $s > 0$. Applying the kernel Q to Π_j yields

$$\sum_{t=0}^{\infty}\int_x Q(t, x; s, dy)\Pi_j(t, dx)$$

$$\propto \sum_{t=0}^{\infty}\mathbf{1}_{\{t+1\}}(s)\sum_{r=0}^{t}\int_z\left(\int_x \mathbf{P}_{t,x,j}^{\mathrm{acc}}(dy)\mathbf{P}_{r,z,j}\left[t < \sigma(r),\, t < \tau,\, X^{(t)} \in dx\right]\right)\bar{\Pi}_j(r, dz)$$

$$+ \bar{\Pi}_j(s, dy)\sum_{t=0}^{\infty}\sum_{r=0}^{t}\int_z\left(\int_x \mathbf{P}_{t,x,j}^{\mathrm{rej}}\mathbf{P}_{r,z,j}\left[t < \sigma(r),\, t < \tau,\, X^{(t)} \in dx\right]\right)\bar{\Pi}_j(r, dz)$$

$$= \sum_{t=0}^{\infty}\mathbf{1}_{\{t+1\}}(s)\sum_{r=0}^{t}\int_z \mathbf{P}_{r,z,j}\left[t+1 < \sigma(r),\, t+1 < \tau,\, X^{(t+1)} \in dy\right]\bar{\Pi}_j(r, dz)$$

$$+ \bar{\Pi}_j(s, dy)\sum_{t=0}^{\infty}\sum_{r=0}^{t}\int_z \mathbf{P}_{r,z,j}\left[\sigma(r) = t+1 \text{ or } t+1 = \tau\right]\bar{\Pi}_j(r, dz)$$

$$= \sum_{r=0}^{s-1}\int_z \mathbf{P}_{r,z,j}\left[s < \sigma(r),\, s < \tau,\, X^{(s)} \in dy\right]\bar{\Pi}_j(r, dz)$$

$$+ \bar{\Pi}_j(s, dy)\sum_{r=0}^{\infty}\int_z\left(\sum_{t=r}^{\infty}\mathbf{P}_{r,z,j}\left[\sigma(r) = t+1 \text{ or } t+1 = \tau\right]\right)\bar{\Pi}_j(r, dz)$$

$$= \Pi_j(s, dy) - \bar{\Pi}_j(s, dy)$$

$$+ \bar{\Pi}_j(s, dy)\left(\sum_{r=0}^{\infty}\int_z \mathbf{P}_{r,z,j}\left[\sigma(r) < \infty \text{ or } \tau < \infty\right]\bar{\Pi}_j(r, dz)\right).$$

The expression in parentheses on the right side of last equality is equal to one since we assume that $\mathbf{E}[\tau] < \infty$, which implies $\tau < \infty$ with probability one. Thus, for $s > 0$,

$$\sum_{t=0}^{\infty}\int_x Q(t, x; s, dy)\Pi_j(t, dx) \propto \Pi_j(s, dy).$$

A very similar argument shows that the same expression holds when $s = 0$. Thus, Π_j is invariant for $Y_j^{(t)}$. $\quad\square$

Appendix C: An alternative implementation

The Exact Milestoning method recently proposed for computing certain dynamic averages with respect to a stationary time-homogenous Markov process [24] has a structure very similar to the steady-state version of NEUS that we have described in Section IV. The differences in implementation between the two methods suggest an alternative implementation of NEUS in the more general context of Sections II B and III.

To describe this alternative implementation, it is convenient to work with a slightly different set of variables that reflects the interface-to-interface emphasis in Mile-

stoning. Here we show how these new variables are equivalent to the variables that we used to describe NEUS. In particular, we define the substochastic matrix

$$H_{ij} = \frac{\sum_{\ell=0}^{\infty} \mathbf{P}\left[S^{\ell+1} < \tau,\, J^{(S^{\ell+1})} = j,\, J^{(S^{(\ell)})} = i\right]}{\sum_{\ell=0}^{\infty} \mathbf{P}\left[J^{(S^{(\ell)})} = i,\, S^{\ell} < \tau\right]}$$

$$\text{(C1)}$$

$$= \sum_{t=0}^{\infty} \int_{x \in \mathbb{R}^d} \mathbf{P}_{t,x,i}\left[\sigma(t) < \tau,\, J^{(\sigma(t))} = j\right] \bar{\pi}_i(t, dx)$$

$$\text{(C2)}$$

which can be expressed in terms of G by

$$H_{ij} = \begin{cases} G_{ij}/(1 - G_{ii}), & i \neq j \\ 0, & i = j. \end{cases}$$

We also define

$$v_j = \sum_{\ell=0}^{\infty} \mathbf{P}\left[S^{\ell} < \tau,\, J^{(S^{(\ell)})} = i\right] = z_j(1 - G_{jj}),$$

which satisfies

$$v^{\mathrm{T}} = v^{\mathrm{T}} H + a^{\mathrm{T}},$$

and we define the matrix of subprobability distributions

(they do not integrate to 1)

$$\xi_{ij}(s, dy)$$
$$= \sum_{t=0}^{\infty} \int_{x \in \mathbb{R}^d} \mathbf{P}_{t,x,i}\left[\sigma(t) = s,\, s < \tau,\, X^{(s)} \in dy,\, J^{(s)} = j\right]$$
$$\times \bar{\pi}_i(t, dx)$$

which are related to the conditional flux distributions γ_{ij} as defined in (24) by

$$\xi_{ij} = \begin{cases} H_{ij}\, \gamma_{ij}, & i \neq j \\ 0, & i = j. \end{cases}$$

Note that $\sum_j \gamma_{ij}$ is the distribution of transition points leaving $J = i$. We have already seen in (25) that

$$v_j\, \bar{\pi}_j(s, dy) = \begin{cases} \sum_i v_i\, \xi_{ij}(s, dy), & \text{if } s > 0, \\ a_j\, \mathbf{P}\left[X^{(0)} \in dy \mid J^{(0)} = j\right], & \text{if } s = 0. \end{cases}$$

We will construct an algorithm that finds the unnormalized distribution $\theta_j = v_j \bar{\pi}_j$.

In light of these observations, and following the development in Section III, for any vector of subprobability distributions $\{\tilde{\theta}_j\}$ approximating $\{\theta_j\}$, we define the functions

$$\Xi_{ij}(s, dy; \tilde{\theta}) = \frac{1}{\tilde{v}_i} \sum_{t=0}^{\infty} \int_{x \in \mathbb{R}^d} \mathbf{P}_{t,x,i}\left[\sigma(t) = s,\, s < \tau,\, X^{(s)} \in dy,\, J^{(s)} = j\right] \tilde{\theta}_i(t, dx),$$

and

$$\Theta_j(\tilde{\theta}) = \begin{cases} \sum_{i=1}^{n} \tilde{v}_i\, \Xi_{ij}(\tilde{\theta}), & \text{if } s > 0, \\ a_j\, \mathbf{P}\left[X^{(0)} \in dy \mid J^{(0)} = j\right], & \text{if } s = 0. \end{cases}$$

for $\tilde{v}_j = \sum_{t=0}^{\infty} \int_y \tilde{\theta}_j(s, dy)$. The fixed-point problem describing this alternative implementation finds a vector of time and position subprobability distributions $\{\theta_j\}$ satisfying

$$\Theta_j(\theta) = \theta.$$

Given the current iterate $\tilde{\theta}(m)$ with the vector of normalizations $\tilde{v}(m)$, the stochastic approximation algorithm corresponding to this fixed-point problem fixes a non-negative integer N and, for each i, initiates N independent trajectories of $(t, X^{(t)})$ from $\tilde{\theta}_i(m)/\tilde{v}_i(m)$, all of which are run until τ or the next transition of the index process. Suppose that N_i of these N trajectories end upon a transition of the index process (rather than upon reaching time τ). Let $\{(J_i^{\ell}(m), T_i^{\ell}(m), X_i^{\ell}(m))\}$, be the

corresponding values of the index process, time, and position upon those transition events. Then we can define the empirical subprobability measures

$$\hat{\xi}_{ij}(s, dy; m) = \frac{1}{N} \sum_{\ell=1}^{N_i} \mathbf{1}_{\{j\}}(J_\ell^i(m))\, \mathbf{1}_{\{s\}}(T_i^\ell(m))\, \delta_{X_\ell^i(m)}(dy)$$

and set

$$\hat{\theta}_j(s, dy; m) = \begin{cases} \sum_{i=1}^{n} \tilde{v}_i(m)\, \hat{\xi}_{ij}(m), & \text{if } s > 0, \\ a_j\, \mathbf{P}\left[X^{(0)} \in dy \mid J^{(0)} = j\right], & \text{if } s = 0. \end{cases}$$

Note that $\mathbf{E}\left[\hat{\theta}_j(m)\right] = \Theta_j(\tilde{\theta}(m))$.

The alternative stochastic approximation algorithm is then

$$\tilde{\theta}_j(m+1) = \tilde{\theta}_j(m) + \varepsilon_m\left(\hat{\theta}_j(m) - \tilde{\theta}_j(m)\right). \quad \text{(C3)}$$

An approximation of $\bar{\pi}_j$ is obtained at any time by normalizing $\tilde{\theta}_j(m)$.

Both the implementation described in this section and the one described in Section III store an empirical estimate of $\bar{\pi}$ (in Section III the estimate of γ directly corresponds through (27) to an estimate of $\bar{\pi}$). The core difference between the two implementations is that in this section one generates a fixed number of trajectory fragments for each value of the index process, while in the implementation in Section III the number of trajectory fragments generated is random but the total computational expense for each value of the index process is fixed. This alternative iteration has the advantage that the update in (C3) is completely uncorrelated between iterations (conditioned on the current iterate). On the other hand, a potential disadvantage is that the random and varied computational effort for each value of the index process could make synchronization difficult in a parallel implementation.

---

[1] D. Frenkel and B. Smit, *Understanding Molecular Simulation* (Academic Press, 2002).

[2] J. Neyman, Journal of the Royal Statistical Society **97**, 558 (1934).

[3] G. M. Torrie and J. P. Valleau, Journal of Computational Physics **23**, 187 (1977).

[4] C. Pangali, M. Rao, and B. J. Berne, J. Chem. Phys. **71**, 2975 (1979).

[5] D. Chandler, *Introduction to Modern Statistical Mechanics* (Oxford University Press, 1987).

[6] T. Leliévre, M. Rousset, and G. Stoltz, *Free Energy Computations: A Mathematical Perspective* (Imperial College Press, 2010).

[7] A. Warmflash, P. Bhimalapuram, and A. R. Dinner, J. Chem. Phys. **127**, 154112 (2007).

[8] A. Dickson, A. Warmflash, and A. R. Dinner, J. Chem. Phys. **131**, 154104 (2009).

[9] E. Vanden-Eijnden and M. Venturoli, J. Chem. Phys. **131**, 044120 (2009).

[10] P. Glasserman, P. Heidelberger, P. Shahabuddin, and T. Zajic, "A look at multilevel splitting," in *Monte Carlo and Quasi-Monte Carlo Methods 1996: Proceedings of a conference at the University of Salzburg, Austria, July 9–12, 1996*, edited by H. Niederreiter, P. Hellekalek, G. Larcher, and P. Zinterhof (Springer New York, New York, NY, 1998) pp. 98–108.

[11] G. A. Huber and S. Kim, Biophys. J. **70**, 97 (1996).

[12] Z. Haraszti and J. K. Townsend, ACM Trans. Model. Comput. Simul. **9**, 105 (1999).

[13] T. S. van Erp, D. Moroni, and P. G. Bolhuis, J. Chem. Phys. **118**, 7762 (2003).

[14] R. J. Allen, P. B. Warren, and P. R. ten Wolde, Phys. Rev. Lett. **94**, 018104 (2005).

[15] A. Johansen, P. Del Moral, and A. Doucet, in *Proceedings of the 6th International Workshop on Rare Event Simulation* (Bramberg, 2006).

[16] F. Cérou and A. Guyader, Stochastic Analysis and Applications **25**, 417 (2007).

[17] N. Guttenberg, A. R. Dinner, and J. Weare, J. Chem. Phys. **136**, 234103 (2012).

[18] M. Hairer and J. Weare, Commun. Pure Appl. Math. **67**, 1995 (2014).

[19] E. Thiede, B. Van Koten, J. Weare, and A. R. Dinner, J. Chem. Phys. **145**, 084115 (2016).

[20] A. R. Dinner, E. Thiede, B. Van Koten, and J. Weare, in prep..

[21] H. J. Kushner and G. G. Yin, *Stochastic Approximations and Recursive Algorithms and Applications*, second edition ed. (Springer, 2003).

[22] A. K. Faradjian and R. Elber, J. Chem. Phys. **120**, 10880 (2004).

[23] C. Schütte, F. Noé, J. Lu, M. Sarich, and E. Vanden-Eijnden, J. Chem. Phys. **134**, 204105 (2011).

[24] J. M. Bello-Rivas and R. Elber, J. Chem. Phys. **142**, 094102 (2015).

[25] A. Dickson, A. Warmflash, and A. R. Dinner, J. Chem. Phys. **130**, 074104 (2009).

[26] A. Dickson, M. Maienschein-Cline, A. Tovo-Dwyer, J. R. Hammond, and A. R. Dinner, J. Chem. Theory Comput. **7**, 2710 (2011).

[27] J. A. D. MacKerell, D. Bashford, M. Bellott, J. R. L. Dunbrack, J. D. Evanseck, M. J. Field, S. Fischer, J. Gao, H. Guo, S. Ha, D. Joseph-McCarthy, L. Kuchnir, K. Kuczera, F. T. K. Lau, C. Mattos, S. Michnick, T. Ngo, D. T. Nguyen, B. Prodhom, W. E. Reiher, B. Roux, M. Schlenkrich, J. C. Smith, R. Stote, J. Straub, M. Watanabe, J. Wiórkiewicz-Kuczera, D. Yin, and M. Karplus, J. Phys. Chem. B **102**, 3586 (1998).

[28] T. Schneider and E. Stoll, Phys. Rev. B **17**, 1302 (1978).

[29] S. Plimpton, J. Comp. Phys. **117**, 1 (1995).

[30] J. Ryckaert, G. Ciccotti, and J. C. Berendsen, J. Comp. Phys. **23**, 327 (1977).

[31] C. Jarzynski, Phys. Rev. Lett. **78**, 2690 (1997).

[32] C. Chipot and A. Pohorille, *Free Energy Simulations* (Springer, 2007).

[33] R. E. Kass and A. E. Raftery, Journal of the American Statistical Association **90**, 773 (1995).

[34] R. M. Neal, Stat. Comput. **11**, 125 (2001).

[35] G. Hummer, J. Chem. Phys. **114**, 7330 (2001).

[36] G. Hummer and A. Szabo, Biophys. J. **85**, 5 (2003).

[37] F. M. Ytreberg and D. M. Zuckerman, J. Chem. Phys. **120**, 10876 (2004).

[38] H. Oberhofer, C. Dellago, and P. L. Geissler, J. Phys. Chem. B **109**, 6902 (2005).

[39] C. Jarzynski, Phys. Rev. E **73**, 046105 (2006).

[40] S. Vaikuntanathan and C. Jarzynski, J. Chem. Phys. **134**, 054107 (2011).

[41] E. Vanden-Eijnden and M. Venturoli, The Journal of Chemical Physics **130**, 194101 (2009).