Nonequilibrium umbrella sampling in spaces of many order parameters

Alex Dickson, Aryeh Warmflash, and Aaron R. Dinner

Nonequilibrium umbrella sampling in spaces of many order parameters

Alex Dickson, Aryeh Warmflash, and Aaron R. Dinner[a)]
James Franck Institute, The University of Chicago, Chicago, Illinois 60637, USA

We recently introduced an umbrella sampling method for obtaining nonequilibrium steady-state probability distributions projected onto an arbitrary number of coordinates that characterize a system (order parameters) [A. Warmflash, P. Bhimalapuram, and A. R. Dinner, J. Chem. Phys. **127**, 154112 (2007)]. Here, we show how our algorithm can be combined with the image update procedure from the finite-temperature string method for reversible processes [E. Vanden-Eijnden and M. Venturoli, "Revisiting the finite temperature string method for calculation of reaction tubes and free energies," J. Chem. Phys. (in press)] to enable restricted sampling of a nonequilibrium steady state in the vicinity of a path in a many-dimensional space of order parameters. For the study of transitions between stable states, the adapted algorithm results in improved scaling with the number of order parameters and the ability to progressively refine the regions of enforced sampling. We demonstrate the algorithm by applying it to a two-dimensional model of driven Brownian motion and a coarse-grained (Ising) model for nucleation under shear. It is found that the choice of order parameters can significantly affect the convergence of the simulation; local magnetization variables other than those used previously for sampling transition paths in Ising systems are needed to ensure that the reactive flux is primarily contained within a tube in the space of order parameters. The relation of this method to other algorithms that sample the statistics of path ensembles is discussed. © *2009 American Institute of Physics*. [DOI: 10.1063/1.3070677]

I. INTRODUCTION

Many systems of interest in the physical and biological sciences take energy and matter from their surroundings by one means and return it by another; this exchange can drive them far from equilibrium. In the past decade or so, there have been dramatic advances in our ability to describe such systems quantitatively.[1] Because few models are analytically tractable, simulations are essential for interpreting experimental data and for providing results that can be used to validate theories. The central quantity of interest for such systems is the steady-state probability distribution as a function of a set of collective variables (order parameters). This quantity plays the role of a thermodynamic potential in that statistical averages that enable connection with experimental observables can be calculated from it.

Just as in the more familiar case of systems in equilibrium, systems in nonequilibrium steady states can be quite microscopically dynamic even when macroscopically static. A number of robust, well-established algorithms exist for simulating the dynamics of microscopically irreversible systems,[2–5] but relaxation to the steady state can often be prohibitively slow, especially when there are multiple (meta)stable states with bottlenecks between them in phase space. This problem can be overcome by methods that enhance sampling of low probability states and reaction pathways but still enable retrieval of the steady-state distribution associated with the original dynamics. Indeed, such methods are essential for obtaining information about dynamical bottlenecks (transition states), which are of central impor-

tance for elucidating mechanisms. However, almost all such algorithms that exist rely on detailed balance (microscopic reversibility) and *a priori* knowledge of the distribution function (typically, Boltzmann weighting).[2,3] By definition, these do not hold in systems in nonequilibrium steady states. New simulation paradigms are needed to treat this important class of systems.

The key to developing algorithms for enhanced sampling of nonequilibrium processes is that one must account carefully for the fluxes into regions. Perhaps the most well-known method that does so is forward flux sampling (FFS).[6–11] In FFS, the phase space between two attractors is divided by a series of nonintersecting interfaces. The system is then ratcheted from one attractor to the other as follows. A straightforward simulation is run in the first stable state, and crossings of the interface closest to that stable state are recorded. Then, the configurations sampled at the first interface are used to initialize new simulations. The trajectories that reach the second interface prior to returning to the basin around the attractor are counted, and configurations at the second interface are saved. The last step is repeated for successive interfaces until the second attractor is reached. FFS was introduced originally to calculate rates of nonequilibrium processes, but information from forward and backward FFS simulations can be combined to obtain steady-state probability distributions.[9]

FFS has enjoyed much success, but it was designed for studies of transitions between two strong attractors. This essentially precludes its use to study transitions with long-lived intermediates. Even when studying a system with only two strong attractors, a poor choice of interfaces can diminish the

---

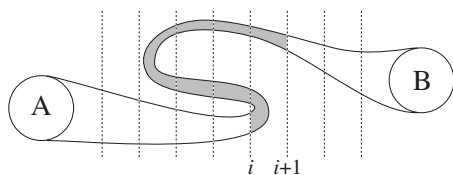a)Electronic mail: dinner@uchicago.edu.

FIG. 1. A schematic illustrating the result of poor interface placement in FFS. The tube containing most transition paths that connect A and B is indicated by solid lines, and the interfaces are indicated by dotted lines. One wants to divide the reaction path into short trajectory segments, but interfaces i and $i+1$ are separated by long trajectory segments (gray) for this choice of order parameter. Particularly if this portion of the tube contains the rate-limiting step of the reaction, computational gains over physically weighted simulations can be lost. This problem cannot be alleviated by simply spacing the interfaces closer together.

computational gains over physically weighted simulations. If the interfaces do not capture the important features of the reaction path, then the simulation time required to connect two interfaces can be prohibitively long no matter how closely one spaces the interfaces. One such situation is depicted schematically in Fig. 1. Even in simpler cases, the sampling of interface crossing points can be incomplete, and there is no means for correcting resulting errors that arise early in a FFS simulation.[12,13] Choosing the interfaces optimally (i.e., identifying the variables that best characterize the progress of the reaction) is tantamount to determining the reaction path itself.

We recently developed an alternate algorithm for enhanced sampling of a nonequilibrium steady-state probability distribution.[14] Our original motivation was that we wanted to sample the tails of unimodal steady-state probability distributions (e.g., for calculating correlation functions like those in Warmflash and Dinner[15]). Our algorithm can be viewed as a form of umbrella sampling,[2,3,16,17] in which the exploration of a coordinate is divided into a series of separate simulations (windows) with artificial weighting, and the results are then combined in such a way that the physical probability distribution (or free energy in the case of equilibrium systems) is recovered. Although it requires the existence of a well-defined steady state, our algorithm does not assume that there are two strong attractors and permits an arbitrary number of order parameters to be used to constrain sampling, in contrast to FFS. The iterative nature of the method makes it robust to errors that arise early in a simulation. Warmflash *et al.*[14] demonstrated the ability of the method to accelerate sampling of a distribution as a function of two order parameters for a genetic toggle switch; the method is also sufficiently sensitive to reveal the bifurcation of transition paths in the well-studied Maier–Stein model (see Ref. 18).[8,19–21]

One problem with umbrella sampling (whether for reversible or irreversible systems) is that, since the volume of the space of order parameters scales exponentially with its dimension, so does the total computational expenditure required for convergence of the probability distribution. Although in many systems a small number of collective variables that capture the main features of the dynamics of interest can be identified, their choice is often not obvious.[22] A poor choice can actually slow down the convergence of an umbrella sampling calculation because the remaining de-

grees of freedom can then be very slow in relaxing. One thus wants to constrain the dynamics with as many order parameters as possible to ensure good sampling in those that ultimately prove to be important but at the same time minimize the computational cost of the simulation. It has been shown for equilibrium systems that these competing demands can be met by restricting sampling to the vicinity of a path in a space of many order parameters. This is the idea of the finite-temperature string method.[23–30] The effective reduction of the space to one dimension (the arc length along the path) decouples the computational expenditure from the number of order parameters originally specified. The path is not known *a priori*; rather, it is dynamically evolved during the course of the simulation.

Here, we exploit the ability of our algorithm[14] to enforce even sampling in a space of an arbitrary number of order parameters to develop a stringlike method for nonequilibrium processes. We illustrate the method by applying it to a two-dimensional (2D) system consisting of a Brownian particle in a unidirectional flow on a periodic surface. We then examine nucleation in an Ising model under "shear;"[11] the configuration of the system is projected onto a set of coarse-grained coordinates that generalize readily to more complex systems. The transition mechanism is described and magnetization histograms are obtained for several different system sizes. The relation to other methods, advantages and disadvantages of the new algorithm, and the effect of the choice of order parameters are discussed.

II. METHODS

In this section, we review pertinent operational features of the original nonequilibrium umbrella sampling algorithm[14] and the finite-temperature string method.[29] Formalism and additional simulation details can be found in the original papers. We then combine aspects of the two methods in a new procedure for restricted sampling of nonequilibrium steady-state probability distributions.

A. Original algorithm

The essential idea of the nonequilibrium umbrella sampling algorithm is that a separate simulation can be performed in each region if the fluxes into its component states are known (Fig. 2). The space of order parameters is divided into regions that form a lattice. To ensure even sampling of the space, one copy of the system (a "replica" or "walker") is confined to each region but otherwise evolves according to the unperturbed dynamics. During the course of the simulation, two quantities are self-consistently determined: weight factors (W) that account for the unphysical distribution of walkers and flux probability distributions (Φ) that are used to reinitialize the walkers when they attempt to exit their region. To break the feedback between these two quantities, two overlapping lattices are used. Simulations in the first lattice determine the flux probability distributions of the second lattice and vice versa.

We employed two sets of weight factors: a continuously updated one in which rapid fluctuations were allowed and an
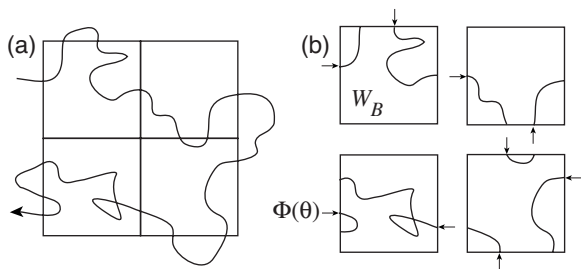
FIG. 2. Schematic motivating the original method. (a) An unconstrained trajectory that passes through four regions that cover a space of order parameters. (b) Four separate simulations in which the sampling in each region is the same but the paths are initiated from the incoming fluxes (indicated by arrows). During the umbrella sampling, there is little communication between the regions, and the trajectory segments shown can be sampled in any order. The algorithm self-consistently determines the fluxes and the relative probability of each region (the weight factor).

averaged one that was used in computing the fluxes. With every ΔT_{wt}, we replaced the latter with the average of the former over the previous time interval:

$$\bar{W}_k^{(y)} = \frac{1}{n} \sum_{i=1}^{n} W_k^{(y)}(i\Delta T_{\mathrm{wt}}/n + T_{\mathrm{int}}), \qquad (1)$$

where $\bar{W}_k^{(y)}$ is the weight of region k on lattice y and $W_k^{(y)}(t)$ is the instantaneous weight of that region at time t. ΔT_{wt} is the weight update interval, n is an arbitrary discretization constant set here to 100, and T_{int} denotes the time at the beginning of that sampling interval. The $\{W_k^{(y)}\}$ are evolved by transferring an incremental amount between weight variables each time a walker attempts to leave its region according to

$$-\Delta W_B^{(y)} = \Delta W_{B'}^{(y)} = s W_B^{(y)} T^*/T_B^{(y)}, \qquad (2)$$

where B labels the region to which the walker is confined, B' labels the region to which it attempted to go, Δ denotes an additive change, and s is a user-defined parameter chosen to optimize convergence. The factor T^*/T_B allows the simulations in different regions to be run asynchronously and accounts for the fact that longer simulations allow walkers more opportunities for escape; $T_B^{(y)}$ is the time elapsed in region B and T^* is an arbitrary standard that in practice is chosen to be the time elapsed in a particular region of the system. In addition to the parameters discussed above, the rate at which the weights converge to their steady-state values depends on the quality of the flux distributions.

The technique used to calculate the flux probability distributions in the original algorithm differs considerably from that in the algorithm presented here, but we briefly review the original technique for completeness. In the original algorithm, the regions are subdivided by hand, typically into bins of equal size. The flux distributions for each lattice are accumulated as histograms over these bins by counting the crossings of interfaces of one lattice by the walkers on the other lattice [Eq. (4) of Warmflash et al.[14]]. When a walker attempts to leave its region, its position in the order parameter space is reset to one of the bins chosen with probability proportional to its histogram value [Eq. (3) of Warmflash et al.[14]]. Once the bin is determined, the remaining degrees of

freedom are chosen from their joint distribution at that position. In practice the latter step is accomplished by storing complete phase space points at the region boundaries. In the event that a dynamics that is non-Markovian in the phase space is employed (e.g., a generalized Langevin equation), all systematic information needed to continue the trajectory (e.g., the momenta, systematic forces, and the integral of the memory kernel) is saved upon an attempted exit and restored when that trajectory is reinitialized in the neighboring region.

B. String method

In the finite-temperature string method,[23–29] the phase space is also divided into regions, but instead of a lattice with the dimensionality of the order parameter space, the regions are arranged along a one-dimensional string that winds its way through that space. The computational cost of the calculation scales linearly with the number of regions along the string (N_{im}) but is now independent of the dimensionality of the order parameter space.

There are several versions of the finite-temperature string method. Here, we describe that of Vanden-Eijnden and Venturoli,[29] which we follow most closely in our own implementation. The steps of the algorithm are as follows.

(1) Construct an initial path connecting two stable states of interest (possibly separated by intermediates), typically by linear interpolation.
(2) Discretize the path with points (images) uniformly spaced in arc length.
(3) Sample the phase space by simulating the dynamics of replicas constrained to remain in the vicinity of the images defining the path. Vanden-Eijnden and Venturoli[29] did this by infinite square-well confinement to Voronoi polyhedra defined by associating each point in the order parameter space with the image closest to it. Symbolically,

$$\Omega_i = \{\theta(x) | D_{ix} = \min\{D_{jx}\}\}, \qquad (3)$$

where Ω_i denotes the ith polyhedron, x is a point in phase space, the function $\theta(x)$ returns the position in the order parameter space given x in the full space, j is an integer in the range $1 \le j \le N_{\mathrm{im}}$, and D_{ix} is the distance between the point $\theta(x)$ and the ith image,

$$D_{ix} = \|z_i - \theta(x)\|, \qquad (4)$$

where z_i is the position of image i. Note that both z_i and $\theta(x)$ are vectors with the dimensionality of the order parameter space and $\|\cdot\|$ denotes the Euclidean norm. For systems that satisfy detailed balance, the infinite square-well confinement can be enforced by simply rejecting moves that lead out of a region in Monte Carlo simulations or reflecting the momenta in molecular dynamics simulations.[29]

(4) Update the string by moving the images individually using the results of the simulation in step (3). This is done using a three-part procedure.

(a) The images are moved toward the average positions of the replicas $\langle \theta(x_i) \rangle$ according to

$$z_i^\dagger = z_i^\circ + \tau(\langle \theta(x_i) \rangle - z_i^\circ), \tag{5}$$

where z_i° is the position of image i before the movement, τ is a parameter between 0 and 1 controlling the rate of the image movement, and

$$\langle \theta(x_i) \rangle = \frac{\Delta t}{T_f} \sum_{k=1}^{T_f/\Delta t} \theta(x_i(k\Delta t)), \tag{6}$$

where $x_i(t)$ is the position of replica i at time t, Δt is a small time interval comparable to the decorrelation time of the system, and T_f is the time at the end of the sampling period.

(b)   The string is then smoothed according to

$$z_i^\ddagger = \kappa(z_{i+1}^\dagger + z_{i-1}^\dagger - 2z_i^\dagger), \tag{7}$$

where κ is a positive parameter that controls the strength of the smoothing.

(c)   Finally, the string is reparametrized such that the images are spaced equally in arc length to ensure even sampling along the path. The new position of image i is determined by first finding the index j such that

$$\sum_{k=1}^{j} D_{k,k-1}^\ddagger < \left(\frac{i-1}{N_{im}-1} \right) D_{tot} < \sum_{k=1}^{j+1} D_{k,k-1}^\ddagger, \tag{8}$$

where N_{im} is the number of images, $D_{k,k-1}^\ddagger$ is the distance between image k and image $k-1$ before reparametrization as calculated in the order parameter space, $D_{1,0}^\ddagger = 0$, and D_{tot} is the total distance along the string. Image i is then placed at position z_i^f between $z_j^\ddagger$ and $z_{j+1}^\ddagger$ according to

$$z_i^f = z_j^\ddagger + (z_{j+1}^\ddagger - z_j^\ddagger) \left[\left(\frac{i-1}{N_{im}-1} \right) D_{tot} - \sum_{k=1}^{j} D_{k,k-1}^\ddagger \right]. \tag{9}$$

Without the reparametrization step, the images tend to drift toward the stable states, and the transition states become sampled less frequently.

(5)   Go to step (3).

The loop is iterated until the path stops changing within the desired level of precision. As discussed in more detail elsewhere[24,26] the efficacy of the method relies on the order parameters being able to describe the transition sufficiently well and, relatedly, the most probable transition paths being largely confined to a tube in the space of order parameters. It is not always obvious whether these conditions are satisfied for a given system and choice of order parameters, as we illustrate in the Appendix. In particular, for nonequilibrium processes, it is not guaranteed that the most probable forward and backward paths overlap (e.g., Ref. 6), and it could be necessary to define separate strings for each direction of the transition. For the examples considered here, the overlap between the forward and backward paths is sufficient that this issue can be neglected.

C. Nonequilibrium Voronoi-based sampling procedure

The algorithm introduced here uses the general outline of our previous nonequilibrium sampling method.[14] The order parameter space is covered with regions and a walker is placed in each. The regions are Voronoi polyhedra that are defined by images along a path that winds its way through a high-dimensional order parameter space as in the string method.[29] In analogy to the two overlapping lattices in the original algorithm, two strings are employed. The first string has N_{im} images, and the images of the second string lie at the midpoints of the line segments connecting successive images of the first string.

The flux probability distributions and weights of the regions are determined self-consistently. We update the weights as in the original algorithm [Eq. (1)], except that we reinitialize the instantaneous weights to their averages each time the averages are updated. The flux distributions, however, are sampled in a different fashion. Here, the sampling regions are irregularly shaped and, as the path evolves, their sizes, shapes, and connectivities change. Thus the flux distributions cannot be accumulated as histograms with equally sized bins as in Ref. 14. Instead, we randomly sample the probability density for the walker in region B on lattice k to be reintroduced at the point x in the full space, $\Phi_B^{(k)}(x)$:

$$\Phi_B^{(k)}(x) = \sum_{i \in \mathbb{F}_B^{(k)}} \bar{W}_{b_B(i)}^{(l)} \delta(x - x_i)/Z_B^{(k)}, \tag{10}$$

where $\mathbb{F}_B^{(k)}$ is a list of entry points for region B on lattice k that are accumulated using simulations on lattice $l \neq k$ ($k, l = 1, 2$ or $2, 1$), $\bar{W}_{b_B(i)}^{(l)}$ is the weight of the region $b_B(i)$ on lattice l where $b_B(i)$ is an indicator function that returns the region in which point i was recorded, $\delta(x - x_i)$ selects the points with the x value of interest, and

$$Z_B^{(k)} = \int_{\Omega_B^{(k)}} \sum_{i \in \mathbb{F}_B^{(k)}} \bar{W}_{b_B(i)}^{(l)} \delta(x - x_i)\,dx \tag{11}$$

is a normalization factor. The weights are updated simultaneously, and in Eq. (10) we use the value of $\bar{W}_{b_B(i)}^{(l)}$ at the time that the crossing was recorded. To reinitialize the walker after an attempted exit, a point is chosen from $\mathbb{F}_B^{(k)}$ with likelihood proportional to $\Phi_B^{(k)}(x)$, and since x fully defines the system, no further coordinates need to be determined. In practice, $\mathbb{F}_B^{(k)}$ is a list of N_{list} phase space points in which the oldest entries are overwritten once the list is full. If early in the simulation $\mathbb{F}_B^{(k)}$ does not yet contain any entries, we reinitialize the replica to the position of the image, but other choices might be preferable in specific systems. Because there are typically a relatively small number of images, boundary crossings between regions can be detected simply by monitoring the distance of each walker from the images.

After a specified length of time (T_{move}), the average position of each replica is determined, and the images of the first string are moved toward their average positions according to Eq. (5); the string is then smoothed and reparametrized by applying Eqs. (7) and (9). The images of the second string are then placed at the midpoints of the line segments con-
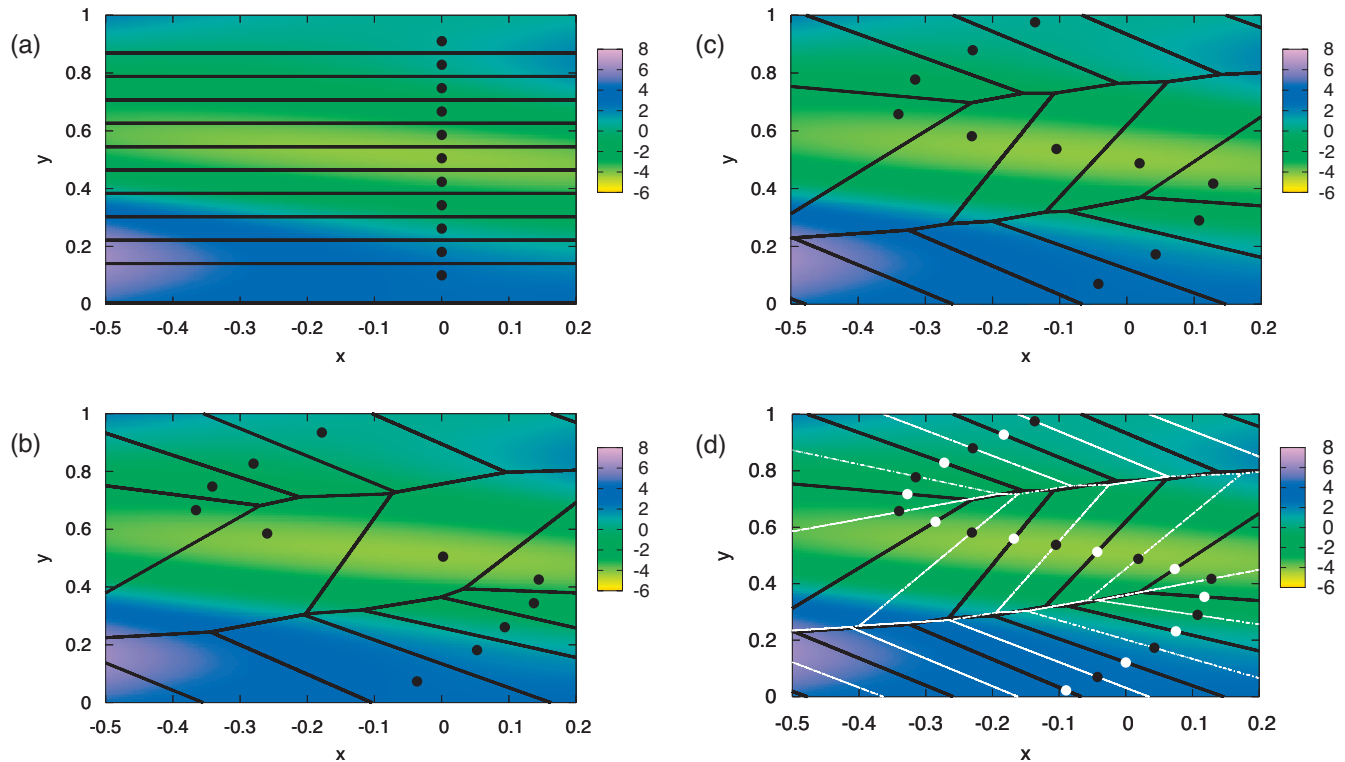
FIG. 3. (Color) Plots from the 2D periodic model illustrating features of the algorithm. (a) A path of 11 images (circles) is shown with their associated Voronoi polyhedra, outlined in black. (b) The images from (a) are moved and new regions are defined. (c) The path is reparametrized such that the images are equidistant, but it is otherwise unchanged. (d) The images and Voronoi polyhedra of the second string are superimposed on (c) and shown in white; by definition, the images in the second set lie on the boundaries of the first. In all plots, the colors indicate an effective potential surface accounting for a driving force, and the scale bar is in units of kT.

necting successive images of the first string. After the strings are modified, the regions are changed and the flux lists are no longer valid. From each list, we remove only the points that are no longer contained by the corresponding region. Selectively (rather than completely) pruning the lists speeds convergence because the errors introduced are typically small when the images do not move too much and are mitigated by continually refreshing the lists. The evolution of the string is illustrated with a 2D system in Fig. 3.

D. Summary

In summary, our algorithm proceeds as follows. A set of points (images) in the order parameter space is initialized arbitrarily (typically, the initial and final states are linearly interpolated); these images form the first string. A second string is created by placing images at the midpoints between the segments connecting successive images of the first string. The images of each string define Voronoi polyhedra that cover the entire order parameter space (an irregular lattice). A separate simulation is run in each region, during which crossings of the boundaries on the other lattice are recorded. In this fashion, flux lists $(\mathbb{F}_B^{(k)})$ are built for all regions on both lattices. These lists contain all of the observed entry points and their corresponding weights. Once a replica tries to leave its own region B and enter a region B', the weights are updated by an additive transfer as in Eq. (2). The replica is then reinitialized at $x \in \mathbb{F}_B^{(k)}$ with probability density $\Phi_B^{(k)}(x)$ given by Eq. (10). After T_{move} steps, the first string is updated as in Eqs. (5)–(9). The second string is moved by

again placing its images at the midpoints between the segments connecting successive images of the first string. The flux lists are pruned, and sampling is continued. In practice, we divide the simulation into two phases (I and II). Phase I follows the outline immediately above and continues until the path stops changing significantly. In phase II, the positions of the images are frozen ($T_{\text{move}} \rightarrow \infty$) and $\{W_B^{(k)}\}$ and $\{\Phi_B^{(k)}(x)\}$ are determined more accurately. Fixing the images is necessary for complete convergence of the weights and fluxes since the values of these quantities depend on the definitions of the regions.

III. EXAMPLE I: PERIODIC 2D SYSTEM

To illustrate the method, we first apply the nonequilibrium Voronoi-based sampling procedure to a periodic 2D system where the potential surface is defined by

$$V(x,y) = \gamma\left[x - \tfrac{1}{2}\sin(2\pi y)\right]^2 + \alpha\cos(2\pi y), \qquad (12)$$

where x and y are the spatial coordinates and γ and α are adjustable parameters. In addition to the potential, the dynamics are determined by a constant external force in the y direction ($\mathbf{F}_{\text{ext}} = F\hat{y}$). The system thus evolves according to the equation of motion:

$$\mathbf{r}(t + \delta t) = \mathbf{r}(t) - \frac{\delta t}{m\xi}(\nabla_{\mathbf{r}}V - \mathbf{F}_{\text{ext}}) + \delta \mathbf{r}^G, \qquad (13)$$

where δt is the numerical integration time step, $\delta \mathbf{r}^G$ is a noise term with components chosen randomly from a Gaussian

TABLE I. Parameters used in the umbrella sampling algorithm for the periodic 2D potential.

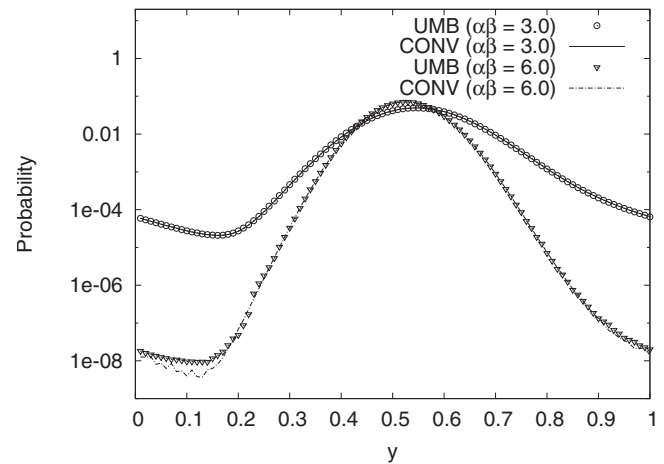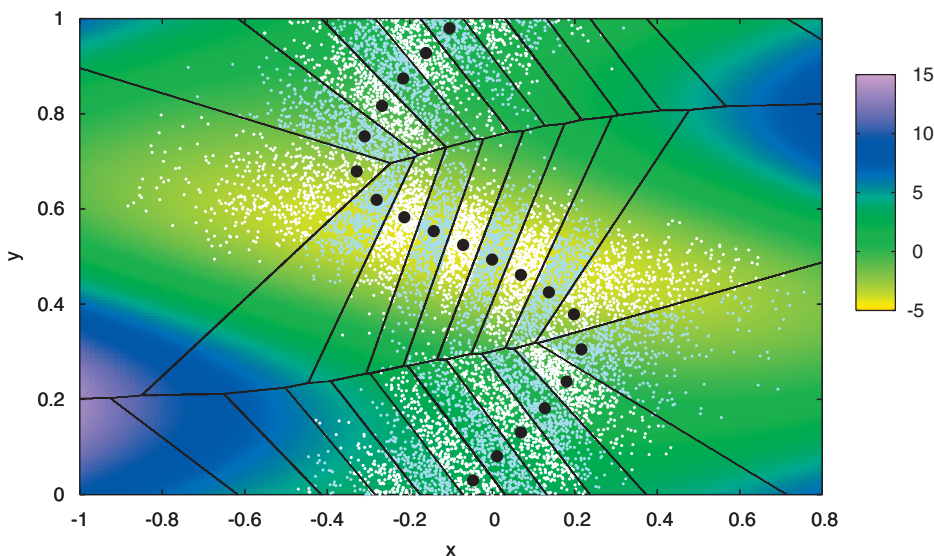|	Common		Rare	
	Phase I	Phase II	Phase I	Phase II
$T/\delta t$	1×10^6	1×10^7	2×10^6	1×10^7
$T_{\text{move}}/\delta t$	10 000	∞	10 000	∞
N_{list}	8 000	8 000	8 000	8 000
N_{im}	10	10	30	30
$T_{\text{wt}}/\delta t$	10 000	10 000	10 000	10 000
s	5×10^{-4}	5×10^{-4}	5×10^{-4}	5×10^{-4}
τ	0.1	$\cdots$	0.1	$\cdots$
$\Delta t/\delta t$	10	$\cdots$	10	$\cdots$
κ	0.1	$\cdots$	0.3	$\cdots$

FIG. 5. Projections onto the y axis for the 2D periodic system. Each curve is an average of ten independent simulations. The conventional simulations (CONV) were initialized in the basin $\mathbf{r} = (0.0, 0.5)$ and integrated according to Eq. (13) for 1.5×10^8 steps for the $\alpha\beta = 3.0$ case in which transitions are common and 1.0×10^9 steps for the $\alpha\beta = 6.0$ case in which transitions are rare. Details of the umbrella sampling simulations (UMB) are given in Table I.

distribution with zero mean and variance $2D\delta t$, and $\mathbf{r} = (x, y)$. D is the diffusion coefficient, which is related to the friction coefficient ξ and the inverse temperature β by $D = (m\beta\xi)^{-1}$. We restrict $y \in [0, 1)$ with periodic boundary conditions. The constant force creates a unidirectional flow through the system, which drives it out of equilibrium.

For the results presented here, $\gamma\beta = 6.0$, $\delta t = 0.005$, $\xi = 3.0$, and $F = 4.8$. Two different values of the depth of the cosinusoidal contribution to the potential (α) are used: one where transitions are common ($\alpha\beta = 3.0$) and another where transitions are rare ($\alpha\beta = 6.0$). There are a number of different parameters associated with the umbrella sampling algorithm: T is the total time of the simulation for each region, T_{move} is the string update period, N_{list} is the size of the flux list, N_{im} is the number of images used, T_{wt} is a parameter controlling the weight averaging, s is the weight transfer parameter defined in Eq. (2), τ is the image movement parameter given in Eq. (5), Δt controls the discretization of the sum in Eq. (6), and κ is the smoothing parameter defined in Eq. (7). The values of these parameters are given in Table I for the different cases and simulation phases. In each simulation, the string is initialized with its images equidistant along the y axis and phase I runs until the path stops changing significantly.

Figure 4 shows a converged configuration of the string

and the Voronoi polyhedra. To evaluate the sampling, we examine projections of the steady-state distribution onto the y axis, which measure the relative probability of the system being in the basin (around $y = 0.5$) to the transition state ($y = 0$ and $y = 1$) (Fig. 5). These projections are constructed by summing the contributions from different regions according to their weight. Good agreement with long conventional simulations is obtained for both the choices of α. We measure error by the difference between the accumulated y-axis projections and target histograms on a logarithmic scale:

$$\text{error} = \left(\frac{1}{n} \sum_{i=1}^{n} E_i^2 \right)^{1/2} \qquad (14)$$

and

FIG. 4. (Color) The 2D potential surface [Eq. (12)], accounting for the external force, with the parameters for which transitions are common. A path of 20 images and its Voronoi polyhedra is superimposed. The images are shown in black. For each region, a random sample of 500 points is shown in white or light blue. The scale bar is in units of kT. See text for parameters.

(a) UMB ——— CONV ·······
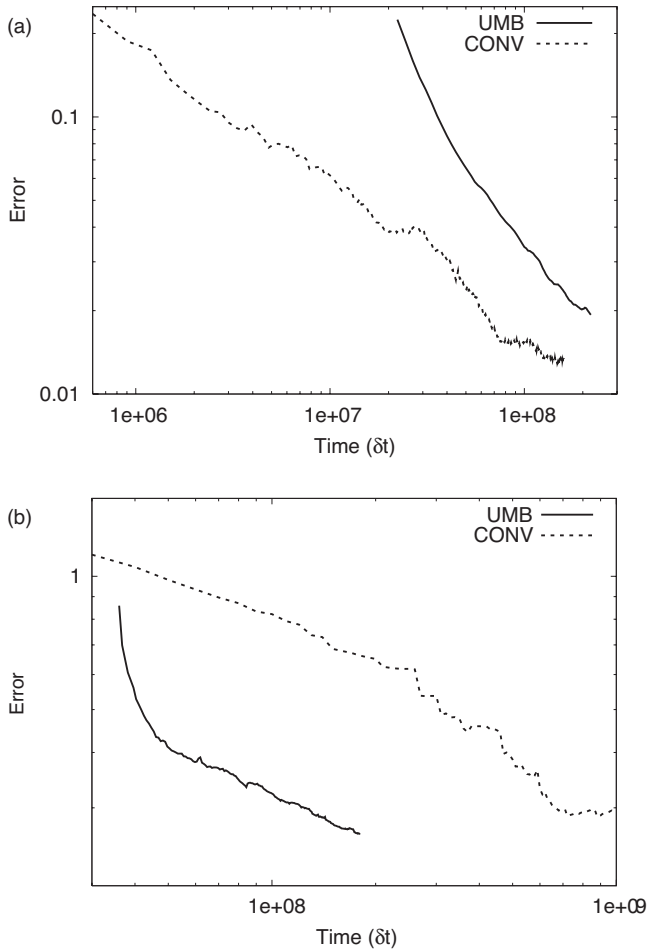

(b) UMB ——— CONV ·······

FIG. 6. Convergence of the projections in Fig. 5 as measured by the error defined in Eq. (14) for (a) the $\alpha\beta=3.0$ parameter set and (b) the $\alpha\beta=6.0$ parameter set. Both the conventional and umbrella sampling errors shown are averages over individual error curves obtained from ten independent simulations. The time for the umbrella sampling plots includes the path evolution time and it is the total time for all regions on both lattices. The error for the umbrella sampling algorithm is calculated by averaging the y axis projections starting 1.0×10^6 steps after the start of phase II in both cases to exclude early inaccurate results from the averages. In the "common" case, the target histogram is built using a single conventional simulation of 1.5×10^9 steps. In the "rare" case the target histogram was an average of ten conventional simulations, each of 1.5×10^{10} steps. The target simulations were independent of those used to calculate the error of the conventional simulations.

$$E_i = \begin{cases} \log P(i) - \log P_t(i) & \text{for } P(i) \neq 0 \\ \log 1/T - \log P_t(i) & \text{for } P(i) = 0, \end{cases} \quad (15)$$

where P is the y-axis projection that is discretized into n evenly spaced windows from 0 to 1 (here n is set to 100), P_t is the target histogram, and T is the total time of the simulation. The special case for $P(i)=0$ is introduced to avoid infinite errors when y-axis projections are incomplete. This choice only impacts the conventional efficiency curves, since the umbrella y-axis projections span [0,1] almost immediately. The error is determined using $(\log P(i) - \log P_t(i))$ rather than $(P(i) - P_t(i))$ to emphasize low probability regions of the y-axis histograms, which are the most difficult to sample. Figure 6 compares the convergence of simulations with the $\alpha\beta=3.0$ and the $\alpha\beta=6.0$ parameter sets. It is seen that for the $\alpha\beta=3.0$ case conventional sampling is sufficient,
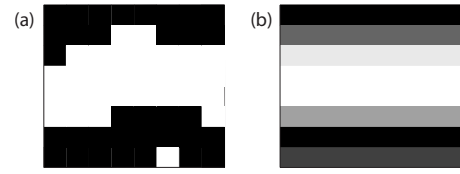

FIG. 7. A lattice of 8×8 Ising spins before and after the coarse-graining procedure. (a) The full state of the system, with the black squares representing $\sigma_{ij}=-1$ and the white squares representing $\sigma_{ij}=1$. (b) The corresponding coarse-grained representation as defined in Eq. (19).

but in the $\alpha\beta=6.0$ case conventional sampling requires five to ten times more steps than the umbrella sampling algorithm to converge to the same level of precision.

IV. EXAMPLE II: THE ISING MODEL UNDER SHEAR

Although 2D systems are useful for the purpose of visualization, they are not the targets of the method introduced in the present work. Effective projection of the dynamics onto the one-dimensional string does not provide a large boost in computational efficiency because the space is already small. Moreover, in a 2D system the mechanism for transition is straightforward and appropriate order parameters can be easily found. We thus now consider a many-dimensional example that better demonstrates the strengths of the method.

To this end, we study the kinetics of a 2D Ising model with nearest neighbor interactions. The system is composed of a square lattice of $L \times L$ sites, at each of which there is a binary variable (spin). We denote the spin at row i and column j by σ_{ij}; $\sigma_{ij}=-1$ or 1. The Hamiltonian of the (microscopically reversible) system is

$$H(\{\sigma_{ij}\}) = -J \sum_{\langle ij,kl \rangle} \sigma_{ij}\sigma_{kl} - h \sum_{i=1}^{L} \sum_{j=1}^{L} \sigma_{ij}, \quad (16)$$

where J is the coupling constant between the spins, h is a uniform external magnetic field, and $\langle ij, kl \rangle$ restricts the sum to nearest neighbors on the lattice. Intuitively, there are two stable states in this system: one with most of the spins pointed up ($\sigma_{ij}=1$) and one with most of the spins pointed down ($\sigma_{ij}=-1$). The overall state of the system can be characterized by the magnetization

$$M = \frac{1}{L^2} \sum_{i=1}^{L} \sum_{j=1}^{L} \sigma_{ij}. \quad (17)$$

We study the mechanism of transition between the $M \approx -1$ and the $M \approx 1$ states under shear with periodic boundary conditions. We define the elementary dynamics for consistency with Allen et $al.$[11] Namely, we flip a single spin in each Monte Carlo move and accept the change with probability

$$P(\{\sigma_{ij}\} \rightarrow \{\sigma_{ij}\}') = \min\{e^{-\beta[H(\{\sigma_{ij}\}') - H(\{\sigma_{ij}\})]}, 1\}, \quad (18)$$

where $\{\sigma_{ij}\}$ and $\{\sigma_{ij}\}'$ are the configurations of the system before and after the spin flip, respectively. In this scheme, M is not conserved. Shear is simulated as in Ref. 11 but altered to subtract the average velocity field. This alteration is made so that the average configuration associated with each image can be more clearly determined. Shear moves are conducted
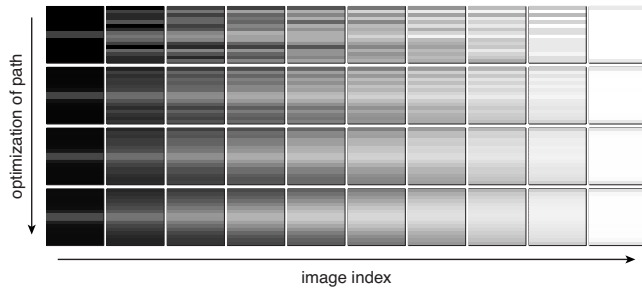
FIG. 8. The evolution of the path over the course of the simulation for the Ising model under shear with $L=16$. From top to bottom, the plots show the state of the path at 0%, 8%, 20%, and 100% of the phase I simulation length. The parameters used to generate the paths are given in Tables II and III.

after every cycle ($L \times L$ steps) by choosing a row (r) at random. Assuming for convenience that L is even, if $r > L/2$ then all rows *above* that row are shifted to the *right* by one lattice site with probability P_{sh}. If $r \leq L/2$ then all rows *below* that row are shifted to the *left* by one lattice site with probability P_{sh}. L shear moves are attempted after each cycle. We use the value $P_{\text{sh}}=0.02$ for all simulations below. Lees-Edwards boundary conditions[2,31] are enforced through the nearest neighbor lists at the top and bottom rows.

As mentioned in Sec. II B, the efficacy of string-based methods is dependent on the transition paths being limited to a tube in the order parameter space. This localization condition is trivially violated by many order parameter choices if there is translational invariance in the system. To remove this degeneracy, we introduce four fixed spins in the center of the

lattice: $\sigma_{L/2,L/2}=\sigma_{L/2,L/2+1}=\sigma_{L/2+1,L/2}=\sigma_{L/2+1,L/2+1}=1$. These spins seed the forward transition. To keep the original symmetry between "up" spins and "down" spins, we introduce four fixed spins at the corners of the lattice: $\sigma_{L,L}=\sigma_{L,1}=\sigma_{1,L}=\sigma_{1,1}=-1$. With $h=0$, the forward transition and the backward transition are equally probable, and the constraint that the probability of magnetization is symmetric about $M=0$ can thus serve as an additional error check. To keep the clusters intact, we do not shear between $y=L/2$ and $y=L/2+1$ or at the periodic boundary in y. Note that the two clusters move with respect to one another as shear is applied. The localization condition discussed above (termed "the string assumption") and its relation to the convergence of the method presented here are further discussed in the Appendix.

A previous application of the string method to Ising systems used local magnetizations over regions of spins as the order parameters.[28] Here, we base the umbrella sampling on local magnetizations of the rows of the lattice (Fig. 7):

$$M_j = \frac{1}{L}\sum_{i=1}^{L} \sigma_{ij}, \tag{19}$$

where i indexes the columns of the lattice. $\{M_j\}$ is invariant to shear moves. Figure 8 shows the evolution of the path for $L=16$ from a linear interpolation to its final form. The changes are most visible in the middle of the path, where $M \approx 0$. In Fig. 9, we compress the information from one snapshot of the lattice in Fig. 8 into one column (a particular image index) and show examples of converged transition paths in the order parameter space for $L=6, 8, 12,$ and 16.
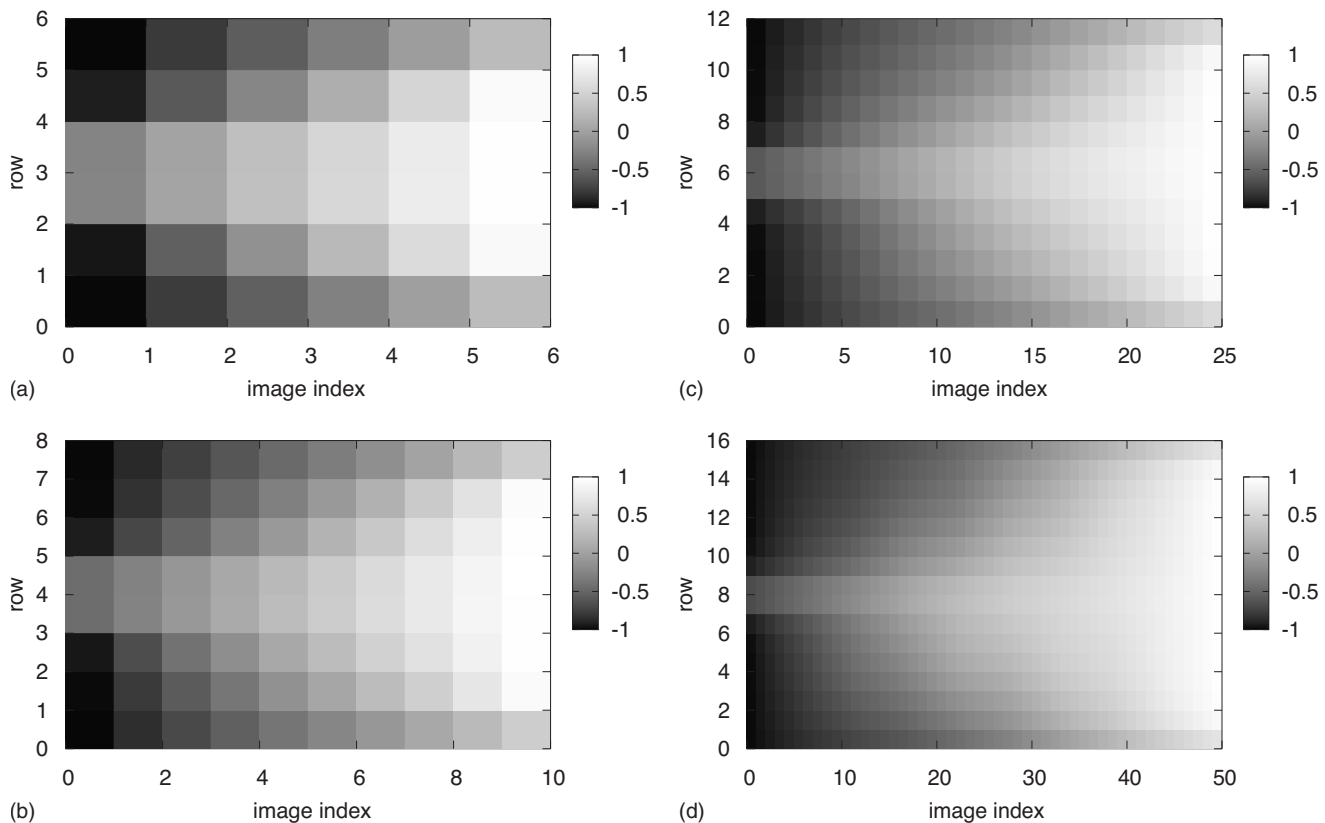

FIG. 9. Examples of converged paths for different Ising system sizes. From (a) to (d), the system sizes are $L=6, 8, 12,$ and 16. The parameters used to obtain these paths are given in Tables II and III.

TABLE II. Parameters used for the Ising model simulations regardless of the value of L.

Parameter	Value
βJ	0.65
βh	0.00
P_{sh}	0.02
N_{list}	5000
s	0.005
τ	0.1
$\Delta t / \delta t$	200
κ	0.4

The parameter sets used to obtain these paths are given in Tables II and III. Each plot shows the progression of the order parameters along the path. For all the plots, the leftmost image is mostly composed of down spins with a gray band in the middle that reflects the fixed up spins. As the image index increases, the gray band becomes more pronounced and spreads out until most of the spins are up, except for those fixed down at the top and bottom.

The actual structures associated with each image need not be homogeneous across each row. To give the reader a better sense of these structures, we calculate the average configuration of the walker associated with each image (Fig. 10). These averages reveal the general mechanism for transition, which is an enlargement of the central cluster until it spans the lattice in the direction of the applied shear, followed by growth perpendicular to the shear. This mechanism is consistent with structures observed in other simulation studies.[11,32]

To quantify the convergence, we project the sampled steady-state distribution, which is a function of L order parameters, onto the total magnetization of the system (M). The probability of magnetization is obtained by summing magnetization histograms for each image in proportion to its weight. Figure 11 shows the probability of magnetization for $L=6$, 8, 12, and 16 as computed by the umbrella sampling algorithm. For $L=6$, 8, and 12, we compare the results from the umbrella sampling with ones from conventional simulations. For $L=16$, conventional simulations become unfea-
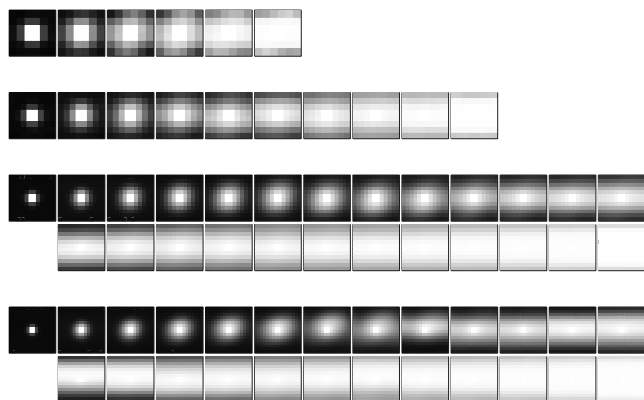

FIG. 10. Average configurations associated with the images for different Ising system sizes. From top to bottom, the system sizes are $L=6$, 8, 12, and 16. For the $L=16$ simulations, there are 50 images and only every other image is shown.
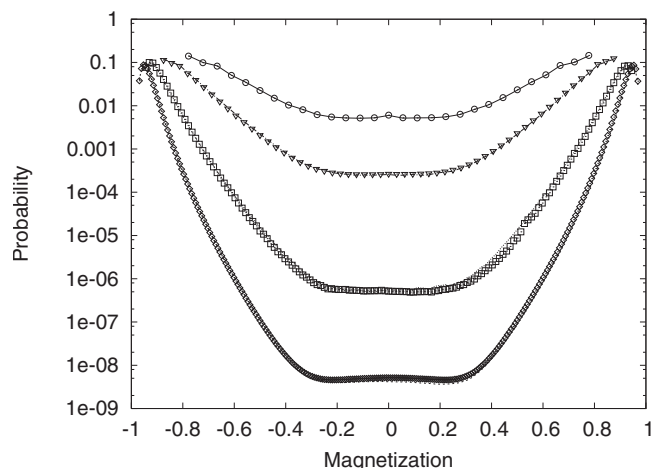

FIG. 11. Comparison of probability of magnetization curves obtained by the umbrella sampling algorithm and references for different Ising system sizes. For $L=6$, 8, and 12 the conventional Monte Carlo sampling is used as a reference, while for $L=16$ a one-dimensional implementation of the umbrella sampling algorithm is used. The reference results are shown with lines and the umbrella sampling algorithm with symbols. From top to bottom, the system sizes are $L=6$, 8, 12, and 16. The magnetization curves for the umbrella algorithms are time averages starting once the weights have converged to their steady-state values and continuing until the end of the simulation. The parameters used in these simulations are given in Tables II and III. All curves (except for the one-dimensional implementation) are the average of ten independent simulations. For each of the conventional simulations, we used 1×10^8 steps for $L=6$ and 8 and 1×10^9 steps for $L=12$. For the one-dimensional umbrella sampling simulations with $L=16$, $T=3\times10^8\delta t$, $N_{im}=30$, $N_{list}=5000$, $N_{wt}=200,000$, $s=0.005$, and $\Delta t=20\delta t$.

sible, and we compare with the output of the original umbrella sampling algorithm[14] with M as the only order parameter and equal spacing of the interfaces. The algorithm differed slightly from that in the original study in that Eq. (10) was used for the flux distributions.

For $L=12$ and 16 we used a larger number of images in phase II than in phase I (Table III). Although a large number of sampling regions facilitate convergence of the weights and fluxes, the configuration of the path can be determined efficiently using a smaller number of images. Indeed, we found that in general, during the optimization, paths stabilized well before the steady-state distribution itself was accurate. Conversely, good projections onto the magnetization could be obtained even if the paths from phase I varied somewhat from one simulation to another simulation. In these regards, it is also worth noting that the visual appearance of the path was much more robust to alternate choices of order parameters than was the probability of magnetization, as discussed in the Appendix.

Each of the ten umbrella sampling runs took approximately 1.0, 1.5, 17, and 45 h of central processing unit (CPU) time for $L=6$, 8, 12, and 16, respectively, on a 2.2 GHz AMD Opteron processor, but these timings depend on the choice of N_{im}. As a comparison, conventional simulations for $L=16$ were run for 2×10^{12} steps, requiring 262 CPU hours and failed to produce results of similar quality to those obtained using the umbrella sampling algorithm. While the Ising system is better suited for the stringlike method than the 2D model considered above, it is still sufficiently simple that the overhead associated with the algorithm dominates

TABLE III. Parameters that varied with L for the Ising model simulations.

L	Phase	N_{im}	$T_{move}/\delta t$	$\Delta T_{wt}/\delta t$	$T/\delta t$
6	I	6	10 000	20 000	2×10^5
	II	6	∞	20 000	1×10^7
8	I	10	10 000	20 000	2×10^5
	II	10	∞	20 000	2×10^7
12	I	10	50 000	20 000	5×10^6
	II	25[a]	∞	200 000	2×10^8
16	I	10	100 000	200 000	5×10^6
	II	50[a]	∞	200 000	2×10^8

[a]Between phase I and phase II, the number of images was increased using linear interpolation.

the simulation time, at least for these lattice sizes. We thus expect the comparative advantage of the umbrella sampling to be even greater for more complex systems.

V. DISCUSSION

Here we have introduced a quasi-one-dimensional non-equilibrium umbrella sampling algorithm for application to large order parameter spaces. It works by evolving a collection of images that when put together form a string that winds through the order parameter space from a reactant region to a product region. The images are evolved using time-averaged statistics of walkers, which are separate simulations confined to the Voronoi polyhedra defined by the images. In this way, the algorithm focuses sampling on the transition tube connecting the initial and final states. It is basically an extension of our previous work,[14] with the string acting as a convenient way to establish sampling regions in phase space. The primary change to the sampling component of the algorithm was a streamlined means of determining the fluxes into each region. Computation time still scales linearly with the number of replicas, but now those replicas can be allocated more efficiently.

Fundamentally, both the original and stringlike versions of nonequilibrium umbrella sampling are based on accumulating statistics of paths leading from one interface in a space of order parameters to another. There are now several methods of this general nature, and all share certain features. At the same time, they differ in detail and, as a result, applicability. Most other algorithms were designed for obtaining rates for transitions between two attractors. They have the advantage over our algorithms that they can determine a rate for one direction of a process without a well-defined steady state (e.g., see Ref. 12). On the other hand, our methods are designed for enhanced sampling of the steady-state probability distribution, the analog of the free energy for irreversible systems. The iterative nature of the algorithms makes them robust to errors that arise early in a simulation, and they can be applied to the wide range of problems in which a steady state exists but there is no transition with a separation in time scales between the dynamics within and between stable states.

The earliest interface-based algorithm for molecular systems of which we are aware is the weighted ensemble method.[33,34] The method works by simulating an ensemble of particles, each carrying a probability. The particles can clone themselves (giving half of the probability to each copy) or merge with other particles (combining their probability). Particles are cloned in low probability regions and merged in high probability regions to obtain a more even sampling along a progress coordinate that connects the initial and final states. Although introduced to study systems in equilibrium, this method can be used to study nonequilibrium ones as well. Indeed, the "branched growth" version of FFS (Ref. 7) is essentially equivalent to it; FFS is discussed in Sec. I. Since we introduced our original nonequilibrium umbrella sampling algorithm,[14] we became aware of a similarly named method[35,36] that ratchets the system forward like FFS. This unidirectional procedure fails to produce an accurate steady-state distribution for the reasons discussed by Valeriani *et al.*[9] in justifying the need for two FFS simulations.

In the transition interface sampling[37,38] and milestoning methods,[39–41] the sampling assumes microscopic reversibility. These methods are thus inapplicable to the class of systems considered in the present study. In particular, in published versions of milestoning, systems are initiated from the interfaces with Boltzmann weighting to maximize parallelization. In contrast, nonequilibrium umbrella sampling does not assume a particular distribution on the interfaces *a priori*. That said, one could view the nonequilibrium umbrella sampling method as an expanded form of milestoning in which the statistics of the trajectories are used to improve the distributions on the interfaces iteratively to capture deviations from equilibrium and obviate the loss of memory assumption. However, rates can be calculated exactly by combining the trajectories obtained piecewise in nonequilibrium umbrella sampling rather than approximately through interface-to-interface passage times.[42]

All of the above methods assume a fixed set of locally nonintersecting interfaces defined by their spacing along a single coordinate. Their efficiencies thus depend sensitively on the choice of that coordinate. Our nonequilibrium umbrella sampling algorithm enables one to constrain the sampling with an arbitrary number of order parameters. In the present study, we exploited this feature to introduce a stringlike approach to improve the interfaces. Such an approach is expected to be especially important for systems with complex compositions such as biomolecular ones, where manual selection of a progress variable for a reaction is often very challenging. The ability to constrain the sampling with an arbitrary number of order parameters could also facilitate

bridging between separate reaction channels to obtain their relative probabilities (see Ref. 43 for progress in this regard for reversible systems).

The FFS-LSE (for "least-squares estimation") method is an alternate approach introduced recently for selecting interfaces.[10] The method builds on the idea first introduced by Ma and Dinner[22] and subsequently refined by Peters *et al.*[44,45] (see Methods in Ref. 46 for discussion) that statistical methods can be used to identify physical variables that correlate with a representative set of precalculated stable-state commitment probabilities (commitors). In the FFS-LSE method, the commitors are approximated in terms of quantities accumulated in FFS simulations, and these estimates are then fitted with linear combinations of physical variables to identify improved interfaces. Borrero and Escobedo[10] selected the parameters manually, which limits the number of order parameters that can be considered, but more sophisticated methods have been demonstrated to enable the efficient search of combinations of thousands of order parameters (without the need for repeated calculation of commitors or the so-called histogram test, contrary to the statements in Refs. 10 and 44).[22,46] Such statistical approaches are complementary to that taken here, and it will be interesting to explore whether they can be combined to advantage. Because statistical approaches are likely to be sensitive to the quality and distribution of the commitor estimates, we expect stringlike methods to be more effective for the initial determination of the path when little is known about a mechanism; statistical approaches could be useful though for interpreting the path and accelerating convergence of additional quantities (e.g., rates).

While the present study represents an important first step toward being able to treat reactions in complex, heterogeneous irreversible systems, some advances in the details of implementing the algorithm are likely to be required for realizing this goal. Larger systems will require that the lists of saved boundary points be written to disk rather than carried in memory. Care will be required to prevent the overhead associated with accessing the lists from becoming prohibitively computationally costly. For the simulations in the present study, the limiting feature of the algorithm was the redistribution of the weights between high and low probability regions. We are thus also investigating whether schemes based on solution of an overall balance condition can accelerate convergence.

In closing, it is important to note that, in the present study, the string serves only as a means of restricting sampling of the steady-state distribution to the most important parts of the order parameter space. The dynamical meaning of the string itself is imprecise. Recently, a stringlike method for treating nonequilibrium processes that can be characterized by a Landau–Ginzburg-like action has been introduced and applied to modeling nucleation under shear.[32] It will be interesting in the future to explore whether such an approach can be combined with the more general sampling algorithm presented here to improve the interpretation of the path.
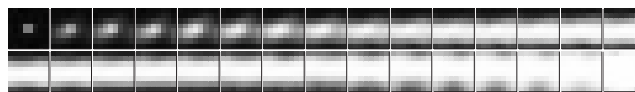

FIG. 12. A representative path obtained with the box order parameters $\{B_{ij}\}$ defined in the Appendix. This path has $L=16$, $N_{\text{box}}=8$, and $N_{\text{im}}=30$.

APPENDIX: CHOICE OF ORDER PARAMETERS

Here we discuss the effect of the choice of order parameters on the convergence of the system by examining the Ising model with periodic boundary conditions using a different set of order parameters:[28]

$$B_{ij}^n = \left(\frac{N_{\text{box}}}{L}\right)^2 \sum_{\alpha=1}^{L/N_{\text{box}}} \sum_{\beta=1}^{L/N_{\text{box}}} \sigma_{kl}^n, \tag{A1}$$

where

$$k = (i-1)\left(\frac{L}{N_{\text{box}}}\right) + \alpha,$$
$$l = (j-1)\left(\frac{L}{N_{\text{box}}}\right) + \beta, \tag{A2}$$

$1 \le i, j \le N_{\text{box}}$, and n is the image index. All B_{ij} are between -1 and 1, and the dimensionality of the order parameter space is now $(N_{\text{box}} \times N_{\text{box}})$. In summary, each image is divided into $N_{\text{box}} \times N_{\text{box}}$ coarse-grained boxes, each containing $(L/N_{\text{box}})^2$ spins. We will refer to this choice of order parameters as the "box" representation and the set of order parameters in the main text as the "stripe" representation.

In comparison with the simulations in the stripe representation, the paths now contain more information, and we can immediately visualize the path (Fig. 12). We found that this set of order parameters is able to reproduce the results shown above for $L=6$ and 8, but for higher values of L we were unable to converge to the correct steady-state distribution even for large numbers of images. This is interesting since (as is seen in Fig. 12) for much of the path, there is not much variation in B_{ij} along the row (in the direction of the applied shear), and intuitively there should not be such a difference between the results obtained using the two different sets of order parameters. Nonetheless, there are key differences between the two sets of order parameters that warrant further discussion.

We observed that there are a larger number of transitions between regions that are not adjacent in image index (nonlocal transitions) in the box representation than in the stripe representation. With all other parameters held constant (L

$=16$, $\kappa=0.4$, $N_{im}=50$), nonlocal transitions account for $20(\pm 3)\%$ of the total using $\{B_{ij}\}$ and $2.1(\pm 0.6)\%$ of the total using $\{M_j\}$ (error estimates are the standard deviation over ten independent trials in each case). Nonlocal transitions can have a destabilizing effect on the convergence of the algorithm since they can correspond to transitions between regions with drastically different weights. Consider two regions, A and B, that are adjacent in phase space, where W_A (the weight of region A) $\gg W_B$. If a transition occurs from A to B, then W_B will increase by sW_A, which could be an increase of several orders of magnitude. This makes W_B sensitively dependent on the flux function in region A and can inhibit convergence to steady state. Successive regions in image index can be ensured to have weights that are not drastically different by manipulating the parametrization of the path (image spacing), while spatially adjacent regions that are separated in image index cannot. This makes nonlocal transitions problematic.

It is interesting to note that the assumption that the majority of the reactive flux is concentrated in the neighborhood of the string in the space of order parameters (the string assumption) is equivalent to the assumption that the majority of transitions are local. This assumption has been previously specified as requiring the width of the transition tube to be small compared with the radius of curvature of the string.[24] The latter quantity is approximately equal to the distance between the string and the points at which the Voronoi boundaries meet. As nonlocal transitions are only possible if the reactive flux extends further than the point at which the Voronoi boundaries meet, we can see that the percentage of nonlocal transitions is a good way to measure violations of the string assumption.

[1] D. J. Evans and G. Morriss, *Statistical Mechanics of Nonequilibrium Liquids*, 2nd ed. (Cambridge University Press, New York, 2008).

[2] M. P. Allen and D. J. Tildesley, *Computer Simulation of Liquids* (Oxford University Press, New York, 1987).

[3] D. Frenkel and B. Smit, *Understanding Molecular Simulation: From Algorithms to Applications* (Academic, London, 2002).

[4] B. D. Todd and P. J. Daivis, Mol. Simul. **33**, 189 (2007).

[5] D. T. Gillespie, Annu. Rev. Phys. Chem. **58**, 35 (2007).

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

[7] R. J. Allen, D. Frenkel, and P. R. ten Wolde, J. Chem. Phys. **124**, 024102 (2006).

[8] R. J. Allen, D. Frenkel, and P. R. ten Wolde, J. Chem. Phys. **124**, 194111 (2006).

[9] C. Valeriani, R. J. Allen, M. J. Morellia, D. Frenkel, and P. R. ten Wolde, J. Chem. Phys. **127**, 114109 (2007).

[10] E. E. Borrero and F. A. Escobedo, J. Chem. Phys. **127**, 164101 (2007).

[11] R. J. Allen, C. Valeriani, S. Tanase-Nicola, P. R. ten Wolde, and D. Frenkel, J. Chem. Phys. **129**, 134704 (2008).

[12] R. P. Sear, J. Chem. Phys. **128**, 214513 (2008).

[13] J. Juraszek and P. G. Bolhuis, Biophys. J. **95**, 4246 (2008).

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

[15] A. Warmflash and A. R. Dinner, Proc. Natl. Acad. Sci. U.S.A. **105**, 17262 (2008).

[16] G. M. Torrie and J. P. Valleau, J. Comput. Phys. **23**, 187 (1977).

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

[18] M. Maienschein-Cline, A. Warmflash, and A. R. Dinner (unpublished).

[19] R. S. Maier and D. L. Stein, Phys. Rev. E **48**, 931 (1993).

[20] R. S. Maier and D. L. Stein, J. Stat. Phys. **83**, 291 (1996).

[21] G. E. Crooks and D. Chandler, Phys. Rev. E **64**, 026109 (2001).

[22] A. Ma and A. R. Dinner, J. Phys. Chem. B **109**, 6769 (2005).

[23] W. E, W. Ren, and E. Vanden-Eijnden, J. Phys. Chem. B **109**, 6688 (2005).

[24] W. Ren, E. Vanden-Eijnden, P. Maragakis, and W. E, J. Chem. Phys. **123**, 134109 (2005).

[25] W. E, W. Ren, and E. Vanden-Eijnden, Chem. Phys. Lett. **413**, 242 (2005).

[26] L. Maragliano, A. Fischer, E. Vanden-Eijnden, and G. Ciccotti, J. Chem. Phys. **125**, 024106 (2006).

[27] T. F. Miller, E. Vanden-Eijnden, and D. Chandler, Proc. Natl. Acad. Sci. U.S.A. **104**, 14559 (2007).

[28] M. Venturoli, E. Vanden-Eijnden, and G. Ciccotti, J. Math. Chem **45**, 188 (2009).

[29] E. Vanden-Eijnden and M. Venturoli, "Revisiting the finite temperature string method for calculation of reaction tubes and free energies," J. Chem. Phys. (in press).

[30] A. C. Pan, D. Sezer, and B. Roux, J. Phys. Chem. B **112**, 3432 (2008).

[31] A. W. Lees and S. F. Edwards, J. Phys. C **5**, 1921 (1972).

[32] M. Heymann and E. Vanden-Eijnden, Phys. Rev. Lett. **100**, 140601 (2008).

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

[34] B. W. Zhang, D. Jasnow, and D. M. Zuckerman, Proc. Natl. Acad. Sci. U.S.A. **104**, 18043 (2007).

[35] R. Blaak, S. Auer, D. Frenkel, and H. Löwen, Phys. Rev. Lett. **93**, 068303 (2004).

[36] R. Blaak and H. Lowen, Comput. Phys. Commun. **169**, 64 (2005).

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

[38] D. Moroni, P. G. Bolhuis, and T. S. van Erp, J. Chem. Phys. **120**, 4055 (2004).

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

[40] D. Shalloway and A. K. Faradjian, J. Chem. Phys. **124**, 054112 (2006).

[41] A. M. A. West, R. Elber, and D. Shalloway, J. Chem. Phys. **126**, 145104 (2007).

[42] A. Warmflash, A. Dickson, and A. R. Dinner (unpublished).

[43] T. S. van Erp, Comput. Phys. Commun. **179**, 34 (2008).

[44] B. Peters and B. L. Trout, J. Chem. Phys. **125**, 054108 (2006).

[45] B. Peters, G. T. Beckham, and B. L. Trout, J. Chem. Phys. **127**, 034109 (2007).

[46] J. Hu, A. Ma, and A. R. Dinner, Proc. Natl. Acad. Sci. U.S.A. **105**, 4615 (2008).