Enhanced Jarzynski free energy calculations using weighted ensemble

Cite as: J. Chem. Phys. 153, 134116 (2020); doi: 10.1063/5.0020600

Submitted: 1 July 2020 • Accepted: 16 September 2020 •

Published Online: 7 October 2020

Nicole M. Roussey¹ and Alex Dickson^{1,2,a)}

AFFILIATIONS

¹Department of Biochemistry and Molecular Biology, Michigan State University, East Lansing, Michigan 48823, USA

²Department of Computational Mathematics, Science and Engineering, Michigan State University, East Lansing, Michigan 48823, USA

^{a)} Author to whom correspondence should be addressed: alexrd@msu.edu

ABSTRACT

The free energy of transitions between stable states is the key thermodynamic quantity that governs the relative probabilities of the forward and reverse reactions and the ratio of state probabilities at equilibrium. The binding free energy of a drug and its receptor is of particular interest, as it serves as an optimization function for drug design. Over the years, many computational methods have been developed to calculate binding free energies, and while many of these methods have a long history, issues such as convergence of free energy estimates and the projection of a binding process onto order parameters remain. Over 20 years ago, the Jarzynski equality was derived with the promise to calculate equilibrium free energies by measuring the work applied to short nonequilibrium trajectories. However, these calculations were found to be dominated by trajectories with low applied work that occur with extremely low probability. Here, we examine the combination of weighted ensemble algorithms with the Jarzynski equality. In this combined method, an ensemble of nonequilibrium trajectories are run in parallel, and cloning and merging operations are used to preferentially sample low-work trajectories that dominate the free energy calculations. Two additional methods are also examined: (i) a novel weighted ensemble resampler that samples trajectories directly according to their importance to the work of work and (ii) the diffusion Monte Carlo method using the applied work as the selection potential. We thoroughly examine both the accuracy and efficiency of unbinding free energy calculations for a series of model Lennard-Jones atom pairs with interaction strengths ranging from 2 kcal/mol to 20 kcal/mol. We find that weighted ensemble calculations can more efficiently determine accurate binding free energies, especially for deeper Lennard-Jones well depths.

Published under license by AIP Publishing. <https://doi.org/10.1063/5.0020600>

I. INTRODUCTION

The importance of free energy calculations from atomic simulations has been well established for many biological and chemical applications. The free energy of a transition in a biological system is the key thermodynamic quantity governing the transition's reversibility and the work required to perform the transition in an unfavorable direction. In this way, free energy forms the theoretical underpinnings of a range of biological processes including ligand (un)binding, protein folding, solubility, and biologically significant conformational changes in protein structures.¹ Free energy calculations also provide a means of estimating other relevant values for biological processes such as permeability coefficients and rate constants.¹ Due to the significance of this value, new,

computationally efficient methods of calculating free energies are always of interest.

Increases in computational power, new developments in enhanced sampling, and advances in statistical mechanics have led to the development of a wide variety of methods to calculate free energies (see Refs. 1–3). Despite decades of systematic improvement, there are complications and drawbacks associated with each method. Relative binding free energies can now be routinely calculated by free energy perturbation (FEP) with robust computational tools available (e.g., FEP+ and Schrödinger⁴); however, these are limited to sets of ligands with high similarity.⁵ There are also a set of methods to calculate the absolute binding free energy for a protein–ligand or protein–protein complex. This can be done either through physical sampling of the unbinding pathway [e.g., umbrella

sampling,^{6,7} Markov state modeling,^{8,9} metadynamics,^{10,11} milestone-ing,^{12,13} and weighted ensemble (WE) strategies^{14,15} or through alchemical transformations¹⁶ (e.g., using double decoupling¹⁷ with thermodynamic integration¹⁸ or Hamiltonian replica exchange¹⁹) and varying combinations of these free energy estimators and methods. The two main obstacles of physical sampling strategies are (i) convergence of the free energy estimate and (ii) projection of the binding process onto one or more order parameters. In alchemical transformations, single topology models are hindered by residual charges from the vanishing atoms that can create large forces in the system and double topology models can have overlap between groups leading to instability in the system.¹ Alchemical transformations also only predict the free energy difference of the endpoints and cannot produce complete free energy profiles. Even for small systems such as host–guest pairs used in recent SAMPL challenges,²⁰ computational free energies obtained with these methods are often not consistent with each other and can deviate by several kcal/mol from experimentally determined values.^{20,21}

An alternative method for calculating the free energy of a process at equilibrium (ΔF) is to exploit a relationship between ΔF and the work applied to a system by a nonequilibrium force (W). The maximum work theorem states that in a nonequilibrium process, the ensemble average of work required, $\langle W \rangle$, is greater than the difference in free energy, ΔF , between the initial and final states of the process,²²

$$\langle W \rangle \geq \Delta F. \quad (1)$$

Differences between individual trajectories are significant in microscopic systems and can result in very different values of work being observed for the same process. For most realizations of a process, the work done is greater than ΔF ; however, the work can be smaller than the free energy difference (seemingly violating the second law of thermodynamics).^{22,23} Performing many realizations of a process produces a statistical distribution of work values for that process. These statistical distributions of W can be accounted for in an “equality transformation” of the maximum work theorem²² referred to as the Jarzynski equality,²³

$$\langle e^{-\beta W} \rangle = e^{-\beta \Delta F}, \quad (2)$$

where $\beta = 1/k_B T$, with k_B being the Boltzmann constant and T being the temperature at which the transformation process is performed. Remarkably, this equation generates equilibrium properties from short, continuous, nonequilibrium trajectories that have little sensitivity to the quality of the reaction coordinate. Another appealing aspect of the Jarzynski equality is that the equilibration of the final state of the system is not required, as it would not change the measured nonequilibrium work.

The Jarzynski equality has been utilized in a broad range of applications. Procacci and Guarnieri²⁴ determined water–octanol partition coefficients for small molecules using the solvation free energies in both solvents. Binding free energies have been found using Jarzynski-based unidirectional estimators for small host–guest systems, such as those used in SAMPL6.²¹ Similarly, Steered Molecular Dynamics (SMD) has been used with more complex systems, such as the Translocator Protein (TSPO), in tandem with the Jarzynski equality to reconstruct potential of mean force (PMF) profiles

and determine ΔF_{off} for different ligands.²⁵ Xiong *et al.*²⁶ applied the Jarzynski equality in a number of contexts: calculating free energies with quantum-mechanical-molecular-mechanical (QM-MM) to study conversion reactions by the enzyme chorismate mutase; mechanical unfolding of the Ace–Alanine₈–NMe biopolymer; and creating free energy profiles for ligand diffusion in globin proteins.

Unfortunately, the trajectories that dominate the expectation value in Eq. (2) are those with low (or negative) values of W , which occur rarely and can be difficult to observe in simulation. These dominant trajectories in a forward process occur with a probability roughly approximated as

$$P_f \sim e^{-\beta \langle W_d^r \rangle}, \quad (3)$$

where W_d^r is the dissipative work of the reverse process, as estimated by Jarzynski.²⁷ This value is equal to $\frac{1}{N_c^r}$, where N_c^r is the number of realizations of processes necessary for convergence of free energy. This suggests that the number of realizations required for a process increases relative to the exponential of the average work and therefore the system size.^{27–29} In pulling experiments done to determine helix propensities for different amino acids, the Ala₁₂ peptide was found to require $\sim 10^5$ realizations for convergence of free energy for the forward process (and, thus, a $\sim 10^{-5}$ probability of a dominant realization occurring) and $\sim 10^7$ realizations for convergence of free energy for the reverse process.³⁰

Efforts have been made previously to apply path sampling algorithms to preferentially generate trajectories with lower values of work. One difficulty is that of the algorithms that have been developed to sample low probability events, only a small subset of these are applicable to nonequilibrium systems. Transition Path Sampling (TPS) methods^{31,32} have been applied to generate ensembles of trajectories with lower work values that contribute most significantly to the ensemble average.^{32–34} One challenge of this approach is that TPS is only able to select low-work trajectories after they have been generated and does not aim to accelerate their production. It is also difficult to sample new low-work trajectories as there are typically substantial free energy barriers that separate different low-work pathways. Other methods compatible with nonequilibrium systems—Nonequilibrium Umbrella Sampling (NEUS)^{35,36} and diffusion Monte Carlo³⁷—have also been applied to preferentially sample low-work trajectories for model systems.^{38,39} The weighted ensemble (WE) method^{14,40}—which utilizes a set of weighted, unbiased trajectories that can be cloned and merged—is another path sampling algorithm that is applicable to nonequilibrium systems and has been used to increase the chances of observing rare biomolecular conformations and to simulate long-timescale processes such as protein folding,⁴¹ large conformational transitions,⁴² and ligand (un)binding.^{43–45}

Here, we investigate the capability of the WE method to enhance the sampling of rare trajectories, specifically those with atypical values of work. Doing so requires a customized WE algorithm that considers the whole trajectory history during resampling. We adapt our recently developed algorithm called REVO (“Resampling Ensembles by Variation Optimization”)⁴⁶ for this purpose. We also implement a new resampling algorithm that explicitly considers the importance of individual trajectories and specifically focuses sampling on trajectories with the most significant work values. This

new algorithm includes a tunable parameter we call “amplification” that governs the depth to which the method will dig into the work distribution tail. We apply these methods to unbinding trajectories of a two-particle Lennard-Jones system with well depths ranging from 2 kcal/mol to 20 kcal/mol. The efficiency of these different resampling strategies is then analyzed for systems over a wide range of binding free energies. We also compare these results to the diffusion Monte Carlo approach, where the applied work is used as the selection potential.⁴⁷ Finally, we conclude with a discussion of this enhanced Jarzynski method and history based resampling, including future calculations of protein–ligand binding free energies.

II. METHODS

A. Generalized outline of weighted ensemble sampling

The framework for a generalized WE algorithm includes two main steps: the propagation of a group of trajectories (or “walkers”) forward in time by MD and resampling, which performs merging and cloning operations on the ensemble of walkers. Resampling operations function with the goal of cloning walkers with desirable features and merging together less-desirable walkers based on some feature of interest.

In a WE simulation, all walkers have a statistical weight or probability. When a walker is cloned, two independent walkers are created that have the conformation of the original cloned walker and half of its weight. In a merging operation, two walkers, A and B, are combined to create walker C with a weight of $w_C = w_A + w_B$ and C takes on the conformation of either A or B with a probability proportional to their weights. A resampling function takes in an ensemble of walkers and returns a new ensemble with conformations drawn from the original input ensemble. In general, the returned ensemble can have the same or a different number of walkers, but the sum of walker weights, typically equal to one, is unchanged. In this work, we maintain a constant number of walkers over the course of all simulations.

B. History-dependent REVO resampling

To perform work-based resampling simulations, the Resampling of Ensembles by Variation Optimization, or REVO⁴⁶ resampler is used. This method is briefly described below, with specific details regarding this application in Sec. II D. REVO governs cloning and merging through the maximization of an objective function called the “trajectory variation” (V), which is a scaled sum of all-to-all pairwise distances between the walkers,

$$V = \sum_i V_i = \sum_i \sum_j \left(\frac{d_{ij}}{d_0} \right)^\alpha \phi_i \phi_j, \quad (4)$$

where d_{ij} is the distance between two walkers i and j , α is an exponent that modifies the distances’ influence on the overall value of V , and d_0 is a “characteristic distance” that does not affect merging and cloning and serves to make V unitless. The distance metric is defined to capture the system-specific event of interest and is described

below. ϕ is a non-negative function that measures the relative importance of each walker and can again be designed in a system-specific fashion based on walker attributes, such as conformation, history, or weight. Here, ϕ is defined as

$$\phi_i = \log(w_i) - \log\left(\frac{p_{\min}}{100}\right), \quad (5)$$

where $p_{\min}$ is the minimum statistical weight that a walker can hold and w_i is the current weight of walker i . Setting a value of $p_{\min}$ is useful to avoid spending simulation time on trajectories that will not contribute meaningfully to the observables you wish to calculate. Similarly, a maximum value of the walker weight $p_{\max}$ is also useful to prevent the agglomeration of all of the weight into a single walker. Both $p_{\min}$ and $p_{\max}$ are enforced by simply preventing the resampling algorithm from suggesting at-risk walkers for cloning and merging, respectively. The overall goal of the REVO resampler is the optimization of V . To do this, walkers with a high V_i [Eq. (4)] are selected for cloning, and walkers with a low V_i are selected for merging.

While REVO is designed with high-dimensional systems in mind, here we found it sufficient to compute distance in a one-dimensional space, where d_{ij} is simply the difference in the sum of the time-elapsed work (as explained in Sec. II E) between a pair of walkers i and j . Notably, this distance metric relies on the entire history of a trajectory and not just the instantaneous conformation of the system. These history-dependent quantities, which we refer to as “activities,” can be utilized with the WEPY simulation package⁴⁸ and the WEPY-activity plugin.⁴⁹ The value of the activity, in this case, the work, for each walker is calculated after every cycle of dynamics. This value is then added to a running sum of work for that walker. The value of the running sum is then passed to a distance metric to do resampling.

C. Importance resampling

Multiple realizations of a process will result in the generation of a statistical distribution of work values $[p(W)]$. While most realizations of a process result in a “typical” work value near the peak of $p(W)$, $\langle e^{-\beta W} \rangle$ is dominated by realizations near the peak of $i(W) = p(W)e^{-\beta W}$ (Fig. 1). We refer to $i(W)$ as the “importance” of a specific work value, as it measures the extent to which that work value contributes to the ensemble average $\langle e^{-\beta W} \rangle$ and subsequently to ΔF using Eq. (2).

FIG. 1. Probability and importance curves. Most values of work observed are near the peak of $p(W)$ or around the mean of the work distribution, $\bar{W}$. Values of work that dominate $\langle e^{-\beta W} \rangle$ occur near the peak of $i(W)$, around $W^\dagger$.

With this in mind, we introduce a new trajectory ensemble resampling algorithm, “importance resampling,” for the WE framework that works with the WEPY simulation package. The overall goal of this process is to generate an ensemble with multiple trajectories that have weights at or around the peak of the importance curve. The importance resampler uses a new algorithm to select walkers for merging and cloning, which is comprised of two selection steps, one based on trajectory importance and another based on weight. Note that resampling based on selection (rather than cloning and merging) in a weighted ensemble context was described previously in Refs. 50 and 51. For importance-based selection, N_I trajectories are randomly selected with a probability proportional to their relative importance of that trajectory,

$$I_j = \frac{w_j e^{-\mu \beta W_j}}{\sum_k w_k e^{-\mu \beta W_k}}, \quad (6)$$

where I_j is the relative importance of trajectory j , w_i is the weight of trajectory i , and W_i is the total work applied to the system so far in trajectory i . Equation (6) also includes an additional factor called the “amplification” (μ). This can be set to any positive number to tune the strength with which we want to amplify the importance of low-work walkers. We examine different values of μ below. If a walker is selected once, it retains its weight for the next cycle. If a walker is selected more than once, its weight is divided evenly among the clones.

The remainder of the slots ($N_W = N - N_I$) are reserved for walkers that are selected based on walker weight. All walkers that were not chosen by importance are eligible for weight-based selection. Once selected, each walker takes on the probability,

$$p_W = \frac{1}{N_W} \left(1 - \sum_{i \in \mathbb{I}} w_i \right), \quad (7)$$

FIG. 2. Importance resampler schematic. Each walker is represented by a different colored circle. The walker weight is represented by circle size, and different colors indicate different conformations of the system. In resampling, importance based walkers are selected randomly with a probability proportional to their relative importance. The weight of walkers selected by their importance is equal to the weight of the parent walker divided by the number of times that walker was selected. The weight based walkers are selected randomly with a probability proportional to their weight. Each walker selected by weight is given an equal weight, given by Eq. (7). In this schematic, $N_I = 3$ and $N_W = 4$.

where $\mathbb{I}$ is the set of walkers that were selected based on their importance. Note that weight-based selection is necessary for the importance-based trajectories to maintain the proper relative walker weights. While, intuitively, low N_W values would lead to better sampling of the high-importance walkers, here we aim to maintain a number of weight-based walkers (e.g., $N_W = 5$) to ensure some diversity in the weight-based ensemble. A schematic of the importance resampler is shown in Fig. 2 for one round of resampling.

D. Lennard-Jones pair test system and simulation setup

We use a Lennard-Jones pair test system modified from the LennardJonesPair module of OpenMMTools⁵² to analyze the performance of both the REVO resampler and the importance resampler. We chose this system due to its simplicity and the fact that its difficulty can be easily tuned by changing ϵ , the depth of the inter-particle interaction energy well,

$$V_{LJ} = 4\epsilon \left(\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right). \quad (8)$$

The default values of particle mass and σ —39.9 Da and 3.35 Å, respectively—were used for all simulations with a box size set to 40 Å with the NonBondedForce set to CutOffPeriodic. ϵ was varied from 2 kcal/mol to 20 kcal/mol. For this system, the target free energy as a function of the inter-particle separation, r , can be analytically solved as

$$F_r = -2kT \ln(r) + V_{LJ}. \quad (9)$$

Using this system, nonequilibrium pulling simulations were performed without resampling [referred to as “straightforward (SF) MD”], with the REVO resampler, and with the importance resampler. All simulations were run with OpenMM⁵³ version 7.4.0 and OpenMMTools version 0.18.3. The system is run at a constant pressure of 1 atm and temperature of 300 K using Langevin dynamics with a friction coefficient of 1 ps^{−1} and an integration step size of 2 fs. For REVO simulations, a minimum and maximum walker weight ($p_{\min}$ and $p_{\max}$) of 10^{-100} and 0.5, respectively, were used with a maximum merging distance of 2.5 kJ/mol and $\alpha = 4$. For importance resampler simulations, 5 of 50 slots were designated for weight-based walkers ($N_W = 5$ and $N_I = 45$).

To perform the pulling simulations, a harmonic CustomBondForce from OpenMM is applied to the system. The force applied is

$$U(\vec{x}; r_0) = \frac{k}{2} (r(\vec{x}) - r_0)^2, \quad (10)$$

where k is a spring-constant with a constant value of 2000 kJ/mol/nm², r_0 is the target interatomic distance for that cycle with a starting value of 0.32 nm, and $r(\vec{x})$ is the interatomic distance calculated from a state vector $\vec{x}$. After every cycle of dynamics, the value of r_0 is updated for all trajectories by a predefined value, $\Delta r_0 = (r_0^f - r_0^i)/n_c$, where r_0^f is the final separation distance (2.0 nm) and n_c is the number of cycles, set to either 500 (examined first in Sec. III) or 53 (examined later).

Starting positions for each well depth are generated from straightforward simulations run at r_0^t . A set of 1000 starting positions was generated using the final positions of 1000 independent, 20 ns trajectories. These positions were saved and are used to randomly initiate pair starting positions for the nonequilibrium simulations.

E. The work equation and free energy surfaces

The nonequilibrium trajectories use a set of biasing potentials [Eq. (10)] to restrict the system to progressively increasing values of r_0 . The potential energy related to this biasing force at cycle t is denoted as $U(\vec{x}_t; r_0^t)$ and depends on both the current state of the system $\vec{x}_t$ (in this case used to calculate r , the observed interatomic distance) and the location of the biasing force r_0 at cycle t , which we denote as r_0^t .

Both the REVO resampler and the importance resampler require that work is calculated after every cycle by an activity metric and added to a running sum. For the REVO resampler, this total activity is then used for determining distances for the resampling process. For the importance resampler, the total activity is passed to resampler to determine importance values for each walker. The work observed along a trajectory (W_t) up to and including cycle t is calculated as follows:

$$W_t = \sum_{c=0}^t U(\vec{x}_c, r_0^{c+1}) - U(\vec{x}_c, r_0^c), \quad (11)$$

where the work from cycle c in the summand is equal to the resultant energy change from instantaneously changing the control parameter r_0 .⁵⁴

Using work values and walker weights, free energy surfaces can be generated by pulling simulations using a nonequilibrium adapted weighted histogram method.^{55,56} Here, we adapt Eq. (8) from Hummer and Szabo⁵⁵ to incorporate averages over weighted trajectory sets as

$$F_0(r) = -kT \ln \frac{\sum_t \frac{\langle \delta(r - r(\vec{x}_t)) e^{-\beta W_t} \rangle}{\langle e^{-\beta W_t} \rangle}}{\sum_t \frac{e^{-\beta U(r; r_0^t)}}{\langle e^{-\beta W_t} \rangle}}, \quad (12)$$

where r denotes a specific value of the interparticle separation. Here, $\langle \dots \rangle$ indicates an ensemble average for a specific timepoint, t , which for weighted ensemble simulations is calculated as

$$\langle f(\vec{x}_t) \rangle = \frac{1}{N_{\text{runs}}} \sum_{\text{runs}} \sum_{i=1}^N w_i f(\vec{x}_t^i), \quad (13)$$

where N is the number of walkers and $\vec{x}_t^i$ is the state vector of walker i at cycle t . The free energy (F_0) in Eq. (12) is calculated for each value of r and is used here to generate a free energy profile.

As this is a simple system, the target free energy profile for the Lennard-Jones pair can be solved for analytically using Eq. (9). To analyze the convergence of binding free energies, we define ΔF as the difference in free energy between the interatomic distances of 0.38 nm and 1.5 nm. We then track $\Delta\Delta F$ as a function of simulation time, which is the difference between the target ΔF determined analytically and the ΔF determined from simulation.

F. Diffusion Monte Carlo

An existing path sampling method, called diffusion Monte Carlo, is similar to the importance resampler in a number of ways. Both use an ensemble of trajectories (of size N_{traj}) that evolve according to unperturbed dynamics and are periodically resampled. Diffusion Monte Carlo, a quantum Monte Carlo method, uses a “selection potential” for each trajectory in an ensemble that can be determined from the applied work after each cycle of dynamics. Briefly, dynamics are run, the work increment for each trajectory is calculated, and a weighting factor for each trajectory is determined as $w_i = e^{-\beta W_{\text{inc}}}$, where W_{inc} is the incremental work for that trajectory. Note that these weights wholly determined by this work increment and are unrelated to the weights resulting from resampling in the weighted ensemble algorithm.

Following the determination of the weight, a resampling process is done by determining the branching numbers for each trajectory according to their weighting factors and choosing the resulting trajectories for the next cycle using a stochastic process, where the average number of times a trajectory is chosen is equal to $w_i / \sum_j w_j$.

For a given cycle of dynamics (say, c), the average weight is determined and saved as a variable, $z_c = \sum_i w_i / N_{\text{traj}}$. The z values calculated for an ensemble can be utilized for analysis with Eq. (12) through the substitution of the running product of these terms ($Z_c = \prod_{i=0}^c z_i$) for the existing $e^{-\beta W}$ term for each cycle. A full explanation of the algorithm implemented in this work can be found in Algorithm 6.3 of Ref. 47.

III. RESULTS

A. Reconstruction of free energy curves

We first ran a set of 200 independent simulations with 50 walkers each for both straightforward dynamics (with no resampling) and the REVO resampler. Separate sets were run for $\epsilon = 2$ kcal/mol, 5 kcal/mol, 10 kcal/mol, and 20 kcal/mol (or 8.37 kJ/mol, 20.9 kJ/mol, 41.8 kJ/mol, and 83.7 kJ/mol). All simulations are 500 cycles in length with 100 steps per cycle and were set up as described in Sec. II D. Figure 3 shows the free energy predictions as well as target free energy curves computed analytically [see Eq. (9)] for each value of ϵ . Both straightforward and REVO simulations accurately recreate the target free energy surfaces for all values of ϵ . Free energies are calculated from simulation using the Hummer and Szabo equation [Eq. (12)].

Figure 4(a) shows the probability distributions for the cumulative work applied to the system during the dissociation process. We observe that REVO more extensively samples the tails of the work distributions for all four values of ϵ . These also show how the most probable values of work depend on the depth of the free energy well. For $\epsilon = 20$, large values of applied work are necessary to perform dissociation, while for $\epsilon = 2$, many trajectories can be observed with low or even negative applied work. Importantly, we also observe that the probability distributions of applied work are not Gaussian, especially at higher values of ϵ . This indicates that direct sampling of low-work trajectories is important to obtaining accurate free energies with the Jarzynski equality.

As noted above, the importance of a given work value can be calculated as $i(W) = p(W) e^{-\beta W}$.²⁷ Importance distributions are

FIG. 3. Calculated free energy curves. Reconstructions of free energy surfaces are shown as a function of inter-particle separation (r) for (a) straightforward resampling and (b) REVO. Four different well-depths, $\epsilon = 2$ kcal/mol, 5 kcal/mol, 10 kcal/mol, and 20 kcal/mol, are shown. In each panel, the target free energy for each ϵ is shown as a thick line.

averaged over the entire trajectory set and shown in Fig. 4(b). As expected, peaks in $i(W)$ (green dashed lines) do not correspond to the peaks in $p(W)$ (blue solid lines). They are shifted to lower work values by a gap that increases with ϵ . For all values of ϵ , both methods are able to successfully sample the peak of the importance distributions; however, for $\epsilon = 20$, this peak lies at the limit of the distribution sampled by straightforward dynamics.

The REVO method is thus able to sample both the high and low tails of the work distribution more effectively than straightforward dynamics. To visualize how the ensemble of trajectories in REVO evolves in time, we show a “resampling tree,” which is a directed acyclic graph where the nodes represent walker states at each cycle and the edges show how walkers are cloned during REVO resampling (Fig. 5). Note that the bottom of the tree shows the initial state of the walkers and the time axis is in the upward direction. The highest weight walkers have a final value of work between 80 kJ/mol and 100 kJ/mol, which corresponds to the peak of the $\epsilon = 20$ probability curve in Fig. 4(a). Low weight walkers have the values of work that represent the high and low tails of the work distribution. This tree shows how REVO identifies and amplifies low-work walkers early on in the simulation, leading to better sampling of the final work distribution and more efficient calculation of binding free energies.

B. Convergence of free energy predictions

To analyze the relative efficiency of history-based REVO, we study the convergence of the free energy profile toward the target

FIG. 4. Probability distributions of applied work and trajectory importance. (a) The corresponding work distributions for three are shown for $\epsilon = 2$ kcal/mol, 5 kcal/mol, 10 kcal/mol, and 20 kcal/mol. Probability peaks are marked by blue dashed lines and are found at -4.02 kJ/mol, 9.14 kJ/mol, 29.12 kJ/mol, and 88.48 kJ/mol, respectively. (b) The probability distribution for the importance $[i(W) = p(W)e^{-\beta W}]$ is shown for $\epsilon = 2$ kcal/mol, 5 kcal/mol, 10 kcal/mol, and 20 kcal/mol. Importance peaks are marked by green dashed lines and are found at -5.6 kJ/mol, 5.3 kJ/mol, 24 kJ/mol, and 53 kJ/mol, respectively.

FIG. 5. A REVO resampling tree. A resampling tree for a single REVO simulation for $\epsilon = 20$. Node color and size correspond to work and walker weight, respectively. For clarity, we show here a 53 cycle simulation, which generally samples a wider distribution of work values than 500 cycle simulations shown in Fig. 4(a). This figure was made with Gephi.⁵⁷

profile as a function of simulation time. Rather than considering the whole free energy curve, we monitor the free energy difference, between two points, one for the bound state ($r = 0.38$ nm) and the other for the unbound state ($r = 1.5$ nm). We denote this as ΔF and refer to it as the unbinding free energy. $\Delta\Delta F$ is the difference between the unbinding free energy determined from simulation and the unbinding free energy for the analytically determined solution given by Eq. (9). Figure 6 shows the expectation values of the root mean squared $\Delta\Delta F$ and its uncertainty for both straightforward dynamics and REVO simulations.

For smaller numbers of trajectories, straightforward dynamics generally outperforms REVO, although this difference is not statistically significant for $\epsilon = 20$. The final root mean square errors (RMSEs) for 8000 trajectories are roughly equivalent for $\epsilon = 2$ and 5. For a well depth of 10 kcal/mol, REVO performs significantly better than straightforward dynamics after 400 trajectories.

FIG. 6. Error vs time. The convergence of the binding free energy for $\epsilon =$ (a) 2 kcal/mol, (b) 5 kcal/mol, (c) 10 kcal/mol, and (d) 20 kcal/mol for straightforward (SF) and REVO as a function of the total number of walkers used to calculate each free energy value. For REVO simulations, an ensemble of 50 was used, and multiple simulations are combined to reach the number of walkers shown. The shaded areas show the uncertainty calculated at each point using standard error of the mean over 15 trajectory sets.

For $\epsilon = 20$, higher error is observed for both methods throughout, although REVO outperforms SF at long times, obtaining a final RMSE (0.49 kJ/mol) that is roughly half of the final value obtained with straightforward dynamics (0.78 kJ/mol).

C. Importance resampler results

Although REVO is able to broadly enhance sampling of the work distribution, we investigate whether a method that specifically amplifies sampling of walkers at the peak of the importance distribution would more efficiently calculate ΔF . The importance resampler is described in Sec. II C and uses a parameter, μ , to govern how deeply it samples low-work trajectories. Five sets of 200 independent, 50 walker simulations were run for multiple values of μ for each ϵ analyzed above. Datasets were generated for simulations with both 53 cycles and 500 cycles and setup as described in Sec. II D. We found that values of $\mu < 1$ perform best, with the RMSE gradually increasing with μ for simulations with 53 cycles. Errors were consistently within a small range for increasing values of μ for 500 cycle simulations (Fig. S1). However, these errors were, in general, larger than those observed for both SF and REVO for similar numbers of trajectories.

There is variability across different sets of trajectories obtained with the same value of μ as shown in Fig. S2 for $\mu = 0.1$. This is true

FIG. 7. Comparison of two importance sampling trajectory sets with $\mu = 0.4$. The (a) free energy surfaces, (b) work distributions, and (c) importance curves for two sets of 53 cycle simulations.

even for very large trajectory sets. We find that this is due to differences in the sampling of the work distribution as shown for two different, 10 000-trajectory sets in Fig. 7. In Fig. 7(a), the free energy surfaces for two sets of data for $\varepsilon = 20$ with $\mu = 0.4$ are compared, with one set outperforming the other with a lower overall RMSE. The work distributions for these datasets [Fig. 7(b)] show differences in work values below ~ 65 kJ/mol, and it is seen in Fig. 7(c) that these low-work values have very high importance [consistent

with the results in Fig. 4(b)]. The dataset with higher resolution in the low-work tail (blue) has lower RMSE in the free energy surface.

For 500 cycle simulations, importance resampling performed the worst among the three resampling strategies examined here for all the values of ε . However, we wanted to investigate whether it could be useful for shorter simulations, in systems where generating sufficiently long trajectories is computationally demanding. Figures S2 and S3 analyze the efficiency of the convergence of

FIG. 8. Work and importance curves for 53 and 500 cycle simulations. The work curves for (a) 53 and (b) 500 cycle simulations are shown for REVO, SF, and importance resampling with three different amplification factors (μ). The corresponding importance curves are shown in (c) and (d).

the error of free energy profile ($\Delta\Delta F$) in comparison to straightforward dynamics and REVO for 53 cycle simulations. Broadly, importance resampling performs with comparable efficiency to REVO and straightforward using 53 cycle simulations, although it performs poorly for $\varepsilon = 20$ kcal/mol. The individual values of μ were found to outperform both SF and REVO simulations for each value of ε ; however, the performance was inconsistent between different independent sets of 200 simulations with the same value of μ as shown in Fig. S2 for $\mu = 0.1$.

To show the differences in work distributions sampled by all methods as well as the effect of μ in the importance resampler, work distributions and corresponding importance curves for 53 and 500 cycle simulations are shown in Fig. 8. Each curve shown is an average over 1000 trajectories each. In the 53 cycle simulations, the importance resampler can more thoroughly sample the tails of the work distribution, with a depth that can be tuned by μ . This results in a bimodal distribution of work values with one group of low-work, low-probability trajectories, and another group of average-work, high-probability trajectories. As μ increases, the separation between the two halves of the bimodal distribution increases with peaks moving further into both the low and high work values. However, for 500 cycle simulations, the importance resampler only samples a limited

range of work values, similar to that of straightforward dynamics. For longer simulations, the peak of the importance curve is not reached for all the values of μ .

To further visualize μ 's effects on resampling, Fig. 9 shows the resampling trees for $\mu = 0.5$ and 5.0. For low values of μ , nearly all walkers have high final values of work. In this case, the majority of these end walkers originate from a single early cloning event of a high work walker roughly 1/3 of the way through the simulation. At this point, the weight of the low-work walkers decreased enough that the high-weight, high-work walkers had a larger $i(W)$ value. This does not accomplish the original goals of importance-based resampling and results in final distributions that are similar to those obtained with straightforward resampling.

For high values of μ , there is a stark separation of the weight-based and importance-based walkers beginning early on in the simulation. The weight-based walkers maintain high values of work throughout the remainder of the simulation, whereas the value of work for the importance-based walkers increases and then decreases over time. For high μ , many cloning events also occur en masse from single walkers in some cycles. High μ , 500 cycle simulations have very limited sampling of the work distribution, unlike 53 cycle simulations. Although resampling patterns for these simulations

FIG. 9. Resampling trees for a low and high value of μ . Resampling trees for $\mu =$ (a) 0.5 and (b) 5 for $\varepsilon = 20$ simulations with 53 cycles. Node color and size correspond to work and weight, respectively.

FIG. 10. Error vs time for REVO and diffusion Monte Carlo. The convergence of the binding free energy for $\epsilon = 20$ kcal/mol for straightforward (SF), REVO, and diffusion Monte Carlo (DIFFMC) as a function of the total number of trajectories used to calculate each free energy value. For REVO and DIFFMC simulations, an ensemble size of 50 was used, and multiple simulations are combined to reach the number of trajectories shown. The shaded areas show the uncertainty calculated at each point using the standard error of the mean over 15 independent trajectory sets.

initially resemble their 53 cycle counterparts, eventually many cloning events from the high-weight, high-work walkers overwhelm the low-work walkers. This results in final work distributions that resemble straightforward data [Fig. 8(b)] and again is not able to accomplish the goals of importance-based resampling.

D. Comparison with diffusion Monte Carlo

To analyze the relative efficiency of REVO in comparison to diffusion Monte Carlo, we study convergence toward the target free energy profile, as shown in Fig. 6. For both methods, 800 simulations with 50 walkers each are run with $\epsilon = 20$ kcal/mol. All simulations are 500 cycles in length with 100 steps per cycle. REVO simulations were run as described in Sec. II D, and diffusion Monte Carlo simulations were run in a similar fashion, but using the diffusion Monte Carlo resampling algorithm and free energy calculation strategy described in Sec. II F. Resampling was done every cycle, which corresponds to a relative entropy cutoff of 0.

As shown in Fig. 10, Diffusion Monte Carlo out-performs both straightforward dynamics and REVO for low numbers of trajectories. While diffusion Monte Carlo performs well for smaller datasets, the rate of RMSE decrease appears to slow for larger trajectory sets, whereas REVO and straightforward dynamics continue to gradually decrease. After this point, diffusion Monte Carlo and straightforward dynamics obtain similar final values of 0.77 ± 0.08 kJ/mol and 0.78 ± 0.06 kJ/mol, respectively, with REVO obtaining a final value of 0.51 ± 0.10 kJ/mol.

IV. DISCUSSION

The Jarzynski equality generally requires a very large number of realizations of a process to converge the free energy due to the low probability of observing high-importance work values. Due to the potentially large computational cost, this is a major limiting factor in the use of the nonequilibrium work theorem for calculating free energies. The results above demonstrate that a thorough sampling of the tails of work distributions can increase the accuracy of free energy calculations, potentially unlocking the promise of the Jarzynski equality. This can be achieved through the use of a

promising combination of methods—weighted ensemble rare-event algorithms with short duration nonequilibrium simulations—that efficiently generates a work distribution with low-probability tails. A history-dependent REVO resampler generally outperformed direct straightforward sampling of unbinding trajectories especially for dissociation events with large free energy barriers.

We also examined a novel importance-based resampler that showed sporadic success using short trajectories, but generally failed to work as designed. The efficiency of the Lennard-Jones pair allowed us to run many simulations at increasing amplification factors for the importance resampler. While the inclusion of the amplification factor allowed for further exploration into the tails of the work distribution, it presented a puzzling result. As amplifications increased beyond ~ 1 , the accuracy of free energy calculations began to decrease for shorter simulations (53 cycles in duration). As shown in Fig. 9(a), although the importance resampler is able to sample trajectories with extremely low work values, in practice, it underestimates the probability of these trajectories. This occurs at higher amplifications because the resampler picks one or two high-importance trajectories each round to be cloned repeatedly, although these often have a change in work that results in a lower importance in the next cycle. In contrast, the REVO algorithm only seeks to diversify the work values at each cycle, picking not only the highest importance walkers but a more diverse group. This results in a more thorough sampling of the whole work distribution and indicates that the REVO resampler will be more efficient to work with and develop further moving forward.

REVO is also compared to an existing diffusion Monte Carlo algorithm and was found to show smaller errors in free energy for large ensembles of trajectories for the $\epsilon = 20$ kcal/mol system. In these calculations, for comparative purposes, we used the same parameters between the REVO, importance resampler and diffusion Monte Carlo algorithms, whenever possible. In theory, a more exhaustive search of parameter space, such as the use of a nonzero entropy trigger may improve the results obtainable by diffusion Monte Carlo. Larger trajectory sets may also be beneficial for improving results, as it has been shown that the (intrinsic) RMSE of diffusion Monte Carlo goes to zero as the number of trajectories goes to infinity.⁴⁷ It could be surprising that REVO, which enhances the sampling of the entire work distribution, including higher-work trajectories, would outperform an algorithm that explicitly focuses only on those trajectories that are most important for the calculation of the work distribution. However, like the importance resampler, diffusion Monte Carlo is making predictions of the overall importance of a trajectory based on incomplete work histories. These results imply that maintaining a set of trajectories with diverse work values can result in better sampling of low work trajectories at the end of the simulation. We expect this behavior to be even more important when moving to protein–ligand systems with more orthogonal degrees of freedom.

We note that there is an interesting connection between the amplification factor used in the importance resampler and large deviation theory in which paths are re-weighted according to a factor e^{-sA} , where A is an activity measured along a path and s is the strength of an applied field.⁵⁸ This has been used previously to study dynamical phase transitions, where the $\langle A \rangle$ or its derivative shows a discontinuity as a function of s that give insight into phenomena

such as glass transitions,^{59,60} synchronization to external fields,⁶¹ and kinetic trapping during protein folding.⁶² Methods such as transition path sampling can apply the e^{-sA} field directly in their path acceptance step and sample paths within a given s -ensemble. In contrast, in the importance resampler, only a subset of the weighted ensembles are chosen according to $I(W)$. This said, the possibility of a dynamical phase transition of $\langle\beta W\rangle$ in response to μ is an intriguing direction for future work.

To further improve upon the promising results of the REVO resampler for work calculations, new distance metrics can be created that focus on increasing the cloning of the low work values exclusively. Based on these results, we also anticipate that traditional weighted ensemble binning of the work distribution would also perform well. However, unlike REVO, suitable bins would need to define along the work axis, which would be different for each system and would be difficult to predict beforehand. In addition, sampling efficiency and accuracy could be improved by performing simulations in both the forward and reverse directions.⁶³ The potential for REVO to accelerate free energy calculations within a bi-directional framework will be studied in future work.

Here, we use the notion of a trajectory “activity,” which is a history-dependent observable associated with each trajectory. This is implemented with the WEPY software package and can be written to be system specific and work with either a distance metric (e.g., in the REVO algorithm) or directly with a specific resampler (e.g., in the importance resampler). The utilization of activities in weighted ensemble resampling could find use in systems looking at mobility. For instance, glass transitions could be studied using an activity measuring mean squared displacement, and ion transport could be studied using an activity measuring ion flux through a pore. Another example of a trajectory activity is the classical action, which could be used to calculate the probabilities of trajectories.⁶⁴

In this work, we examine a very simple two-atom system with umbrella potentials defined by the inter-particle separation. For larger, more complex systems (protein-peptide and protein-ligand), there can be many possible (un)binding pathways.^{65,66} While general reaction coordinates, such as the inter-particle separation, can be used to describe the unbinding process, the efficiency of generating low-work trajectories will likely depend on the proper choice of reaction coordinate. Fortunately, the combinations of methods presented here offer many opportunities for modification. For instance, the functional form of the pulling force could be changed, such as making it a double-well potential that gradually destabilizes the bound state. The REVO resampling algorithm could also take into account distances between unbinding trajectories to promote a diversity of transition paths as well as work values. Alternatively, instead of restraint potentials, small, randomly oriented forces can be applied to an unbinding ligand, such as in the random acceleration molecular dynamics (RAMD) method.⁶⁷ This could allow for efficient free energy calculations while requiring no previous knowledge of the unbinding pathway.

SUPPLEMENTARY MATERIAL

See the [supplementary material](#) for Figs. S1–S3.

ACKNOWLEDGMENTS

The authors acknowledge Nazanin Donyapour for helpful discussions regarding the experimental design. A.R.D. acknowledges support from the National Institutes of Health (Award No. R01GM130794).

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon reasonable request. The computer code used to generate these results is freely available on GitHub. See <https://github.com/ADicksonLab/wepy> for the WEPY package and <https://github.com/ADicksonLab/wepy-activity> for the WEPY-activity plugin.

REFERENCES

- 1 A. Pohorille, C. Jarzynski, and C. Chipot, *J. Phys. Chem. B* **114**, 10235 (2010).
- 2 C. D. Christ, A. E. Mark, and W. F. van Gunsteren, *J. Comput. Chem.* **31**, 1569 (2010).
- 3 Z. Cournia, B. Allen, and W. Sherman, *J. Chem. Inf. Model.* **57**, 2911 (2017).
- 4 L. Wang, J. Chambers, and R. Abel, *Biomol. Simul.* **2022**, 201 (2019).
- 5 W. L. Jorgensen and L. L. Thomas, *J. Chem. Theory Comput.* **4**, 869 (2008).
- 6 G. M. Torrie and J. P. Valleau, *J. Comput. Phys.* **23**, 187 (1977).
- 7 N. Nishikawa, K. Han, X. Wu, F. Tofoleanu, and B. R. Brooks, *J. Comput.-Aided Mol. Des.* **32**, 1075 (2018).
- 8 N. Singhal, C. D. Snow, and V. S. Pande, *J. Chem. Phys.* **121**, 415 (2004).
- 9 S. Gu, D.-A. Silva, L. Meng, A. Yue, and X. Huang, *PLoS Comput. Biol.* **10**, e1003767 (2014).
- 10 A. Laio and M. Parrinello, *Proc. Natl. Acad. Sci. U. S. A.* **99**, 12562 (2002).
- 11 P. Tiwary, J. Mondal, and B. J. Berne, *Sci. Adv.* **3**, e1700014 (2017).
- 12 A. K. Faradjian and R. Elber, *J. Chem. Phys.* **120**, 10880 (2004).
- 13 L. W. Votapka, B. R. Jagger, A. L. Heyneman, and R. E. Amaro, *J. Phys. Chem. B* **121**, 3597 (2017).
- 14 G. A. Huber and S. Kim, *Biophys. J.* **70**, 97 (1996).
- 15 A. Dickson, *Biophys. J.* **115**, 1707 (2018).
- 16 M. Aldeghi, J. P. Bluck, and P. C. Biggin, “Absolute alchemical free energy calculations for ligand binding: A beginner’s guide,” in *Computational Drug Discovery and Design*, edited by M. Gore and U. B. Jagtap (Springer New York, New York, NY, 2018), pp. 199–232.
- 17 M. K. Gilson, J. A. Given, B. L. Bush, and J. A. McCammon, *Biophys. J.* **72**, 1047 (1997).
- 18 J. G. Kirkwood, *J. Chem. Phys.* **3**, 300 (1935).
- 19 Y. Sugita, A. Kitao, and Y. Okamoto, *J. Chem. Phys.* **113**, 6042 (2000).
- 20 A. Rizzi, S. Murkli, J. N. McNeill, W. Yao, M. Sullivan, M. K. Gilson, M. W. Chiu, L. Isaacs, B. C. Gibb, D. L. Mobley, and J. D. Chodera, *J. Comput.-Aided Mol. Des.* **32**, 937 (2018).
- 21 A. Rizzi, T. Jensen, D. R. Slochow, M. Aldeghi, V. Gapsys, D. Ntekoimes, S. Bosisio, M. Papadourakis, N. M. Henriksen, B. L. de Groot, Z. Cournia, A. Dickson, J. Michel, M. K. Gilson, M. R. Shirts, D. L. Mobley, and J. D. Chodera, *J. Comput.-Aided Mol. Des.* **34**, 601 (2020).
- 22 C. Dellago and G. Hummer, *Entropy* **16**, 41 (2014).
- 23 C. Jarzynski, *Phys. Rev. Lett.* **78**, 2690 (1997).
- 24 P. Procacci and G. Guarnieri, *J. Comp.-Aided Drug Des.* **34**, 371 (2020).
- 25 A. Bruno, E. Barresi, N. Simola, E. Da Pozzo, B. Costa, E. Novellino, F. Da Settimo, C. Martini, S. Taliani, and S. Cosconati, *ACS Chem. Neurosci.* **10**, 3805 (2019).
- 26 H. Xiong, A. Crespo, M. Marti, D. Estrin, and A. E. Roitberg, *Theor. Chem. Acc.* **116**, 338 (2006).
- 27 C. Jarzynski, *Phys. Rev. E* **73**, 046105 (2006).

- ²⁸R. C. Lua and A. Y. Grosberg, *J. Phys. Chem. B* **109**, 6805 (2005).
- ²⁹J. Gore, F. Ritort, and C. Bustamante, *Proc. Natl. Acad. Sci. U. S. A.* **100**, 12564 (2003).
- ³⁰I. Echeverria and L. M. Amzel, *Proteins: Struct., Funct., Bioinf.* **78**, 1302 (2010).
- ³¹P. G. Bolhuis, D. Chandler, C. Dellago, and P. L. Geissler, *Annu. Rev. Phys. Chem.* **53**, 291 (2002).
- ³²C. Dellago, G. Bolhuis, and L. P. Geissler, *Computer Simulations in Condensed Matter Systems: From Materials to Chemical Biology* (Physica-Verlag, 2006), Vol. 1, p. 349.
- ³³S. X. Sun, *J. Chem. Phys.* **118**, 5769 (2003).
- ³⁴F. M. Ytreberg and D. M. Zuckerman, *J. Chem. Phys.* **120**, 10876 (2004).
- ³⁵A. Warmflash, P. Bhimalapuram, and A. R. Dinner, *J. Chem. Phys.* **127**, 154112 (2007).
- ³⁶A. Dickson, A. Warmflash, and A. R. Dinner, *J. Chem. Phys.* **130**, 074104 (2009).
- ³⁷R. Assaraf, M. Caffarel, and A. Khelif, *Phys. Rev. E* **61**, 4566 (2000).
- ³⁸A. R. Dinner, J. C. Mattingly, J. O. B. Tempkin, B. V. Koten, and J. Weare, *SIAM Rev.* **60**, 909 (2018).
- ³⁹M. Rousset and G. Stoltz, *J. Stat. Phys.* **123**, 1251 (2006); [arXiv:0511412](#) [cond-mat].
- ⁴⁰D. M. Zuckerman and L. T. Chong, *Annu. Rev. Biophys.* **46**, 43 (2017).
- ⁴¹B. Abdul-Wahid, H. Feng, D. Rajan, R. Costaouec, E. Darve, D. Thain, and J. A. Izaguirre, *J. Chem. Inf. Model.* **54**, 3033 (2014).
- ⁴²B. W. Zhang, D. Jasnow, and D. M. Zuckerman, *Proc. Natl. Acad. Sci. U. S. A.* **104**, 18043 (2007).
- ⁴³M. C. Zwier, A. J. Pratt, J. L. Adelman, J. W. Kaus, D. M. Zuckerman, and L. T. Chong, *J. Phys. Chem. Lett.* **7**, 3440 (2016).
- ⁴⁴S. D. Lotz and A. Dickson, *J. Am. Chem. Soc.* **140**, 618 (2018).
- ⁴⁵U. Adhikari, B. Mostofian, J. Copperman, S. R. Subramanian, A. A. Petersen, and D. M. Zuckerman, *J. Am. Chem. Soc.* **141**, 6519 (2019).
- ⁴⁶N. Donyapour, N. M. Roussey, and A. Dickson, *J. Chem. Phys.* **150**, 244112 (2019).
- ⁴⁷T. Lelièvre, M. Rousset, and G. Stoltz, *Free Energy Computations: A Mathematical Perspective* (Imperial College Press, 2010).
- ⁴⁸See <https://github.com/ADicksonLab/wepy> for information about the Weighted Ensemble Python (WEPY) package.
- ⁴⁹See <https://github.com/ADicksonLab/wepy-activity> for the WEPY-activity code.
- ⁵⁰D. Aristoff and D. M. Zuckerman, *Multiscale Model. Simul.* **18**, 646 (2018).
- ⁵¹D. Aristoff, [arXiv:1906.00856](#) [math.NA] (2020).
- ⁵²See <https://openmmtools.readthedocs.io/en/0.18.1/> for information about OpenMMTools.
- ⁵³P. Eastman, J. Swails, J. D. Chodera, R. T. McGibbon, Y. Zhao, K. A. Beauchamp, L.-P. Wang, A. C. Simmonett, M. P. Harrigan, C. D. Stern, R. P. Wiewiora, B. R. Brooks, and V. S. Pande, *PLoS Comput. Biol.* **13**, e1005659 (2017).
- ⁵⁴G. E. Crooks, *J. Stat. Phys.* **90**, 1481 (1998).
- ⁵⁵G. Hummer and A. Szabo, *Proc. Natl. Acad. Sci. U. S. A.* **98**, 3658 (2001).
- ⁵⁶A. M. Ferrenberg and R. H. Swendsen, *Phys. Rev. Lett.* **63**, 1195 (1989).
- ⁵⁷M. Bastian, S. Heymann, and M. Jacomy, "Gephi: An open source software for exploring and manipulating networks," in *International AAAI Conference on Weblogs and Social Media* (AAAI, 2009).
- ⁵⁸R. M. Turner, T. Speck, and J. P. Garrahan, *J. Stat. Mech.: Theory Exp.* **2014**, P09017.
- ⁵⁹L. O. Hedges, R. L. Jack, J. P. Garrahan, and D. Chandler, *Science* **323**, 1309 (2009).
- ⁶⁰J. P. Garrahan, R. L. Jack, V. Lecomte, E. Pitard, K. Van Duijvendijk, and F. Van Wijland, *J. Phys. A: Math. Theor.* **42**, 075007 (2010).
- ⁶¹A. Dickson, S. M. A. Tabei, and A. R. Dinner, *Phys. Rev. E* **84**, 061134 (2011).
- ⁶²A. S. Mey, P. L. Geissler, and J. P. Garrahan, *Phys. Rev. E* **89**, 032109 (2014); [arXiv:1305.5748](#).
- ⁶³M. R. Shirts, E. Bair, G. Hooker, and V. S. Pande, *Phys. Rev. Lett.* **91**, 140601 (2003).
- ⁶⁴R. P. Feynman, *Rev. Mod. Phys.* **20**, 367 (1948).
- ⁶⁵A. Dickson and S. D. Lotz, *Biophys. J.* **112**, 620 (2017).
- ⁶⁶R. Capelli, P. Carloni, and M. Parrinello, *J. Phys. Chem. Lett.* **10**, 3495 (2019); [arXiv:1904.10726](#).
- ⁶⁷D. B. Kokh, M. Amaral, J. Bomke, U. Grädler, D. Musil, H.-P. Buchstaller, M. K. Dreyer, M. Frech, M. Lowinski, F. Vallee, M. Bianciotto, A. Rak, and R. C. Wade, *J. Chem. Theory Comput.* **14**, 3859 (2018).