Reinforced dynamics for enhanced sampling in large atomic and molecular systems

Linfeng Zhang (iD), Han Wang (iD), and Weinan E

View Online

Export Citation

CrossMark

ARTICLES YOU MAY BE INTERESTED IN

Learning free energy landscapes using artificial neural networks
The Journal of Chemical Physics **148**, 104111 (2018); https://doi.org/10.1063/1.5018708

Adaptive enhanced sampling by force-biasing using neural networks
The Journal of Chemical Physics **148**, 134108 (2018); https://doi.org/10.1063/1.5020733

Time-lagged autoencoders: Deep learning of slow collective variables for molecular kinetics
The Journal of Chemical Physics **148**, 241703 (2018); https://doi.org/10.1063/1.5011399

AIP
Publishing

Reinforced dynamics for enhanced sampling in large atomic and molecular systems

Linfeng Zhang,[1] Han Wang,[2,a] and Weinan E[3,b]

[1]*Program in Applied and Computational Mathematics, Princeton University, Princeton, New Jersey 08544, USA*
[2]*Institute of Applied Physics and Computational Mathematics, Fenghao East Road 2, Beijing 100094, People's Republic of China and CAEP Software Center for High Performance Numerical Simulation, Huayuan Road 6, Beijing 100088, People's Republic of China*
[3]*Department of Mathematics and Program in Applied and Computational Mathematics, Princeton University, Princeton, New Jersey 08544, USA and Beijing Institute of Big Data Research, Beijing 100871, People's Republic of China*

A new approach for efficiently exploring the configuration space and computing the free energy of large atomic and molecular systems is proposed, motivated by an analogy with reinforcement learning. There are two major components in this new approach. Like metadynamics, it allows for an efficient exploration of the configuration space by adding an adaptively computed biasing potential to the original dynamics. Like deep reinforcement learning, this biasing potential is trained on the fly using deep neural networks, with data collected judiciously from the exploration and an uncertainty indicator from the neural network model playing the role of the reward function. Parameterization using neural networks makes it feasible to handle cases with a large set of collective variables. This has the potential advantage that selecting precisely the right set of collective variables has now become less critical for capturing the structural transformations of the system. The method is illustrated by studying the full-atom explicit solvent models of alanine dipeptide and tripeptide, as well as the system of a polyalanine-10 molecule with 20 collective variables. *Published by AIP Publishing.*
https://doi.org/10.1063/1.5019675

I. INTRODUCTION

Exploring the configuration space of large atomic and molecular systems is a problem of fundamental importance for many applications, including protein folding, materials design, and understanding chemical reactions. There are several difficulties associated with these applications. The first is that the dimensionality of the configuration space is typically very high. The second is that there are often high energy barriers associated with the exploration. Both difficulties can be reduced by the introduction of collective variables (CVs) and the mapping of the problem to the CV space. The problem then becomes finding the free energy surface (FES) associated with the set of CVs, a problem that has attracted a great deal of interest in the last few decades.[1–14] One of the most effective techniques is metadynamics,[9] which computes a biasing potential by depositing Gaussian bases along the trajectory in the CV space. It is shown that the biasing potential converges to the inverted free energy at the end of the calculation.[10] Also closely related to our work are the recent papers that propose to use machine learning methods to help parameterizing FES.[15–18] In particular, the deep neural network (DNN) model has shown promise in effectively representing the FES defined on high dimensional CV space.[17,18]

In this work, we take metadynamics and machine learning methods one step further by making an analogy between reinforcement learning[19] and the task of configuration space exploration and FES calculation. The classical reinforcement learning scheme involves a state space, an action space, and a reward function. The objective is to find the best policy function, which is a mapping from the state space to the action space, that optimizes the cumulative reward function. Our problem can be thought of as being a multi-scale reinforcement learning problem. We have a micro-state space, the configuration space of the detailed atomic system, and a macro-state space, the space of the CVs. The action space will be represented by the biasing potential in the biased molecular dynamics (MD) on the micro-state space. The optimal policy function is the inverted FES, defined on the macro-state space. The FES is parameterized by a carefully designed DNN model. Among other things, this allows us to handle cases with a large set of CVs. In the absence of an explicit reward function, we introduce an uncertainty indicator that can be used to quantify the accuracy of the FES representation. It is defined as the standard deviation of the predictions from an ensemble of DNN models, which are trained using the same dataset but different initialization of the model parameters. The bias is only adopted in regions where the uncertainty indicator is low, i.e., regions that are sufficiently explored, and thus the exploration in the insufficiently explored region is encouraged. We call this scheme the "reinforced dynamics," to signal its analogy with reinforcement learning.

a)Electronic mail: wang_han@iapcm.ac.cn
b)Electronic mail: weinan@math.princeton.edu

Roughly speaking, reinforced dynamics works as follows: The biasing potential, or the action, is initialized at 0 and is expected to converge to the inverted FES as the dynamics proceeds. Each step of the macro-iteration involves the following components. First, a biased MD is performed, in which the system is biased only in the regions where the uncertainty indicator is low. The biased simulation is likely to visit the CV regions never visited before or where the FES representation quality is poor. Next, a certain number of the newly visited CV values in regions where the uncertainty indicator is high are added to the training dataset. A restrained MD is performed to obtain the mean force, or the negative gradient of the FES, at each of the newly added CV values. Finally, the accumulated CV values and the mean forces are used as labels to train several network models, which give the current estimate of the biasing potential and the uncertainty indicator. This process is repeated iteratively until convergence is achieved, when the newly visited CV values all fall in the regions where the uncertainty indicator is low.

The quality of the free energy surface is determined by the quality of the CVs. Ideally we would like the FES to capture the structural and dynamic information of the system, such as the important metastable states and transitions between the metastable states. For many years, since our ability to accurately approximate the FES has been limited to systems with a small number of CVs, we have always faced the dilemma that choosing the right CVs is both critical and practically impossible. We believe that the ability of the reinforced dynamics to handle a large set of CVs will make the issue of choosing the right CVs much less critical.

In this paper, we give a systematic presentation of the theoretical and practical aspects of reinforced dynamics. We first focus on methodology and introduce the theory and flowchart of the reinforced dynamics scheme. Then we use the classical example of alanine dipeptide and tripeptide with two and four CVs, respectively, as illustrations due to their intuitive appeal. The solvent effect is explicitly considered in both examples. The FESs constructed by the reinforced dynamics are compared with those constructed by long brute-force simulations (5.1 μs for alanine dipeptide and 47.7 μs for tripeptide) to demonstrate the accuracy and efficiency of the method. Finally, an application to the structural optimization of the polyalanine-10 system with 20 CVs is presented to demonstrate the practical promise of reinforced dynamics.

II. THEORY

A. Free energy and mean forces

We assume that the system we are studying has N atoms, with their positions denoted by $r = (r_1, \ldots, r_N)$. The potential energy of the system is denoted by $U(r)$. Without loss of generality, we consider the system in a canonical ensemble. Given M predefined CVs, denoted by $s(r) = (s_1(r), \ldots, s_M(r))$, the free energy defined on the CV space is

$$A(s) = -\frac{1}{\beta} \ln p(s), \quad p(s) = \frac{1}{Z} \int e^{-\beta U(r)} \delta(s(r) - s)\, dr, \quad (1)$$

with $Z = \int e^{-\beta U(r)} dr$ being the normalization factor. The brute-force way of computing the free energy (1) is to sample the CV space exhaustively and to approximate the probability distribution $p(s)$ by making a histogram of the CVs. This approach may easily become prohibitively expensive. In such a case, an alternative way of constructing the FES is to fit the mean forces acting on the CVs, i.e.,

$$F(s) = -\nabla_s A(s). \quad (2)$$

Several ways of computing $F(s)$ have been proposed.[11,13,20] We will adopt the approach of restrained dynamics proposed in Ref. 11. In this formulation, a new term is added to the potential of the system to represent the effect of the spring forces between the configuration variables and the CVs. It can be shown that the mean force is given by $F_\alpha(s) = \lim_{k_\alpha \to \infty} F_\alpha^k(s)$ for $\alpha = 1, 2, \ldots, M$, where the αth component of F^k is defined to be

$$F_\alpha^k(s) = \frac{1}{Z_k(s)} \int k_\alpha(s_\alpha(r) - s_\alpha)\, e^{-\beta U_k(r,s)}\, dr. \quad (3)$$

Here $Z_k(s) = \int e^{-\beta U_k(r,s)}\, dr$ is the normalization factor, $\{k_\alpha | \alpha = 1, \ldots, M\}$ are the spring constants for the harmonic restraining potentials, and $U_k(r, s)$ is defined by

$$U_k(r, s) = U(r) + \sum_{\alpha=1}^{M} \frac{1}{2} k_\alpha(s_\alpha(r) - s_\alpha)^2. \quad (4)$$

In practice, the spring constants are chosen to be large enough to guarantee the convergence to the mean forces. The time duration for the restrained dynamics should be longer than the largest relaxation time scale of the fast modes of the system, in order for the ensemble average in Eq. (3) to be approximated adequately by the time average. In the rest of the paper, we do not explicitly distinguish F and F^k.

B. Free energy representation

The free energy $A(s)$ will be represented by a deep neural network (DNN) model, in which the input CVs are first preprocessed, then passed through multiple fully connected hidden layers, and, in the end, mapped to the free energy. The structure of the DNN model is schematically illustrated in Fig. 1. Mathematically, a DNN representation with N_h hidden layers is given by

$$\mathcal{A}(s) = \mathcal{L}^{\text{out}} \circ \mathcal{L}^{N_h} \circ \cdots \circ \mathcal{L}^1 \circ \mathcal{P}(s), \quad (5)$$

where "$\circ$" denotes the function composition. The differentiable operator $\mathcal{P}$ represents the system-dependent preprocessing procedure for the CVs, which will be illustrated by the examples in Sec. III. For the pth hidden layer, which has M_p neurons $d_p \in \mathbb{R}^{M_p}$, $\mathcal{L}_p$ is the operation that maps d_{p-1} to d_p, using

$$d_p = \mathcal{L}_p(d_{p-1}) = \varphi(W_p d_{p-1} + b_p). \quad (6)$$

Here $W_p \in \mathbb{R}^{M_p \times M_{p-1}}$ and $b_p \in \mathbb{R}^{M_p}$ are coefficients of a linear mapping, often called weights. φ is the so-called activation function, which is in general nonlinear. In this project, we use the component-wise hyperbolic tangent function for φ. The output layer $\mathcal{L}^{\text{out}}$ is defined by

$$s \xrightarrow{\mathcal{P}} d_0 \xrightarrow{\mathcal{L}^1} d_1 \xrightarrow{\mathcal{L}^2} d_2 \xrightarrow{\mathcal{L}^{\text{out}}} \mathcal{A}(s)$$
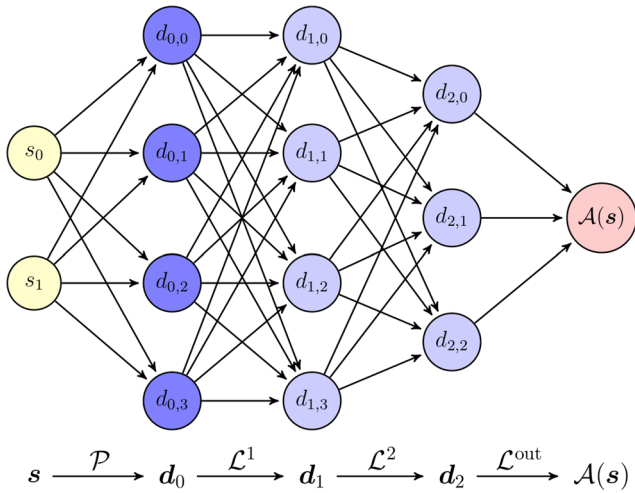
FIG. 1. A schematic plot of the DNN representation of the free energy $A(s)$. As an example, the dimension of the CV space in the figure is $M = 2$. The preprocessing operator $\mathcal{P}$ maps the CV values to an input layer that has $M_0 = 4$ nodes. The DNN has 2 hidden layers, namely, d_1 and d_2, the size of which is $M_1 = 4$ and $M_2 = 3$, respectively. The last hidden layer d_2 is mapped to the free energy $A(s)$ by the output operator $\mathcal{L}^{\text{out}}$.

$$\mathcal{A}(s) = \mathcal{L}_{\text{out}}(d_{N_h}) = W_{\text{out}} \cdot d_{N_h-1} + b_{\text{out}}, \quad (7)$$

where $W_{\text{out}} \in \mathbb{R}^{M_{p-1}}$ and $b_{\text{out}} \in \mathbb{R}$ are the weights of the linear mapping. Finally, $W = \{W_1, \ldots, W_{N_h}, W_{\text{out}}\}$ and $b = \{b_1, \ldots, b_{N_h}, b_{\text{out}}\}$ constitute all the DNN model parameters to be determined. We note that the gradient, representing the mean force

$$\mathcal{F}(s) = -\nabla_s \mathcal{A}(s), \quad (8)$$

is well defined since each layer of the construction (5) is differentiable, and hence the DNN representation of the free energy $\mathcal{A}(s)$ is also differentiable.

It should be noted that the design of the DNN model can be adapted to different kinds of problems. We use the fully connected DNN model here for simplicity of discussion. For example, for some condensed systems, an alternative network model resembling the one used in the Deep Potential method should be preferred.[21,22] We leave this to future work.

C. Training and uncertainty indicator

The DNN representation of the free energy is obtained by solving the following minimization problem:

$$\min_{\{W,b\}} L_D(\{W,b\}). \quad (9)$$

The loss function L_D is defined by

$$L_D(\{W,b\}) = \frac{1}{|D|} \sum_{s \in D} \|\mathcal{F}(s) - F(s)\|^2, \quad (10)$$

where D denotes the set of training data and $|D|$ denotes the size of the dataset D. Here $\mathcal{F}(s)$ comes from the DNN model and $F(s)$ is the collected mean force for the data s. Precise ways of collecting the data will be discussed later. It should be noted that at the beginning of the training process, we have no data. Data are collected as the training process proceeds.

To guarantee accuracy for this model, we require that the CV values in D is an adequate sample of the CV space. This is made difficult due to the barriers on the energy landscape. The MD will tend to be stuck at metastable states without being able to escape. To help overcome this problem, we introduce a biased dynamics. Details of that will be discussed in Subsection II D.

A key notion for reinforced dynamics is the uncertainty indicator. This quantity is important in the data collection step as well as in the biased dynamics step. Our intuition is that the DNN model should produce a reasonably accurate prediction of the free energy in regions that are adequately covered by D, but is much less so in regions that are covered poorly by D (or have not been visited by the MD). To quantify this, we introduce a small ensemble of DNN models, where the only difference between these models is the random weights used to initialize them. We can then define the uncertainty indicator as $\mathcal{E}(s)$, the standard deviation of the force predictions, viz.,

$$\mathcal{E}^2(s) = \langle \|\mathcal{F}(s) - \bar{\mathcal{F}}(s)\|^2 \rangle, \quad \bar{\mathcal{F}}(s) = \langle \mathcal{F}(s) \rangle, \quad (11)$$

where the ensemble average $\langle \cdots \rangle$ is taken over this ensemble of models. One expects that this ensemble of models give rise to predictions of the mean forces F that are close to each other in regions well covered by D. In the regions that are covered poorly by D, the predictions will scatter much more. This is confirmed by our numerical results.

Finally, it is worth noting that the minimization problem (9) is solved by the stochastic gradient descent (SGD) method combined with the back-propagation algorithm.[23] This has become the *de facto* standard algorithm for training DNN models. In all the test examples, we first adopt a random initialization procedure for the weights, where each component in W_p in Eq. (6) is initialized from a normal distribution with mean 0 and standard deviation $1/\sqrt{|d_{p-1}| + |d_p|}$, and each component in b_p is initialized from a normal distribution with mean 0 and standard deviation 1. Then at each training step, the weights are updated based on the evaluation of the loss function on a small *batch*, or subset B of the training data D, i.e.,

$$L = \frac{1}{|B|} \sum_{s \in B} \|\mathcal{F}(s) - F(s)\|^2. \quad (12)$$

D. Adaptive biasing

A way of encouraging the MD to overcome the barriers in the energy landscape and escape metastable regions is to add a bias to the potential. The force on the ith atom then becomes

$$\tilde{f}_i(r) = -\nabla_{r_i} U(r) - \nabla_{r_i} U_{\text{bias}}(s(r)). \quad (13)$$

Since the FES is the best approximation of the potential energy in the space of CVs, it is natural to use the current approximation of the FES, with a negative sign added, as the biasing potential, as is done in metadynamics.[9,10] We will adopt the same strategy, but we propose to switch on the biasing potential only in regions where we have low uncertainty on the DNN representation of the FES,

$$\tilde{f}_i(r) = -\nabla_{r_i} U(r) + \sigma(\mathcal{E}(s(r))) \nabla_{r_i} \mathcal{A}(s(r)), \quad (14)$$

where the biasing potential $\mathcal{A}(s(r))$ is the mean of the predefined ensemble of DNN models and $\sigma(\cdot)$ is a smooth switching function defined by

$$\sigma(e) = \begin{cases} 1, & e < e_0, \\ \frac{1}{2} + \frac{1}{2}\cos\left(\pi\,\frac{e - e_0}{e_1 - e_0}\right), & e_0 \le e < e_1, \\ 0, & e \ge e_1. \end{cases} \quad (15)$$

Here e_0 and e_1 are two uncertainty levels for the accuracy of the DNN model. In regions where the uncertainty indicator $\mathcal{E}(s)$ is smaller than the level e_0, the accuracy of the DNN representation of $\mathcal{A}(s)$ is adequate, and hence the system will be biased by $\mathcal{A}(s)$. In the regions where $\mathcal{E}(s)$ is larger than level e_1, the accuracy of the DNN representation is inadequate, and the system will follow the original dynamics governed by the potential energy $U(r)$. In between e_0 and e_1, the DNN model is partially used to bias the system via a rescaled force term $-\sigma(\mathcal{E}(s(r)))\,\nabla_{r_i}\mathcal{A}(s(r))$.

E. Data collection

After the biased MD, a number of newly visited CV values that are in the regions with high uncertainty are added to the training dataset D. The regions with high uncertainty are defined to be the CV values that give rise to large uncertainty indicator, viz., $\mathcal{E}(s) > e_{accept}$. A reasonable choice of the threshold is $e_{accept} = e_0$. For each value of the CV in D, we use the restrained dynamics to calculate the mean forces F via Eq. (3). These values, together with those computed in previous iterations, are used as the labels for training the next updated model.

F. The reinforced dynamics scheme

Figure 2 is a flowchart of the reinforced dynamics scheme. Given an initial guess of the FES represented by the DNN, a biased MD, i.e., Eq. (14), is performed to sample the CV space from an arbitrarily chosen starting point. If no *a priori* information on the FES is available, then a standard MD is carried
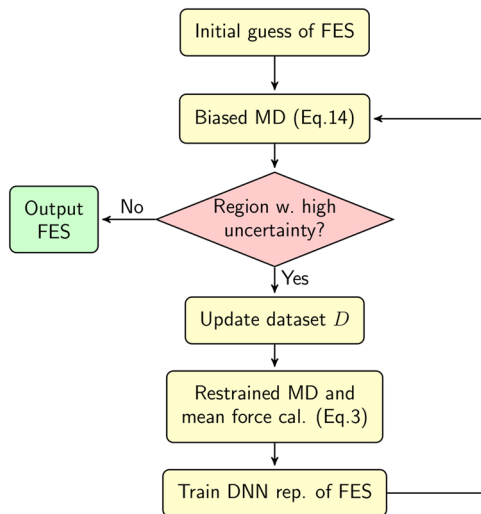

FIG. 2. The flowchart of the reinforced dynamics scheme.

out. The visited CV values are recorded at a certain time interval and tested by the uncertainty indicator to see whether they belong to a region with high uncertainty in the CV space. If all the newly sampled CV values from the biased MD trajectory belong to the region with low uncertainty, it can be that (1) the biased MD is not long enough, so parts of the CV space are not explored; (2) the interval for recording CV values along the biased MD is not small enough, so some visited CV values belonging to the region with high uncertainty are missed; or (3) the DNN representation for FES is fully converged, and then the iteration should be stopped and one can output the DNN representation for the FES, namely, the mean of the predefined ensemble of models. Case (1) can be excluded by systematically increasing the length of the biased simulation. Case (2) can be excluded by decreasing the recording interval.

If CV values belonging to the region with high uncertainty are discovered, they will be added to the training dataset D. The CV values that are already in the training dataset should be retained and serve as training data for later iterations. The mean forces at the added CV values are computed by the restrained dynamics, Eq. (3). A new ensemble of DNN models for the FES are then trained, using different random initial guesses for $\{W, b\}$. The standard deviation of the predictions from these models is again used to estimate the uncertainty indicator $\mathcal{E}(s)$. The iteration starts again using the biased MD simulation with the new DNN models.

Finally, it is worth noting that the restrained MD simulations for mean forces, which take over most of the computation time in the reinforced dynamics scheme, are embarrassingly parallelizable. The training of the ensemble of DNN models is also easily parallelizable. Several independent walkers can be set up simultaneously for a parallelized biased simulation, and this provides a more efficient exploration of the FES. These techniques can help accelerating the data collection process and benefit large-scale simulations for complex systems.

III. NUMERICAL EXAMPLES: ALANINE DIPEPTIDE AND TRIPEPTIDE

A. Simulation setup

We investigate the FES of the alanine dipeptide (ACE-ALA-NME) and alanine tripeptide (ACE-ALA-ALA-NME) modeled by the Amber99SB force field.[24] The molecules are dissolved in 342 and 341 TIP3P[25] water molecules, respectively, in a periodic simulation cell. All the MD simulations are performed using the package GROMACS 5.1.4.[26] The cut-off radius of the van der Waals interaction is 0.9 nm. The dispersion correction due to the finite cut-off radius is applied to both energy and pressure calculations. The Coulomb interaction is treated with the smooth particle mesh Ewald method[27] with a real space cutoff 0.9 nm and a reciprocal space grid spacing of 0.12 nm. The system is integrated with the leap-frog scheme at time step 2 fs. The temperature of the system is set to 300 K by a velocity-rescale thermostat[28] with a relaxation time of 0.2 ps. The solute and solvent are coupled to two independent thermostats to avoid the hot-solvent/cold-solute problem.[29] The Parrinello-Rahman barostat[30] (GROMACS implementation) with a relaxation time scale of 1.5 ps and a compressibility of 4.5×10^{-5} bar^{-1} is coupled to the system to control the

pressure to 1 Bar. For both the alanine dipeptide and tripeptide, any covalent bond that connects a hydrogen atom is constrained by the LINCS algorithm.[31] The H–O bond and H–O–H angle of water molecules are constrained by the SETTLE algorithm.[32]

For the alanine dipeptide, two torsion angles φ (C, N, C_α, C) and ψ (N, C_α, C, N) are chosen as CVs for this system, i.e., $s = (\varphi, \psi)$. While for the alanine tripeptide, the same torsion angles associated with the first and second C_αs, denoted by φ_0, ψ_0, and φ_1, ψ_1, respectively, are used as CVs for the system, i.e., $s = (\varphi_0, \psi_0, \varphi_1, \psi_1)$. The GROMACS source code is modified and linked to PLUMED 2.4b[33] to carry out the biased and restrained simulations. The PLUMED package is modified to compute the DNN biasing force, viz., Eq. (14). The DNN models used in both examples contain three hidden layers of size $(M_1, M_2, M_3) = (48, 24, 12)$. The preprocessing operator for the alanine dipeptide is taken as $\mathcal{P}(\varphi, \psi) = (\cos(\varphi), \sin(\varphi), \cos(\psi), \sin(\psi))$, so the periodic condition of the FES is guaranteed. Similarly, the preprocessing operator for the alanine tripeptide is $\tilde{\mathcal{P}}(\varphi_0, \psi_0, \varphi_1, \psi_1) = (\mathcal{P}(\varphi_0, \psi_0), \mathcal{P}(\varphi_1, \psi_1))$. Model training is carried out under the deep learning framework TensorFlow,[34] using the Adam stochastic gradient descent algorithm[35] with a batch size of $|B| = 20$. The learning rate is 0.001 in the beginning and decays exponentially according to $r_l(t) = r_l(0) \times d_r^{t/d_s}$, where t is the training step, $d_r = 0.96$ is the decay rate, and $d_s = 50 \times |D|/|B|$ is the decay step. The total number of training steps is $12\,500 \times |D|/|B|$. Currently, the DNN structure and hyperparameters in the training algorithm are decided empirically. Before performing the full reinforced dynamics, we typically accumulate some data from some small scale simulations, test the performance of different DNN models and training schemes, and then fix the optimal strategy in terms of accuracy and efficiency. In practice, we find that a DNN model with a decreasing number of nodes going from the innermost to the outermost hidden layers performs better in our test cases.

In each reinforced dynamics step, four DNN models with independent random initialization are trained in the same way to compute the uncertainty indicator. The biased MD simulations of alanine dipeptide and tripeptide last for 100 ps and 140 ps, respectively. The CV values along the MD trajectories are computed and recorded in every 0.2 ps. We assume no *a priori* information regarding the FES, so a brute-force simulation is performed for the 0th iteration step (we count the iterations from 0). In each iteration, at most 50 recorded CV values in the region with high uncertainty are added to the training dataset D. Restrained MD simulations with spring constant 500 (kJ/mol)/rad^2 are performed to estimate the mean forces by Eq. (3). Each restrained MD simulation is 100 ps and 140 ps long for the alanine dipeptide and tripeptide, respectively. The CV values are recorded in every 0.01 ps along the restrained MD trajectory to estimate the mean forces. Both of the alanine dipeptide and tripeptide examples are carried out on a desktop computer with an Intel i7-3770 central processing unit (CPU) and 32 GB memory.

B. Free energy surface construction

The FES of the alanine dipeptide on the $\varphi - \psi$ plane (known as the Ramachandran plot) is reported in Fig. 3. We
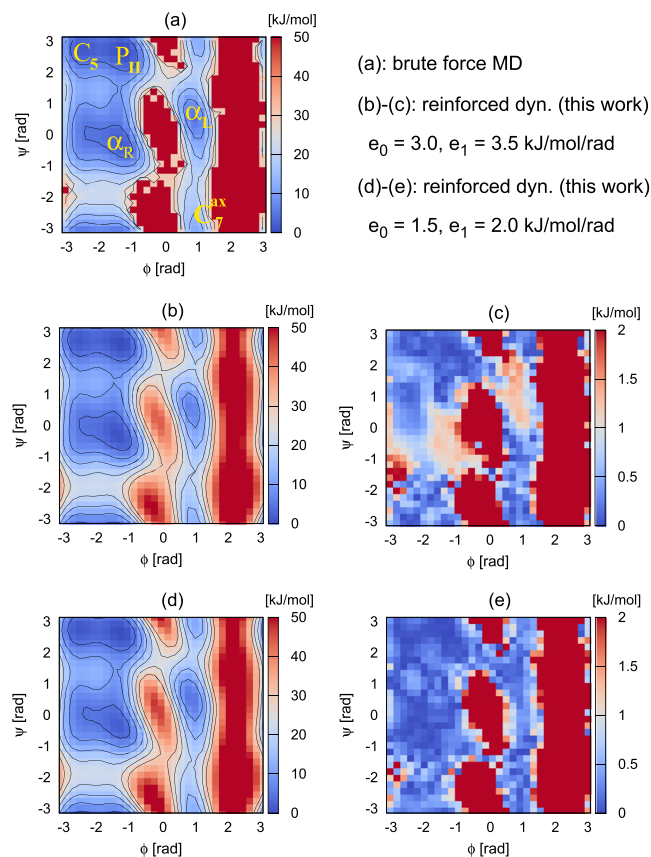

FIG. 3. The free energy of alanine dipeptide on the φ-ψ plane. The plots are obtained by (a) making log-scaled histogram of the CV values from brute-force MD simulations; [(b) and (c)] using the reinforced dynamics with uncertainty levels $e_0 = 3.0$, $e_1 = 3.5$ (kJ/mol)/rad. (b) plots the FES, and (c) plots the error compared with the brute-force MD. (d) and (e) using the reinforced dynamics with uncertainty levels $e_0 = 1.5$, $e_1 = 2.0$ (kJ/mol)/rad. (d) plots the FES, and (e) plots the error compared with the brute-force MD. The contour lines in (a), (b), and (d) are plotted from 0 kJ/mol to 30 kJ/mol with an interval of 5 kJ/mol. The red regions in (a), (c), and (d) are the CV values that are never been visited by the MD trajectories.

perform 6 independent brute-force MD simulations, with each ~860 ns long; thus, in total, 5.1 μs MD trajectories are used to estimate the FES and compare with the reinforced dynamics result. The system has 5 metastable states α_R, C_5, P_{II}, α_L, and C_7^{ax}, as noted in Fig. 3(a). The C_5 and P_{II} regions correspond to the dihedral angles observed in the β-strands conformations. The α_R and α_L regions correspond to the dihedral angles of right- and left-handed α-helix conformations, respectively. The transition between P_{II} and α_L has to go over an energy barrier of ~25 kJ/mol or equivalently ~$10k_BT$. The mean first passage time from the state P_{II} to α_L is shown to be 43 ns for the same model.[36]

In Fig. 3, the FES of alanine dipeptide sampled by the brute-force MD (a) is compared with the one constructed by reinforced dynamics (b) with uncertainty levels $e_0 = 3.0$ (kJ/mol)/rad and $e_1 = 3.5$ (kJ/mol)/rad. At the 9th iteration for (b), the biased simulation does not produce any CV value that belongs to the region with high uncertainty; thus, the computation stops. In total (from the 0th to the 8th iteration), 198 CV values are added to the training dataset D to train the FES. It is observed that the reinforced dynamics is able to reproduce, with satisfactory accuracy, the FES at the

important metastable states and transition paths of the system. The difference between (a) and (b) is plotted in (c). The error of FES at states C_5, P_{II}, and C_7^{ax} is below 0.5 kJ/mol, while the error at α_L and α_R is around 1.5 kJ/mol. The total biased MD simulation time is 10×0.1 ns = 1.0 ns. The total restrained MD simulation time is 198×0.1 ns = 19.8 ns. Thus the total MD simulation time is 20.8 ns, which is only ~0.1% of the brute-force simulation length and half of the mean first passage time from P_{II} to α_L of the brute-force MD simulation. The total wall time of all the trainings is 2.6×10^3 s, while the total wall time of all the restrained MD simulations is 8.9×10^3 s.

It is noted that the accuracy of the FES can be systematically improved by using more strict uncertainty levels. The result of using $e_0 = 1.5$ (kJ/mol)/rad and $e_1 = 2.0$ (kJ/mol)/rad is reported in Figs. 3(d) and 3(e). In this case, the biased MD simulation does not generate CV values belonging to the region with high uncertainty at the 21st iteration. In total (from the 0th to the 20th iteration), 303 CV values are added to the training dataset D to construct the FES. The error of FES at all metastable states and transition regions is uniformly below 0.5 kJ/mol. The total biased MD simulation time is 22×0.1 ns = 2.2 ns. The total restrained MD simulation time is 303×0.1 ns = 30.3 ns. Thus the total MD simulation time is 32.5 ns, which is 50% longer than the reinforced dynamics with higher uncertainty levels [$e_0 = 3.0$ (kJ/mol)/rad and $e_1 = 3.5$ (kJ/mol)/rad], but still shorter than the mean first passage time from P_{II} to α_L of the brute-force simulation (43 ns).

The information of the four-dimensional FES of the alanine tripeptide constructed by brute-force MD sampling and the reinforced dynamics is presented in Fig. 4, by projecting on the (φ_0, ψ_0), (φ_1, ψ_1), and (φ_1, ψ_0) planes. For example, the projection onto the (φ_0, φ_0) variables is defined by

$$A(\varphi_0, \psi_0) = -\frac{1}{\beta} \ln \iint d\varphi_1 d\psi_1 e^{-\beta A(\varphi_0, \psi_0, \varphi_1, \psi_1)} + C, \quad (16)$$

where C is a constant that is chosen to normalize the minimum value of $A(\varphi_0, \psi_0)$ to zero. Projected free energies $A(\varphi_1, \psi_1)$ and $A(\varphi_1, \psi_0)$ are defined analogously. The uncertainty levels of the reinforced dynamics are set to $e_0 = 3.0$ (kJ/mol)/rad and $e_1 = 3.5$ (kJ/mol)/rad. The biased MD simulation of the 72nd iteration does not find any CV value belonging to the region with high uncertainty, so the process stops. From the 0th to the 71st iteration, 1363 CV values are added to the training dataset D. The total biased MD simulation time is 73×0.14 = 10.22 ns, while the total restrained MD simulation time is 1363×0.14 = 190.82 ns. The total wall time of the restrained MD simulations is 6.2×10^4 s, while the total wall time for training the networks is 1.1×10^5 s. For comparison, we carried out 18 independent brute-force MD simulations, each of which is 2.65 μs long, so the total length of brute-force MD trajectories is 47.7 μs. Figure 4 shows that the reinforced dynamics is able to reproduce the FES with satisfactory accuracy on all the projected planes. It is noted that the projected FESs on both the (φ_0, ψ_0) and (φ_1, ψ_1) variables are different from the FES of alanine dipeptide, which indicates the correlation of backbone atoms.
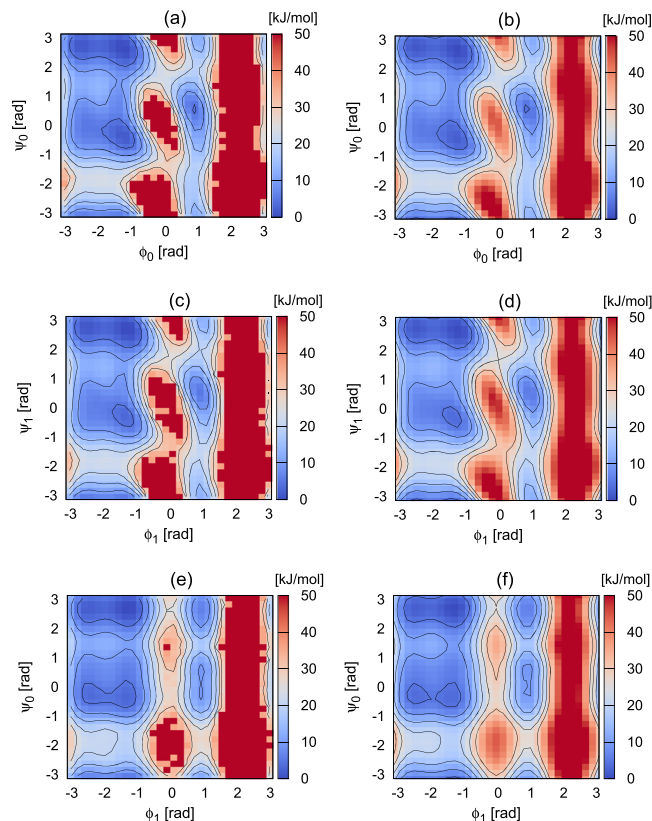

FIG. 4. The free energy of alanine tripeptide projected on the φ_0-ψ_0 [(a) and (b)], φ_1-ψ_1 [(c) and (d)], and φ_1-ψ_0 [(e) and (f)] planes. The left column, (a), (c), and (e), is histogram plots of the CV values from brute-force MD simulations; the right column, (b), (d), and (f), presents the results of reinforced dynamics up to the 71st step, which is trained by 1363 CV values in the dataset D.

C. Illustration of the adaptive feature

To highlight the adaptive feature of the reinforced dynamics, we take the alanine dipeptide as an example and illustrate in Fig. 5 the CV values visited in each biased MD simulation and those iteratively added to the training dataset D. The
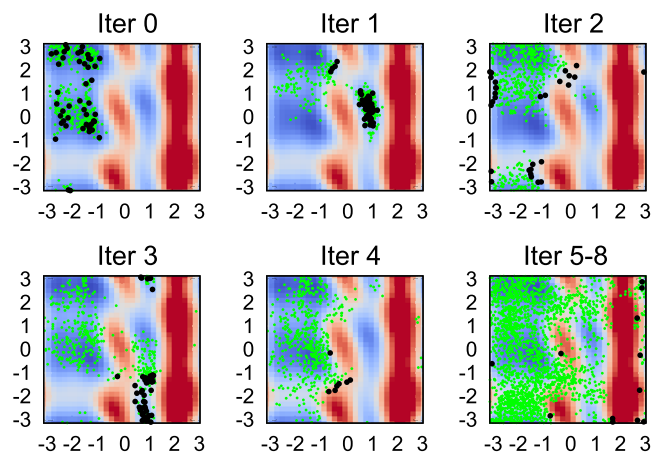

FIG. 5. The CV values visited by the biased MD simulations (thin green dots) and those added to the training dataset D (thick black dots) in each reinforced dynamics step. The uncertainty levels of the reinforced dynamics are set to $e_0 = 3.0$ and $e_1 = 3.5$ (kJ/mol)/rad. The color scale is the same as plot (b) of Fig. 3.
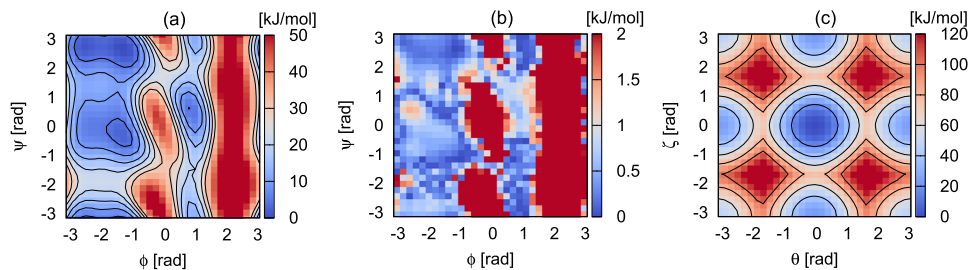
FIG. 6. The FES of the alanine dipeptide computed in the CV space (φ, ψ, θ, ζ). (a) Projection on the ϕ-ψ plane; contour lines are plotted from 0 kJ/mol to 30 kJ/mol at an interval of 5 kJ/mol. (b) Error of the ϕ-ψ projection compared to the brute-force simulation. (c) Projection on the θ-ζ plane; contour lines are plotted from 0 kJ/mol to 100 kJ/mol at an interval of 20 kJ/mol. The uncertainty levels of the reinforced dynamics are set to $e_0 = 3.0$ and $e_1 = 3.5$ (kJ/mol)/rad.

uncertainty levels are $e_0 = 3.0$, $e_1 = 3.5$ (kJ/mol)/rad, and the reinforced dynamics stops at the 9th iteration. In the 0th iteration, no *a priori* information of the FES is available, so the MD simulation is not biased. The starting state of the simulation is P_{II}, and the system spontaneously transforms to states C_5 and α_R in the 0.1 ns simulation;[43] thus; the visited CV values cover P_{II}, C_5, and α_R, and 50 of them are randomly chosen as training data. The first DNN representation of FES is trained by these CV values and is used to bias the system at the 1st iteration. Since the first DNN representation is of good quality at states P_{II}, C_5, and α_R, the system diffuses out of P_{II}, C_5, and α_R and is trapped by a new metastable state α_L. Only the visited CV values that sample the metastable state α_L are added to the training dataset. The DNN representation trained by the updated dataset is of good quality at states P_{II}, C_5, α_R, and α_L.

Following this observation, in the 2nd iteration, although the visited CV values cover a wide region including the metastable states P_{II}, C_5, α_R, and α_L, only those in the transition regions between P_{II} and α_L and between α_R and P_{II}/C_5 are added to the training set. The CV values added in the 3rd iteration are those that sample the metastable state C_7^{ax} and the transition region between C_7^{ax} and α_L. The CV values added in the 4th iteration are those that sample the transition region between C_7^{ax} and α_R.

From the 5th to the 8th iteration, the DNN representation of the FES is of relatively good quality. The CV values added to the training dataset are those that sample the border of high energy peaks at $\varphi \approx 2$ rad and $\varphi \approx -0.5$ rad. At the 9th iteration, no CV value belonging to regions with high uncertainty is found because the pushing-back events happen so quickly that the CV values are not recorded by the biased MD trajectory with the 0.2 ps recording interval. However, if we reduce the CV recording interval from 0.2 ps to 0.04 ps, 19 CV values can still be identified to be in the regions with high uncertainty and used to start the next biasing-and-training iteration. Since the construction of high energy FES peaks is of less interest, for the sake of computational cost, we do not use the smaller recording interval in our result. This means that we ignore the FES regions with sharp gradient so that the biased system can only stay for a time scale that is much shorter than the recording interval. Better stopping criteria that guarantee the representation quality of the important structures of FES and exclude the irrelevant energy peaks are left for future studies.

D. Remark on the choice of CVs

One important issue is to find the right set of CVs in order to capture the structural and dynamics information that we are interested in. However, this is a difficult problem and is not the topic of this work. Here, we will study how the enhanced sampling and free-energy estimation are affected when (unnecessary) additional CVs are included. We will see that the estimated free energy for the larger set of CVs is consistent with the one for the smaller set of CVs in the sense that after projecting the former onto the smaller set of CVs, one recovers the latter.

To this end, we compute the FES of alanine dipeptide in a four-dimensional CV space (φ, ψ, θ, ζ) with two additional torsion angles θ (O, C, N, C_α) and ζ (C_α, C, N, H). The uncertainty levels that we use are $e_0 = 3.0$ and $e_1 = 3.5$ (kJ/mol)/rad. The result is shown in Fig. 6. The 4-dimensional FES projected on the $\theta - \zeta$ plane is shown in plot (c) of the figure. The native state locates at $\theta = \zeta = 0$, while three metastable states are discovered at ($\theta = 0$, $|\zeta| = \pi$), ($|\theta| = \pi$, $\zeta = 0$), and ($|\theta| = \pi$, $|\zeta| = \pi$). They are denoted by S_{00}, S_{01}, S_{10}, and S_{11}, respectively. The barrier between the native state S_{00} and the metastable state S_{01}/S_{10} is around 70 kJ/mol. The free energy of metastable states S_{01}, S_{10}, and S_{11} are 21 kJ/mol, 24 kJ/mol, and 46 kJ/mol, respectively; thus, their contribution to the free energy projection on the $\phi - \psi$ plane is negligible. A direct comparison of the free energy projection on the $\phi - \psi$ plane with the brute-force MD result is shown in plot (b) of Fig. 6. The error is less than 2 kJ/mol. The result is consistent with the $\phi - \psi$ free energy computed using reinforced dynamics (shown in Fig. 3).

IV. APPLICATION TO POLYALANINE-10

In reinforced dynamics, both the neural network representation of the FES and the restrained simulation for mean forces are relatively insensitive to the dimensionality of the CV space. Thus it has the potential to be able to handle systems with a large set of CVs. As an illustrative example, we investigate the metastable conformations of a polyalanine-10 (ACE-(ALA)$_{10}$-NME) molecule. In this example, rather than constructing an accurate free energy in the whole space of CVs, our goal is to efficiently search for the most stable structures in the conformational space of the system. We will demonstrate that reinforced dynamics allows us to explore

very efficiently the most relevant metastable conformations of this molecule, including the α-helix and β-strand conformations, and to provide estimates for the relative stability between different metastable states.

One technical remark is that for computational efficiency, we adopt a multi-walker scheme of reinforced dynamics for this relatively high-dimensional case. In each iteration of this scheme, different walkers undergo biased dynamics independently under the same biased potential. Next, a set of CV values with high uncertainty are selected and restrained simulations are performed to calculate the associated mean forces. Finally, the selected CV values and associated mean forces provided by all the walkers are merged and added to the dataset. An ensemble of new neural network models are then trained with this larger dataset. The multi-walker scheme improves the efficiency of the data collection step, and it helps to accelerate the exploration procedure.

A. Simulation setup

The system of polyalanine-10 (ACE-(ALA)$_{10}$-NME) is modeled by the Amber96 force field.[37] The molecule is in the gas phase and is set in a 3.5 nm × 3.5 nm × 3.5 nm simulation region. To start with, we prepare misfolded initial configurations of the molecule in three stages. In the first stage, starting from an alpha-helix configuration, two ends of the molecule are pulled along the z direction in an extended simulation region (3.5 nm × 3.5 nm × 30 nm) at rate 0.1 nm/ps for 100 ps. During this process, no thermostat is used for the system. At the end of this stage, the backbone of the molecule is fully extended, and the temperature of the system increases to 1155 K. In the second stage, the pulling force is removed and the molecule is equilibrated at 300 K for 200 ps, using the velocity-rescaling thermostat[28] with 0.2 ps of relaxation time and an integration time step of 1 fs. In the third stage, an unbiased productive simulation of 200 ps is carried out at 300 K with a time step of 2 fs. 100 candidate configurations along the trajectory of this simulation are saved in every other 2 ps. Finally, 14 independent walkers are initialized with randomly chosen configurations from these candidates.

The torsion angles ϕ and ψ associated with all C$_\alpha$s are used as CVs for the system; thus, the dimension of the CV space is 20. The DNN model used in this example consists of 5 hidden layers of size $(M_1, M_2, M_3, M_4, M_5) = (360, 180, 90, 45, 20)$. We found the following procedure to be more efficient for the network training. In the first 6 iterations, the weights in different DNN models are randomly initialized and are trained using the Adam stochastic gradient descent algorithm[35] with a batch size of $|B| = 64$. The learning rate r_l is 0.003 in the beginning and decays exponentially according to $r_l(t) = r_l(0) \times d_r^{t/d_s}$, where t is the training step, $d_r = 0.96$ is the decay rate, and $d_s = 10 \times |D|/|B|$ is the decay step. The total number of training steps is $3000 \times |D|/|B|$. After iteration 6, instead of randomly initializing the weights, we restart the training process using weights inherited from the previous iteration. We use the same batch size and decay rate of the first 6 iterations but use a different learning rate of 0.0003 and a decay step of $d_s = 5 \times |D|/|B|$. This reduces the total number of training steps in each iteration to $1200 \times |D|/|B|$. The biased MD simulations are 100 ps long. The uncertainty levels are set to $e_0 = 6.0$ and e_1

= 6.5 (kJ/mol)/rad. The CV values are computed and recorded in every 0.2 ps along the biased trajectories. For each walker, at most 12 CV values in the region with high uncertainty are added to the training dataset D. For each added CV value, a 100 ps restrained MD simulation is carried out, wherein the CV values are recorded in every 0.01 ps to estimate the mean forces using Eq. (3). The simulations are carried out on one cluster node with two Intel Xeon E5-2680 v4 CPUs and 64 GB memory.

B. Structure optimization

To find different metastable states and their relative stability, we combine the exploration stage, provided by the adaptively biasing procedure in reinforced dynamics, with an optimization stage, which can be viewed as a postprocessing of the explored configurations. In the exploration stage, due to the complexity of the 20-dimensional FES, we do not wait for the reinforced dynamics to stop by itself. Instead, we stop the process at the 210th iteration. The outputs of the biased MD simulations in each iteration, in total, 14×211 = 2954 configurations, are thus selected for the next stage. We remark that basins associated with important metastable conformations may not be visited during the 210 iterations. This seems to be a common issue of algorithms for conformation space exploration, no matter by enhanced sampling or by brute-force simulation. However, reinforced dynamics drastically accelerates the efficiency of exploration and, due to the biasing procedure, new low-energy states are more likely to be explored in earlier iterations. Although we stop the process at a certain number of iterations, further tests based on the accumulated dataset and restarted from the simulation can always be performed to check the results. In the optimization stage, the 2954 configurations are first relaxed by brute-force MD for 200 ps. Then the CV values corresponding to the relaxed configurations are taken as initial guesses for the unconstrained minimization on the DNN represented FES, which is solved by the Broyden-Fletcher-Goldfarb-Shanno (known as BFGS) method,[38] and the solutions are local minima of the FES. The configurations are further relaxed with a restrained MD simulation centered at the corresponding local minima for 100 ps at a time step of 1 fs.

The local minimum with the lowest free energy corresponds to the native conformation, which is the α-helix conformation (see C004 in Fig. 7). The FES is thus shifted by the α-helix free energy so that the global minimum takes the value of 0. Among the 2954 configurations, 1047 configurations that have the free energy lower than 31.67 kJ/mol are collected. These configurations are clustered into 30 clusters according to the root mean square deviation (RMSD) of C$_\alpha$s by using the agglomerative clustering method with an average-linkage criterion[39] and are coded as C000, C001, ..., C029. The largest averaged pairwise RMSD within one cluster is 0.86 Å (C003), which indicates a high conformational similarity within the clusters. The configuration with the lowest free energy in one cluster is assigned to be the representative of that cluster, and its free energy is referred to as "the free energy" of the conformation.

The native conformation (C004) and five metastable conformations with the lowest free energies are presented in Fig. 7.
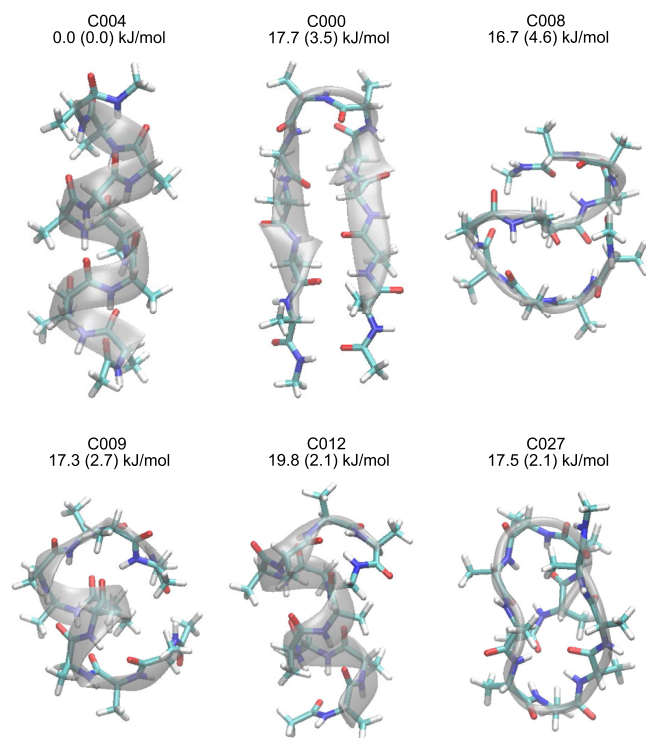
FIG. 7. Schematic plot of the native state and five most stable conformations of polyalanine-10 discovered by the reinforced dynamics. The gray shadows indicate the backbones of the conformations. Above each conformation, the cluster index (see text for details) and its free energy (in unit of kJ/mol) predicted by the reinforced dynamics are provided. The standard deviations of the free energy predictions are presented in the parentheses also in unit of kJ/mol.
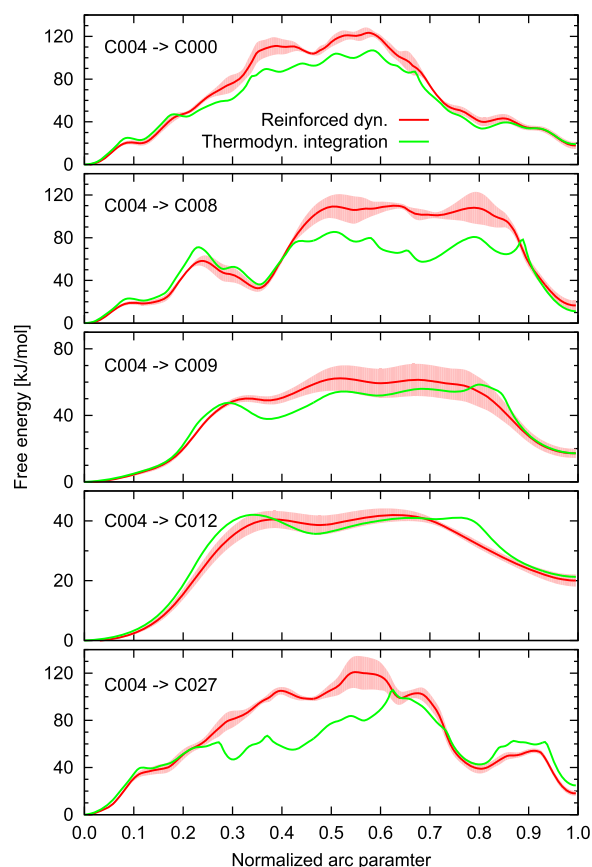

FIG. 8. The free energy along the transition paths from the native to metastable conformations. The transition paths are computed by the string method. The free energies computed by the thermodynamic integration (green lines) and predicted by the reinforced dynamics (red lines) are demonstrated. The standard deviations of the free energy predictions are presented by red shadows.

Their relative stability with respect to the native state and the standard deviations of the free energy predictions are also presented in the figure. The metastable conformation C000 corresponds to the β-strand conformation, while the metastable conformations C008, C009, C012, and C027 are misfolded conformations. The predicted free energies of the metastable conformations are very close; thus, considering the uncertainties in these free energies, we cannot tell whether one metastable state is more stable than another from the current reinforced dynamics simulation.

We also computed the transition paths from the native state to the five metastable state using the string method.[40,41] The strings are discretized by 224 nodes. At each node, a restrained MD of length 1600 ps is performed, and the CV values are recorded every 0.01 ps to compute the mean force by Eq. (3). The free energies are then computed by using thermodynamic integration along the string (see the green lines in Fig. 8). As a comparison, the free energies predicted by the reinforced dynamics along the same paths are plotted as the red lines and the standard deviations in the free energy model predictions are presented by the red shadows. The free energy predicted by the reinforced dynamics is in satisfactory agreement with the thermodynamic integration for the transitions C004 → C000, C004 → C009, and C004 → C012. The computation of the transition paths C004 → C009 and C004 → C012 is easier because the α-helical segments in the conformations C009 and C012 make them closer to the native state. It is also observed that the free energy barriers in transitions C004 → C009 and

C004 → C012 are lower than others. Along the paths C004 → C008 and C004 → C027, the reinforced dynamics is quite accurate near the native and the metastable states. However, in the middle section of the paths, there are clear differences from the result of the thermodynamic integration. Many factors may contribute to this: Between the native and a metastable state, there may exist multiple transition paths; the path computed by the string method may not be the most probable path; some conformations along the path may not be well sampled by the reinforced dynamics.

V. CONCLUSION AND PERSPECTIVE

In summary, reinforced dynamics is a promising tool for exploring the configuration space and calculating the free energy of atomistic systems. Even though we only presented examples of bio-molecules, it should be clear that the same strategy should also be applicable to many different tasks like studying the phase diagrams of condensed systems. In particular, due to the ability of the deep neural networks in representing high dimensional functions,[18,21,22,42] we expect the reinforced dynamics to be particularly powerful when the dimensionality of the CV space is high. In addition, one should be able to couple it with optimization algorithms in order to perform structural optimization.

ACKNOWLEDGMENTS

[1]S. Kumar, J. M. Rosenberg, D. Bouzida, R. H. Swendsen, and P. A. Kollman, "The weighted histogram analysis method for free-energy calculations on biomolecules. I. The method," J. Comput. Chem. **13**(8), 1011–1021 (1992).

[2]S. Kumar, J. M. Rosenberg, D. Bouzida, R. H. Swendsen, and P. A. Kollman, "Multidimensional free-energy calculations using the weighted histogram analysis method," J. Comput. Chem. **16**(11), 1339–1350 (1995).

[3]A. F. Voter, "Hyperdynamics: Accelerated molecular dynamics of infrequent events," Phys. Rev. Lett. **78**(20), 3908 (1997).

[4]Y. Sugita and Y. Okamoto, "Replica-exchange molecular dynamics method for protein folding," Chem. Phys. Lett. **314**(1), 141–151 (1999).

[5]J. VandeVondele and U. Rothlisberger, "Efficient multidimensional free energy calculations for *ab initio* molecular dynamics using classical bias potentials," J. Chem. Phys. **113**(12), 4863–4868 (2000).

[6]D. J. Earl and M. W. Deem, "Parallel tempering: Theory, applications, and new perspectives," Phys. Chem. Chem. Phys. **7**(23), 3910–3916 (2005).

[7]C. D. Christ and W. F. van Gunsteren, "Multiple free energies from a single simulation: Extending enveloping distribution sampling to nonoverlapping phase-space distributions," J. Chem. Phys. **128**, 174112 (2008).

[8]Y. Q. Gao, "Self-adaptive enhanced sampling in the energy and trajectory spaces: Accelerated thermodynamics and kinetic calculations," J. Chem. Phys. **128**, 134111 (2008).

[9]A. Laio and M. Parrinello, "Escaping free-energy minima," Proc. Natl. Acad. Sci. U. S. A. **99**(20), 12562–12566 (2002).

[10]A. Barducci, G. Bussi, and M. Parrinello, "Well-tempered metadynamics: A smoothly converging and tunable free-energy method," Phys. Rev. Lett. **100**(2), 020603 (2008).

[11]L. Maragliano and E. Vanden-Eijnden, "A temperature accelerated method for sampling free energy and determining reaction pathways in rare events simulations," Chem. Phys. Lett. **426**(1), 168–175 (2006).

[12]L. Maragliano and E. Vanden-Eijnden, "Single-sweep methods for free energy calculations," J. Chem. Phys. **128**(18), 184110 (2008).

[13]J. B. Abrams and M. E. Tuckerman, "Efficient and direct generation of multidimensional free energy surfaces via adiabatic dynamics without coordinate transformations," J. Phys. Chem. B **112**(49), 15742–15757 (2008).

[14]T.-Q. Yu and M. E. Tuckerman, "Temperature-accelerated method for exploring polymorphism in molecular crystals based on free energy," Phys. Rev. Lett. **107**(1), 015701 (2011).

[15]T. Stecher, N. Bernstein, and G. Csányi, "Free energy surface reconstruction from umbrella samples using Gaussian process regression," J. Chem. Theory Comput. **10**(9), 4079 (2014).

[16]L. Mones, N. Bernstein, and G. Csányi, "Exploration, sampling, and reconstruction of free energy surfaces with Gaussian process regression," J. Chem. Theory Comput. **12**, 5100–5110 (2016).

[17]R. Galvelis and Y. Sugita, "Neural network and nearest neighbor algorithms for enhancing sampling of molecular dynamics," J. Chem. Theory Comput. **13**(6), 2489–2500 (2017).

[18]E. Schneider, L. Dai, R. Q. Topper, C. Drechsel-Grau, and M. E. Tuckerman, "Stochastic neural network approach for learning high-dimensional free energy surfaces," Phys. Rev. Lett. **119**(15), 150601 (2017).

[19]R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction* (MIT Press, Cambridge, 1998), Vol. 1.

[20]G. Ciccotti, R. Kapral, and E. Vanden-Eijnden, "Blue moon sampling, vectorial reaction coordinates, and unbiased constrained dynamics," ChemPhysChem **6**(9), 1809–1814 (2005).

[21]J. Han, L. Zhang, R. Car, and W. E, "Deep potential: A general representation of a many-body potential energy surface," Commun. Comput. Phys. **23**(3), 629–639 (2018).

[22]L. Zhang, J. Han, H. Wang, R. Car, and W. E, "Deep potential molecular dynamics: A scalable model with the accuracy of quantum mechanics," Phys. Rev. Lett. (to be published); preprint arXiv:1707.09571 (2017).

[23]Y. A. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, "Efficient backprop," in *Neural Networks: Tricks of the Trade* (Springer, 2012), pp. 9–48.

[24]V. Hornak, R. Abel, A. Okur, B. Strockbine, A. Roitberg, and C. Simmerling, "Comparison of multiple amber force fields and development of improved protein backbone parameters," Proteins: Struct., Funct., Bioinf. **65**(3), 712–725 (2006).

[25]W. L. Jorgensen, J. Chandrasekhar, J. D. Madura, R. W. Impey, and M. L. Klein, "Comparison of simple potential functions for simulating liquid water," J. Chem. Phys. **79**(2), 926–935 (1983).

[26]M. J. Abraham, T. Murtola, R. Schulz, S. Páll, J. C. Smith, B. Hess, and E. Lindahl, "GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers," SoftwareX **1**, 19–25 (2015).

[27]U. Essmann, L. Perera, M. L. Berkowitz, T. Darden, H. Lee, and L. G. Pedersen, "A smooth particle mesh Ewald method," J. Chem. Phys. **103**(19), 8577 (1995).

[28]G. Bussi, D. Donadio, and M. Parrinello, "Canonical sampling through velocity rescaling," J. Chem. Phys. **126**, 014101 (2007).

[29]M. Lingenheil, R. Denschlag, R. Reichold, and P. Tavan, "The 'hot-solvent/cold-solute' problem revisited," J. Chem. Theory Comput. **4**(8), 1293–1306 (2008).

[30]M. Parrinello and A. Rahman, "Polymorphic transitions in single crystals: A new molecular dynamics method," J. Appl. Phys. **52**, 7182 (1981).

[31]B. Hess, H. Bekker, H. J. C. Berendsen, and J. G. E. M. Fraaije, "LINCS: A linear constraint solver for molecular simulations," J. Comput. Chem. **18**(12), 1463–1472 (1997).

[32]S. Miyamoto and P. A. Kollman, "Settle: An analytical version of the SHAKE and RATTLE algorithm for rigid water models," J. Comput. Chem. **13**(8), 952–962 (1992).

[33]G. A. Tribello, M. Bonomi, D. Branduardi, C. Camilloni, and G. Bussi, "PLUMED 2: New feathers for an old bird," Comput. Phys. Commun. **185**(2), 604–613 (2014).

[34]M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, and M. Isard *et al.*, "Tensorflow: A system for large-scale machine learning," in OSDI (USENIX Association, 2016), Vol. 16, pp. 265–283.

[35]D. Kingma and J. Ba, "Adam: A method for stochastic optimization," preprint arXiv:1412.6980 (2014).

[36]B. Trendelkamp-Schroer and F. Noe, "Efficient estimation of rare-event kinetics," Phys. Rev. X **6**, 011009 (2016).

[37]P. A. Kollman, "Advances and continuing challenges in achieving realistic and predictive simulations of the properties of organic and biological molecules," Acc. Chem. Res. **29**(10), 461–469 (1996).

[38]R. Fletcher, *Practical Methods of Optimization* (John Wiley & Sons, 1987).

[39]F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," J. Mach. Learn. Res. **12**, 2825–2830 (2011).

[40]W. E, W. Ren, and E. Vanden-Eijnden, "String method for the study of rare events," Phys. Rev. B **66**(5), 052301 (2002).

[41]W. E, W. Ren, and E. Vanden-Eijnden, "Simplified and improved string method for computing the minimum energy paths in barrier-crossing events," J. Chem. Phys. **126**(16), 164103 (2007).

[42]Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature **521**(7553), 436–444 (2015).

[43]The mean first passage time from P_{II} to C_5 and α_R is 0.041 and 0.255 ns, respectively. See Ref. 36.