On sampling minimum energy path

Mouad Ramil, Caroline Boudier, Alexandra Goryaeva, Mihai Cosmin Marinica, Jean-Bernanrd Maillet

On Sampling Minimum Energy Path

Mouad Ramil,[†] Caroline Boudier,[†] Alexandra M. Goryaeva,[‡] Mihai-Cosmin Marinica,[‡] and Jean-Bernard Maillet[*,†,¶]

[†]*CEA - DAM, DIF, Arpajon Cedex F-91297, France.*

[‡]*Université Paris-Saclay, CEA, Service de Recherches de Métallurgie Physique, Gif-sur-Yvette 91191, France*

[¶]*Université Paris-Saclay, CEA, LMCE, 91680 Bruyères-le-Châtel, France*

E-mail: jean-bernard.maillet@cea.fr

Abstract

Sampling the Minimum Energy Path (MEP) between two minima of a system is often hindered by the presence of an energy barrier separating the two metastable states. As a consequence, direct sampling based on Molecular Dynamics or Markov Chain Monte Carlo methods becomes inefficient, the crossing of the energy barrier being associated to a rare event. Augmented sampling methods based on the definition of collective variables or reaction coordinates allow to circumvent this limitation at the price of an arbitrary choice of the dimensionality reduction algorithm. We couple the statistical sampling techniques, namely Metadynamics and Invertible Neural Networks, with autoencoders so as to gradually learn the MEP and the collective variable at the same time. Learning is achieved through a succession of two steps: statistical sampling of the most probable path between the two minima and re-definition of the collective variable from the updated data points. The prototypical Mueller potential with nearly orthogonal minima is employed to demonstrate the ability of such coupling to unravel a complex MEP.

Introduction

The properties of materials are driven by the underlying atomistic free energy landscape. Direct Monte Carlo integration of the free energy variation ΔF between an initial state X_i and a final state X_f, where $X \in \mathbb{R}^{3N}$ is a given atomic configuration, is numerically challenging. This is due to the high dimensionality of the phase space that give rise to a prohibitively large sampling variance. The introduction of reaction coordinates and collective variables so as to characterise the phase space pathways reduces the dimension of the input space, rendering direct sampling tractable[1–4] when a good reaction coordinate can be found[5] . Often, minimum energy path (MEP) methods[6–8] can yield highly effective reaction coordinates with minimal domain knowledge, but the general case requires a good understanding of the problem at hand. With a good set of reaction coordinates, popular free energy calculation methods can be applied, including adaptive biasing potential approaches[1,2,9,10] where the potential energy landscape is continuously modified to accelerate sampling of the canonical measure, and closely related adaptive biasing force approaches, which instead directly modify the forces[8,11–14] to facilitate thermodynamic integration in reaction coordinate space. Recently, deep neural networks have been successfully used to learn mappings between the real systems and simple quadratic models to allow direct sampling of complex canonical distributions.[15,16]

The objective of this paper is to propose strategies to sample the MEP between metastable states using autoencoders. As a prototype, we use the Mueller potential, for which the path between the two low energy minima is almost perpendicular to the axis of main variance in the minima, thus limiting the efficiency of automatic dimensionality reduction methods like Principal Component Analysis. The first method we use is based on the Metadynamics, which is detailed in Section II. The collective variable used in this Metadynamics is computed using an autoencoder involving an augmented loss favouring low energy configurations, leading to the accurate determination of the MEP between states. The second method, described in Section III, involves Invertible Neural Networks (INN), following the seminal work of F.

Noé.[16] The INN is interfaced with an autoencoder to provide a robust way of computing the MEP without the need of sampling the dynamics.

Metadynamics and Autoencoder

Mueller potential

In this work, we are interested in sampling the transition path between two metastable states for a 2-dimensional dynamics $(X_t)_{t\geq 0}$ defined by

$$\mathrm{d}X_t = -\nabla V(X_t)\mathrm{d}t + \sqrt{2k_BT}\mathrm{d}B_t, \tag{1}$$

where $(B_t)_{t\geq 0}$ is the Brownian motion, $k_BT = 2$ (energy units) and the potential V is the Mueller-potential

$$V(x_1, x_2) = \sum_{i=1}^{4} K_i \mathrm{e}^{a_i(x_1-\beta_i)^2 + b_i(x_1-\beta_i)(x_2-\gamma_i) + c_i(x_2-\gamma_i)^2}$$

where $K, a, b, c, \beta, \gamma$ are vectors defined by $K = [-200, -100, -170, 15]$, $a = [-1, -1, -6.5, 0.7]$, $b = [0, 0, 11, 15]$, $c = [-10, -10, -6.5, 0.7]$, $\beta = [1, 0, -0.5, -1]$, $\gamma = [0, 0.5, 1.5, 1]$ with units K (energy units); a, b, c (distance units^{-2}); β, γ (distance units).

The Mueller potential is represented in Figure 1 and is composed of two main wells, denoted by A for the lowest minimum (-14 energy units) and B for the other minimum (-10 energy units). Starting from a point close to the minimum of well A, we sample 1 000 points using standard Euler scheme for molecular dynamics (skyblue dots) and similarly for well B (thistle dots). These sampled points are added to Figure 1 and are used as database points throughout this work. Notice that, given the metastable nature of the dynamics, if we sample the dynamics starting from a point inside the well A or the well B using standard molecular dynamics, the trajectory is likely to remain trapped inside the well for a very long

time. The escape rates follow a Poisson distribution[3] and in trial dynamics that extend the previous trajectory up to 100 000 iterations in A (deepskyblue dots) or B (orchid dots) in Figure 1, we have noted that the trajectory remains trapped in its initial state.

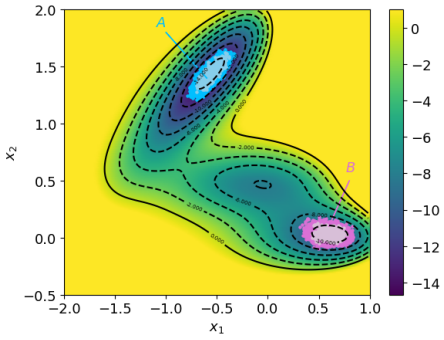


Figure 1: Energy level map of present Mueller potential and the starting database sampled from molecular dynamics around A and B energetic wells. We have used short trajectories from which we have extracted 1 000 points near A (sky blue dots), and B (thistle dots) together with extended trajectory from we have extracted 100 000 points around A (deep sky blue dots), and B (orchid dots).

Metadynamics and collective variable

Several methods have been designed to accelerate the sampling of transition pathways between metastable states, such as the Metadynamics.[2] The idea of Metadynamics is to penalize the region of phase space already visited by the dynamics in order to force an exit event from the metastable state. The usual analogy is the one of the sand bag added on each portion of the trajectory until the sand bags completely flatten the energy landscape.

The Metadynamics requires the definition of a collective variable, i.e. a function $s : (x_1, x_2) \in \mathbb{R}^2 \mapsto \mathbb{R}$ such that s takes different values in the different basins of the potential and transition states, thus capturing, in lower dimension, the multimodal (bimodal - in present particular case) metastability of the process.

The Metadynamics then corresponds to the addition of a perturbation F_t at the iteration t to the potential V such that the projection of the dynamics in the latent space, $s(X_t)$, is

uniform. The iterative perturbations added at each iteration t take the form of a Gaussian

$$X \in \mathbb{R}^2 \mapsto w\mathrm{e}^{-(s(X)-s_t)/2\sigma^2},$$

where w, σ are small fixed parameters and $s_t = s(X_t)$ is the latent projection of the dynamics at time t. The application of successive perturbations will penalize the system dynamics for reaching the neighbourhood of points $y \in \mathbb{R}^2$ having similar latent projection with visited points, i.e. such that $s(y) \approx s_t$. As a result, the perturbed potential at iteration t becomes

$$V_t : X \in \mathbb{R}^2 \mapsto V(X) + \sum_{i=1}^{t} w\mathrm{e}^{-(s(X)-s_i)/2\sigma^2}.$$

In order to find a suitable collective variable, we use a particular neural network achitecture called Autoencoder (AE).[17,18] AEs are neural networks with specific design that can be trained to perform nonlinear dimensionality reduction, called encoding, and then reconstruct the input data from the low-dimensional space, called decoding. The encoder, corresponding to the projection of the input data in the latent space, can then be used as a low-dimensional collective variable.[19–21]

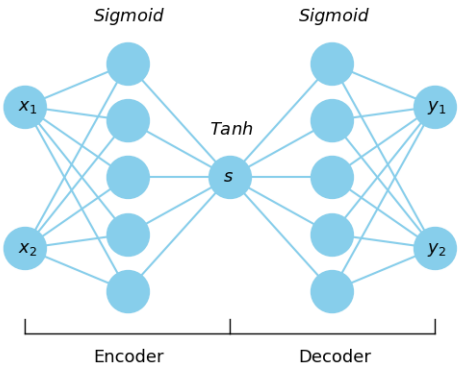


Figure 2: Autoencoder architecture of the 2D input vector $[x_1, x_2]$ reconstructed into output vector $[y_1, y_2]$ through 1D inner layer.

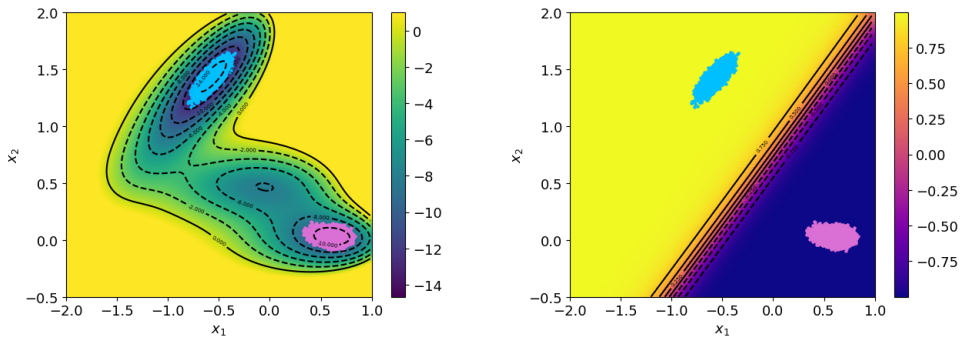
The architecture of the AE used in this work is displayed in Figure 2. Namely, we

consider a symmetric architecture with respect to the encoder and decoder parts, which are composed of one hidden layer with 5 neurons involving a Sigmoid activation function. In addition, we consider a one-dimensional latent variable s corresponding to the bottleneck of the autoencoder involving a Tanh activation function. Finally, the loss function for the AE is the mean-square norm between the input vector $[x_1, x_2]$ and the output vector $[y_1, y_2]$.

Sampling the transition path using Metadynamics

We are interested in sampling the most probable transition path between A and B corresponding to the minimum energy path (MEP).

Collective variable defined by the autoencoder. In order to obtain a suitable collective variable, we train the AE on the database from Figure 1. We then complete 50 000 iterations of Metadynamics using this collective variable starting from A and starting from B; the two trajectories are displayed in Figure 3. As hyperparameters for the Metadynamics we use $w = 0.01$ and $\sigma = 0.1$.



(a) (b)

Figure 3: Metadynamics trajectories for 50 000 iterations starting from A (blue dots) and B (purple dots) that are (a) superimposed to the level sets of the Mueller potential, and (b) superimposed to the level sets of the collective variable given by the autoencoder.

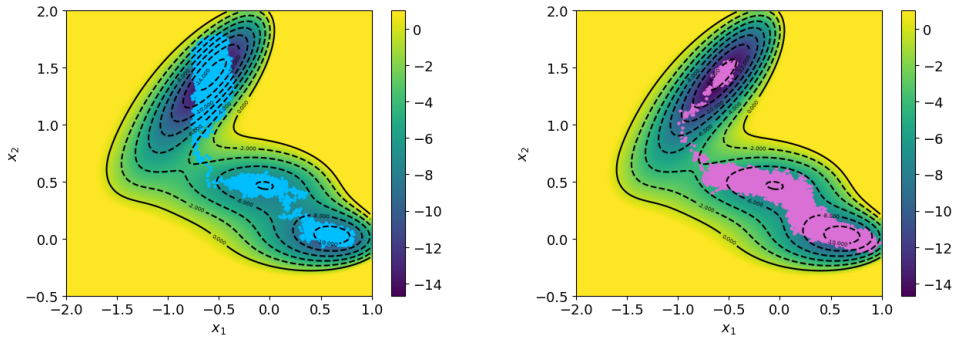
In this case, the Metadynamics does not accelerate the sampling of the transition event $A \to B$ or $B \to A$ compared to standard molecular dynamics. This can be explained by the fact that the collective variable obtained after training on the database is almost constant

in A and B (see Figure 3). In addition, if the variations of the collective variable s are negligible compared to σ then the perturbed potential becomes

$$V_t : X \in \mathbb{R}^2 \mapsto V(X) + \sum_{i=1}^{t} w \mathrm{e}^{-(s(X)-s_t)/2\sigma^2} \approx V(X) + \text{constant},$$

which would explain why the above Metadynamics does not bring much improvement compared to standard molecular dynamics.

Collective variable defined by linear interpolation in real space. In order to avoid the strong non-linear variations of the autoencoder, we can use as a collective variable the orthogonal projection on the line linking the positions of the two minima A and B. When the positions of well centers are not known in advance, they can be obtained, for instance, using a clustering algorithm, e.g., kMeans.[22] The results of the Metadynamics using this projection are displayed in Figure 4.



(a) (b)

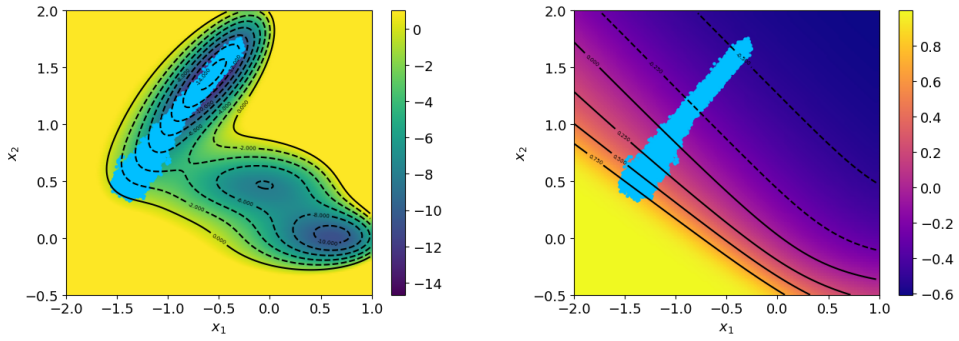
Figure 4: Metadynamics trajectories for 25 000 iterations starting (a) from A and (b) from B superimposed to the level sets of the Mueller potential.

The trajectory starting from B (path $B \to A$) corresponds indeed to the MEP we wish to sample. However, the obtained path $A \to B$ goes through sharp energetic barriers before landing to the intermediate well between A and B, and does not correspond to the MEP. This comes from the fact that the collective variable, defined as the linear interpolation between A and B, is not well suited for finding the optimal path $A \to B$ in this case.

Adaptative Metadynamics

In order to circumvent the issues arising from poor initial collective variable (Fig. 3) or misdirected collective variable (Fig. 4), we further perform a so-called iterative training of the AE based on adaptive Metadynamics. The utilty of such a strategy was previously demonstrated in the literature.[19,20,23] In this approach, the training of the AE is done adaptatively, during the metadynamics, based on the previous trajectory. Therefore, if the direction is not suited or poorly calibrated, it can be fixed at a later iteration.

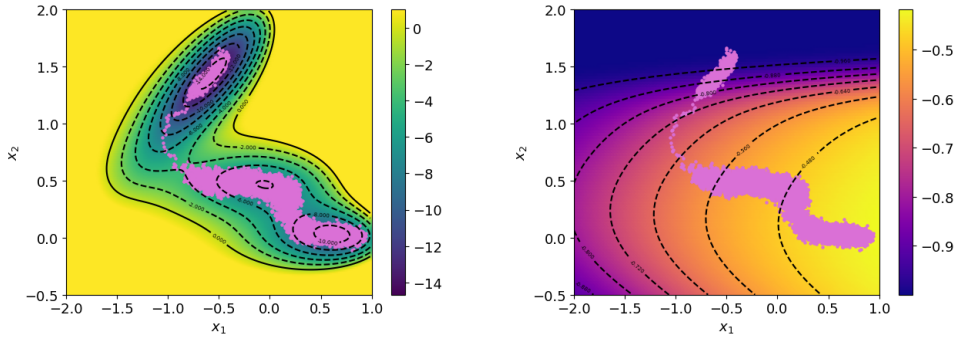
Here, we train the AE after every batch of 1 000 Metadynamics iterations on the trajectory thus far and this provides a new collective variable for the next 1 000 Metadynamics iterations. The AE is trained on a subset of around 1 000 samples taken uniformly across the trajectory in order to avoid overfitting of the autoencoder inside the wells. With this strategy, the cost of AE training remains almost the same at every batch of Metadynamics iterations. The trajectories starting from A and B are displayed in Figure 5 and Figure 6, respectively.



(a) (b)

Figure 5: Adaptative Metadynamics trajectories for 40 000 iterations starting from A (blue dots) that are (a) superimposed to the level sets of the Mueller potential, and (b) superimposed to the level sets of the collective variable given by the autoencoder.

When exploring the trajectory along the path $B \to A$, the adaptative training of the AE allows to well sample the MEP from B to A. However, this is not the case for the path $A \to B$, as the MEP is directed orthogonaly to the local variations of the dynamics on which the AE is trained. Based on these results, we conclude that application of adaptive



(a) (b)

Figure 6: Adaptative Metadynamics trajectories for 30 000 iterations starting from B (purple dots) that are (a) superimposed to the level sets of the Mueller potential, and (b) superimposed to the level sets of the collective variable given by the autoencoder.

Metadynamics for MEP sampling does not provide a general solution and some modifications, e.g., in the loss of AE, are needed to improve its performance.

Adaptative Metadynamics with modified autoencoder loss

In this section we aim to explore how modifications in the loss function of AE impact the sampling of the trajectory between the wells A and B. Below we consider modifications on the loss to ensure that the sampled path visits both A and B.

Augmented loss for path boundary conditions Since the training of the AE only captures the local dynamics, here we impose some supplementary conditions on the path through the loss of the AE such as the path is forced to visit the wells A and B. We introduce the following terms in the loss function:

$$\begin{aligned} \text{Loss} := \frac{1}{N}\sum_{i=1}^{N} \|f(X_i) - X_i\|^2 \\ + \frac{1}{2}(\|f(X_A) - X_A\|^2 + \|f(X_B) - X_B\|^2), \end{aligned} \tag{2}$$

where f is the AE, $(X_i)_{1\leq i\leq N}$ is the training dataset and X_A, X_B are the centers of mass of the wells A and B.

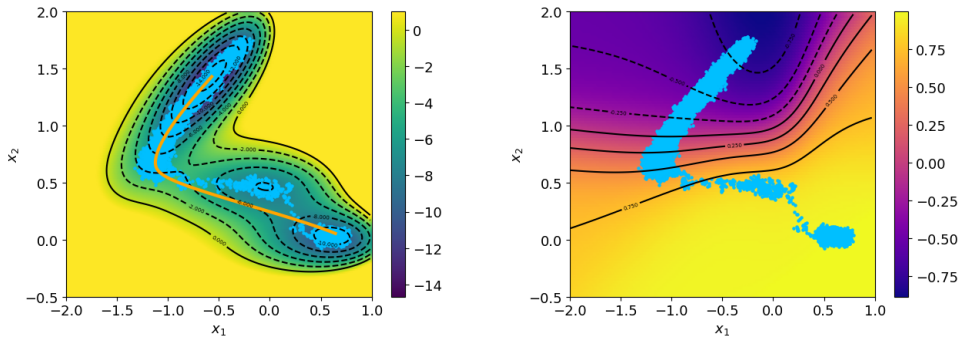
After each batch of 1000 Metadynamics iterations, the AE is now trained on the loss above. The Metadynamics trajectory obtained, starting from the well A, is displayed in Figure 7. The trajectory starting from B is available in Figure 1 of Supplementary Material (SM).

Employing the decoding part of the AE as generative model allows to verify the pertinence of the reaction coordinate in the direct space. We denote by D the decoding function of the autoencoder that maps the latent space into $\mathbb{R}^2$. The system pathway depicted in orange in Figure 7 and Figure 1 of SM corresponds to the system positions generated from the intermediate states between the latent projections of the minima A and B. Here, we chose the discretized points along the segment between the latent projections of A and B, $s_A := s(X_A)$ and $s_B := s(X_B)$. We sample $m = 100$ points along the trajectory $A \to B$ by defining the intermediate points s_i as

$$\forall 0 \leq i \leq m, \qquad s_i = s_A + \frac{i}{m}(s_B - s_A). \tag{3}$$

The sequence of reconstructed points $D(s_0 = s_A)$, ..., $D(s_i)$, ..., $D(s_m = s_B)$ generates the orange pathway presented in Figure 7. The generated trajectory can be interpreted as the most-liked pathway between A and B encoded by the collective variable.

With the augmented loss function, we obtain a better sampling of the transition path $A \to B$ and the path $B \to A$ remains close to the MEP. However, we notice that the collective variable for $A \to B$ in Figure 7a favours a path (orange line) which does not optimize completely the visit of low energy configurations. However, this is not disqualifying since dynamic trajectories at finite temperature are not bound to the lowest part of the underlying potential: there is always an interplay between the value of the energy along the pathway and the local curvature of the energy landscape.



(a) (b)

Figure 7: Adaptative Metadynamics trajectories for 38 000 iterations starting from A (blue dots) that are (a) superimposed to the level sets of the Mueller potential, and (b) superimposed to the level sets of the collective variable given by the autoencoder. The orange curve in (a) represents the path followed by the collective variable.

Augmented loss for Minimum Energy Path. In order to favour low energy configurations for the collective variable, we add an additional term on the AE loss accounting for the "energy-length" of the path. This additional term represents a trade off between the length of the pathway and the magnitude of the energy along pathways. The idea is to favour the sampling of pathways between mimima that i) are as short as possible and ii) which explore the lowest part of the energy landscape i.e. where the minima and first order saddle points are located.

For this purpose we plug into the loss of AE an additional term E_{path} accounting for the "energy-length" of the path that has the dimension of an energy $\times$ distance. We will employ again the generative function $D(s)$ of AE and the same latent discretization given previously in Eq (3). Using the decoder D and projecting from latent to configuration space, the pathway energy-length is then defined as follows:

$$E_{path} = \sum_{i=1}^{m-1} \|D(s_{i+1}) - D(s_i)\|(V(D(s_i)) + C),$$

where the constant $C = -\min_i V(D(s_i)) + \alpha$ with $\alpha = 0.1$ is added to ensure that $V(D(s_i)) + C$ remains positive for all i.

The new loss obtained is the following:

$$\begin{aligned}\text{Loss} = \frac{1}{N}\sum_{i=1}^{N} \|f(X_i) - X_i\|^2 \\ + (\|f(X_A) - X_A\|^2 + \|f(X_B) - X_B\|^2)/2 \\ + E_{path}.\end{aligned} \tag{4}$$

Figure 2 in Supplementary Material (SM) illustrates the trajectory starting from well A, which was obtained by integrating the new loss into the AE during the Adaptative Metadynamics. The trajectory starting from well B is provided in Figure 3 of Supplementary Material (SM). We notice that in this case the path following the collective variable visits low energy configurations and approximates very well the MEP.

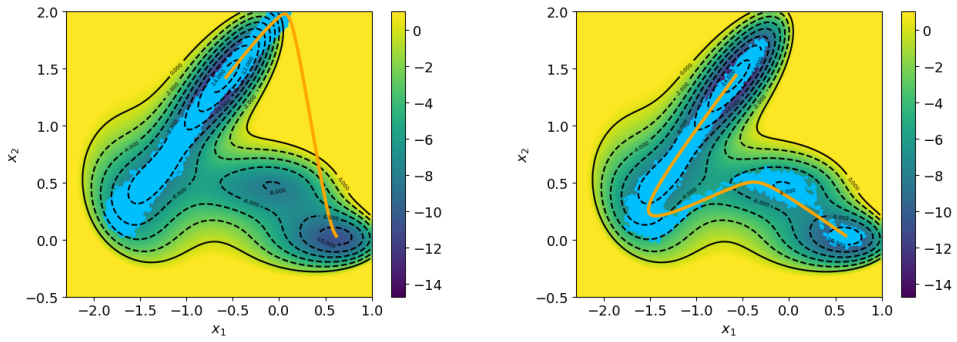
However, regarding the sampling of the trajectory between A and B we do not observe significant differences between the dynamics with the AE loss given by Eqs (2) and (4). Most likely this is due to the topology of the potential close to the bottleneck, which makes it not affected by changes of the collective variable in this zone.

Therefore, in order to emphasize the qualitative differences between the trajectories sampled employing the losses Eq (2) and Eq (4), we will use a modified version of the Mueller potential. The modified potential $\tilde{V}$ is obtained by perturbing the Mueller potential V as follows:

$$\tilde{V}(X) = V(X) + \left(-100 + \|X - \eta\|^2\right) \mathrm{e}^{-2\|X-\eta\|^2}, \tag{5}$$

with $\eta = [-1.7, 0.2]$.

In Figure 8, we provide the sampled trajectories $A \to B$ using AE losses provided by Eq (2) and Eq (4). We notice that the collective variable trained without considering the term E_{path} in the loss (Fig. 8a) goes through regions with sharp increasing gradient potential, whereas the other loss favours (Fig. 8b) low energy configurations, thus managing to sample a path $A \to B$ close to the MEP. This result highlights the need of considering E_{path} in the



(a) (b)

Figure 8: Adaptative Metadynamics trajectories for 100 000 (resp. 45 000) iterations starting from A (blue dots) trained with (a) the AE loss of Eq (2) and (b) the AE loss of Eq. (4), superimposed to the level sets of the Mueller potential. The orange curve represents the path followed by the collective variable.

AE loss.

To conclude this section, training the AE using the augmented loss Eq (4) enables to sample paths close to the MEP, even in the worst scenario where the axis of principal variance of the two minima are orthogonal.

Autoencoder and invertible neural network

Invertible neural network

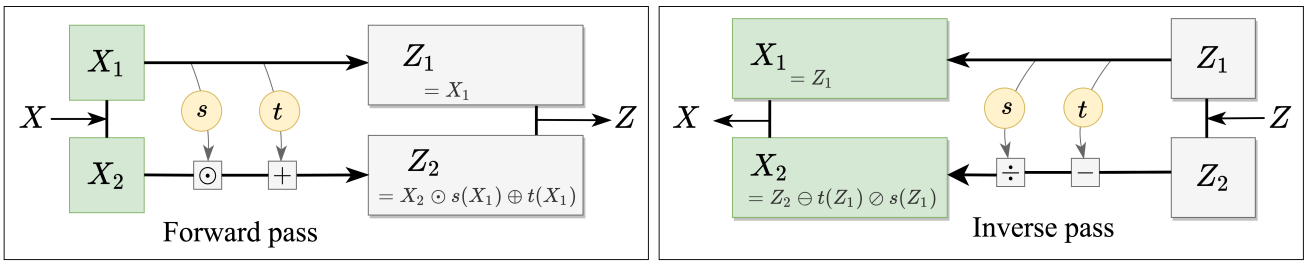


Figure 9: General idea of an INN building block with input $X \in \mathbb{R}^n$ and output $Z \in \mathbb{R}^n$. The input X is divided in two parts : X_1 and X_2. A linear transformation is applied on X_2 at the coefficient level ($\odot$ being the Hadamard product), while X_1 is conserved. These coefficients are functions applied on X_1 (denoted here by s and t). They can be as complex as needed as their inverse is not required to inverse the block. The symmetric operation (replacing X_1 by X_2) would be computed next to obtain a complete building block

An Invertible Neural Network (INN) is a neural network built as a bijective function between two spaces : the input space (also called latent space) and the output space (that will be referred to as the configuration space). Most classical deep neural networks present too complex architectures to be easily invertible. However, the idea behind INNs is to use an ingenious basic building block depicted in Figure 9. The structure of this block makes it an easily invertible function. In practice, an INN will be composed of a series of these basic building blocks (also adding some other simple layers such as normalization or permutation), making the whole network invertible. On top of their invertibility, the INNs are often designed as flow neural networks,[24] that allow for a tractable computation of the density of the sampled distribution through the INN. This property is particularly interesting to design loss functions built to push the sampling towards a known distribution (such as the Gibbs measure).

In this work, the INN considered is composed of 5 building blocks in series. Again, we are interested in the sampling of a transition path between the wells A and B. The method used in this section will resort to the INN architecture described above following a method described in.[16]

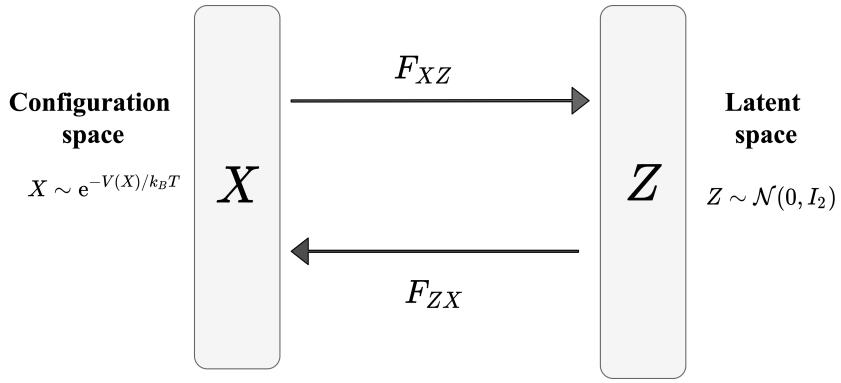


Figure 10: General set-up of the Invertible Neural Network

As described in Figure 10, we train the INN such that starting from the 2-dimensional normal distribution the output samples follow the Gibbs measure, associated to the Mueller potential, which density is proportional to $e^{-V(X)/k_BT}$. Likewise, its inverse should return

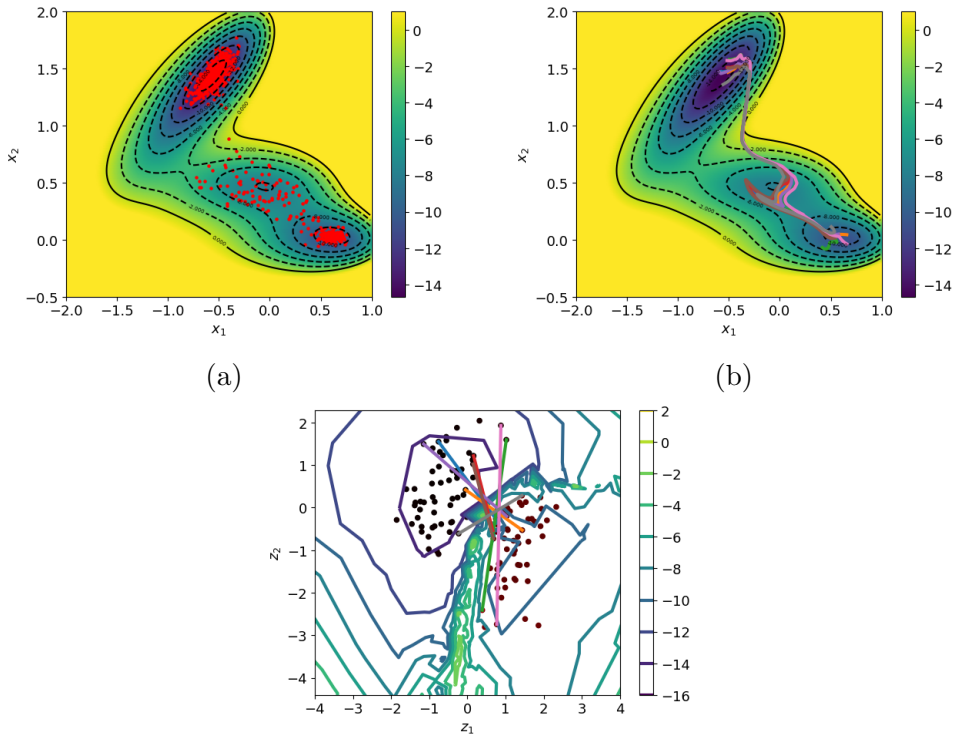


Figure 11: (a): INN sampling after full training (1 000 points) superimposed to the energy level sets of the Mueller potential. (b): Paths obtained by linear interpolation in latent space projected in real space. (c): Sampled points by molecular dynamics projected in latent space, together with linear interpolations in latent space (colored lines), superimposed to the energy level sets of the Mueller potential. The colors of lines with linear interpolations in (c) correspond to the color of paths in (b).

Gaussian samples when the input are distributed according to the Gibbs measure. In other words, if F_{ZX} is the INN output function and F_{XZ} its inverse, we train the INN such that:

- if $Z \sim \mathcal{N}(0, I_2)$, then $F_{ZX}(Z) \sim \mathrm{e}^{-V(X)/k_BT}$,
- if $X \sim \mathrm{e}^{-V(X)/k_BT}$, then $F_{XZ}(X) \sim \mathcal{N}(0, I_2)$.

The space function of the Gaussian samples Z can be seen as a latent space. However, the nature of this latent space is different from the latent space of the AE. Here the samples Z are in a space with same dimension as the input (2) while the encoder of the AE projected in a latent space with lower dimension (1). Therefore, we use a different notation Z for the latent variables of the INN here instead of s used for the latent variables of the autoencoder.

The loss function is computed similarly to [16] using the Kullback-Leibler (KL) divergence between the above distributions. The KL divergence between the law of $F_{ZX}(Z)$ for $Z \sim \mathcal{N}(0, I_2)$ and the Gibbs measure yields the following term

$$J_{ML} = \mathbb{E}\left[\|F_{XZ}(X)\|^2 - \log(R_{XZ})\right],$$

where R_{XZ} is the jacobian of F_{XZ} and can be easily computed for flow neural networks composed of building blocks as in Figure 9 (see for instance[25]). The KL divergence between the law of $F_{XZ}(X)$ for $X \sim \mathrm{e}^{-V(X)/k_BT}$ and the 2-dimensional normal distribution leads to the following loss:

$$J_{KL} = \mathbb{E}\left[V(F_{ZX}(Z)) - \log(R_{ZX})\right],$$

where R_{ZX} is the jacobian of F_{ZX}.

This training ensures that the sampled distributions match the expectations : a normal distribution in the latent space and a distribution that is energetically coherent in the configuration space. However, since we are here mostly interested in a transition path between two states, we favour the sampling in the transition direction. We thus add a third loss to the two previous ones. This loss will promote oversampling in the transition path direction between two bounds. More precisely, given a collective variable function s, and an estimation of the probability p^1 we consider the following loss defined by

$$J_{RC} = \mathbb{E}\left[\log(p(s(F_{zx}(Z))))\right].$$

Thus, as in,[16] the overall loss is defined by

$$J = w_{ML}J_{ML} + w_{KL}J_{KL} + w_{RC}J_{RC},$$

where w_{ML}, w_{KL}, w_{RC} are given weights.

[1] computed as a kernel density estimate between the predefined bounds on a batch of data[16]

The training of the INN is done as follows:

- during 150 epochs, we train the INN to minimize only the J_{ML} loss (batch size is 100)
- during 150 additional epochs, we train the INN to minimize the total loss with equal weights (same batch size)

As in the previous section, the collective variable s used in the loss J_{RC} can be obtained by training an AE on a database composed of sampled data in A and B, or by using a clustering algorithm, like KMeans,[22] and consider the orthogonal projection on the line linking both wells.

Once the training is complete, the transition path between the wells A and B can be sampled following the procedure exposed in[16] and summarized below: (i) take random points y_A, y_B belonging to the A and B minima; (ii) compute their images $z_A, z_B \in \mathbb{R}^2$ in the latent space, i.e. $z_A = F_{XZ}(y_A)$ and $z_B = F_{XZ}(y_B)$; (iii) perform a linear interpolation in the latent space $w_i = z_A + i(z_B - z_A)/m$ $(0 \leq i \leq m)$ between the points z_A, z_B; (iv) finally, project this segment back onto the Cartesian (or configuration) space.

The resulting sequence of points $X_i = F_{ZX}(w_i)$ provides a transition path between A and B in the configuration space, which could be associated to the MEP if the INN is well trained.

Sampling transition path using an INN

Collective variable obtained by linear interpolation in the configuration space

The training of the INN is first performed using a collective variable given by the orthogonal projection on the line linking both wells A and B (in the cartesian space). The INN is trained on the same database of Figure 1, as was done in the previous section. Using kMeans, we obtain the coordinates of the centers of mass on our database $[-0.57, 1.43]$ for A and $[0.62, 0.03]$ for B.

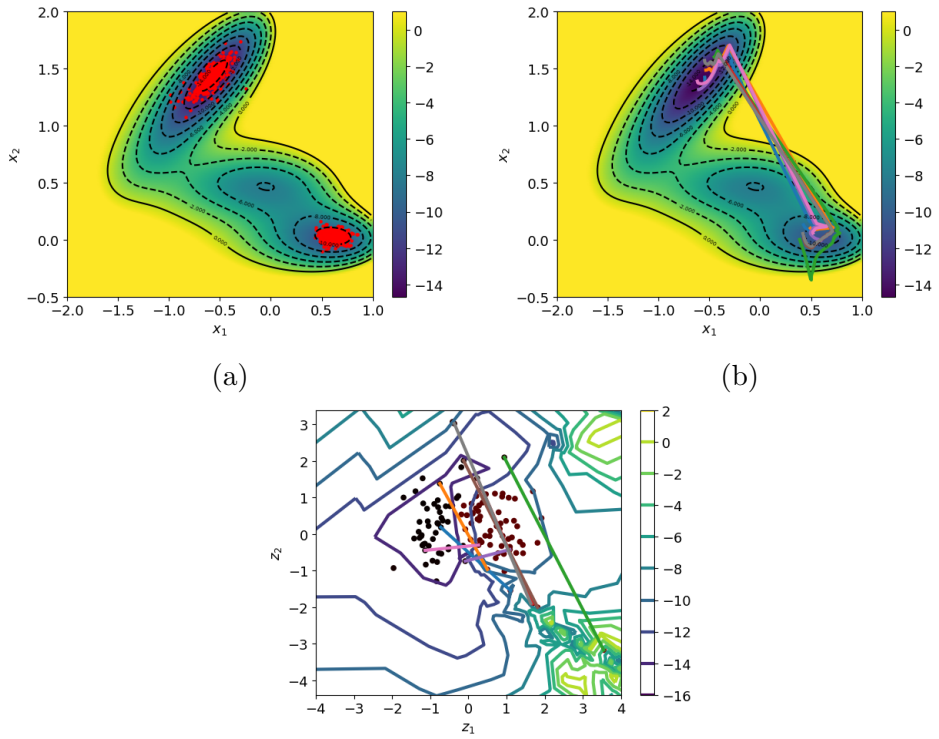


Figure 12: Direct training with the autoencoder. (a) INN sampling after full training (10 000 points). (b) Linear interpolation in latent space projected in configuration space. (c) Sampled points by molecular dynamics projected in latent space together with linear interpolations in latent space (colored lines). The colors of lines with linear interpolations in (c) correspond to the color of paths in (b).

The INN is trained on the ML loss and then the complete loss J with all the weights w equal to 1. We then sample in the Cartesian space by inputting 1000 Gaussian samples to the INN and passing them through the network. We also use the previous linear interpolation technique to plot 8 transition paths between A and B. The sampling and paths are displayed in Figure 11.

This procedure leads to a better sampling between states than standard MD methods, at a reduced CPU cost. However, the projected transition path in real space is still far from the optimal MEP.

The procedure has been reiterated several times, the final results depending on the efficiency of the INN training and on the resulting structure of the latent space.

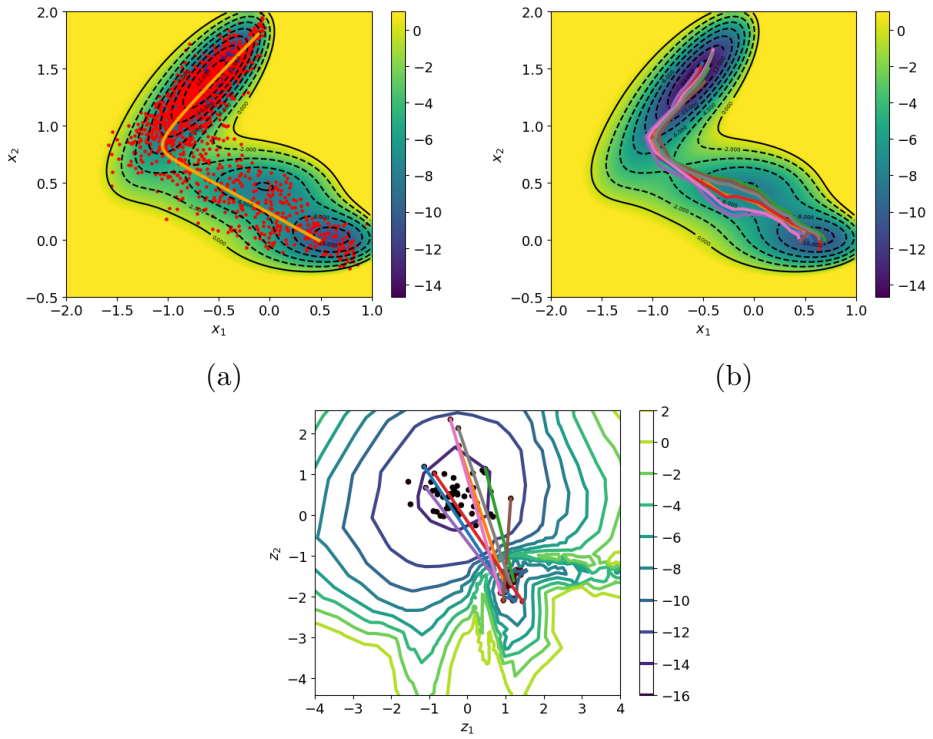


Figure 13: Adaptative sampling after 57 cycles. (a): INN sampling after full training (10 000 of points). (b): Linear interpolation in latent space projected in configuration space. (c): Sampled points by molecular dynamics projected in latent space together with linear interpolations in latent space (colored lines).

Collective variable obtained by an autoencoder

Instead of using an arbitrary definition, the collective variable (used in the J_{RC} loss) is now obtained by training an autoencoder. We keep the same architecture (Figure 2) and database. The subsequent CV is the same as the one used initially in the Metadynamics part and displayed in Figure 3b). The sampling and path obtained are shown in Figure 12.

The sampling is less efficient than in the previous case due to the non-linearity of the AE, particularly showing flat regions surrounding the minima and concentrating its gradient in the middle of the MEP. As a result, the proposed MEP (projection in the real space of the linear interpolation in the latent space) is not optimal.

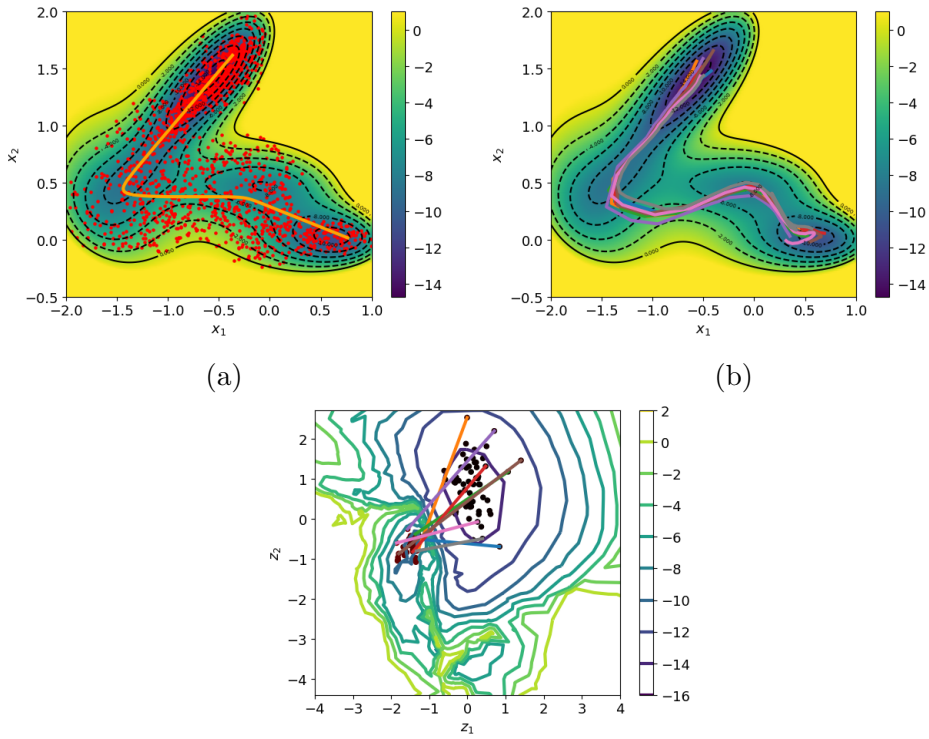


Figure 14: (a): INN sampling after full training (10 000 of points). (b): Linear interpolation in latent space projected in configuration space. (c): Sampled points by molecular dynamics projected in latent space together with linear interpolations in latent space (colored lines).

Sampling transition path adaptatively

In order to improve the accuracy of the predicted transition path we propose to retrain the autoencoder with the data previously obtained by the INN. The following cycle is proposed:

- Sample 10 000 Gaussians points $(Z_i)_{1\leq i\leq 10000}$ with the INN.
- Train an autoencoder on this new database of samples $(F_{ZX}(Z_i))_{1\leq i\leq 10000}$ during 50 epochs.
- Train an INN on the same samples $((F_{ZX}(Z_i))_{1\leq i\leq 10000})$ using the collective variable provided by the autoencoder.

One cycle then corresponds to the training of the autoencoder followed by the training of the INN. This cycle is repeated with the hope of converging towards the best collective

variable and an accurate MEP. In this case, 57 cycles proves to be sufficient to converge to an accurate MEP. Results are shown in Figure 13. The orange points correspond to the encoding then decoding by the AE of the points sampled by the INN.

We also test the cycle training of the INN on the modified potential $\tilde{V}$ Eq. (5) used in the previous section. After 102 iterations we obtain a path close to the MEP in this case as well. The results for this potential are displayed in Figure 14.

In summary, on the two-dimensional case of the Mueller potential, the iterative training of the INN and the autoencoder is able to significantly improve the definition of the collective variable and sampling of the MEP.

Conclusions and perspectives

We have developed two methods to improve the sampling of a transition path between metastable sates. The first one relies on an extension of the Metadynamics used in conjunction with an autoencoder for the definition of the collective variable. We showed that a modification of the autoencoder loss function is required to ensure a precise definition of the collective variable, in particular in the case of the Mueller potential where the axis of principal variance associated to the two minima are orthogonal. The second method is based on the coupling between an Invertible Neural Network and an autoencoder. In both cases, an accurate sampling of the transition path is achieved through an iterative procedure of training the autoencoder and sampling the transition path. This procedure allows to learn adaptatively the collective variable together with the transition path.

Both methods are able to provide a path between the states close to the MEP. Moreover, since these methods mostly rely on the use of neural networks,they can be scaled to larger dimensional real systems. In addition, the augmented loss in the autoencoder could be applied to provide a reaction coordinate for other biasing methods like Adaptative Biasing Force, for instance.

Finally, these two methods perform at a significantly reduced CPU cost compared to traditional sampling methods. The additional cost required for the computation of the augmented loss in the first method remains negligible compared to the cost of Metadynamics. The second method with the INN provides faster results but the number of cycles needed to converge to the MEP is a priori unknown and a convergence criterion for the MEP indeed remains to be defined.

In perspective, the application of these methods to higher dimensional systems can help finding transition paths for complex systems at finite temperature.

References

(1) Torrie, G. M.; Valleau, J. P. Nonphysical sampling distributions in Monte Carlo free-energy estimation: Umbrella sampling. *J. Comp. Phys.* **1977**, *23*, 187–199.

(2) Laio, A.; Parrinello, M. Escaping free-energy minima. *Proc. Natl. Acad. Sci. U. S. A.* **2002**, *99*, 12562–12566.

(3) Lelièvre, T.; Stoltz, G.; Rousset, M. *Free energy computations: A mathematical perspective*; Imperial College Press, London, 2010.

(4) Chipot, C.; Pohorille, A. Free energy calculations. 2007.

(5) Noé, F.; Clementi, C. Collective variables for the study of long-time kinetics from molecular trajectories: theory and methods. *Current Opinion in Structural Biology* **2017**, *43*, 141–147.

(6) Weinan, E.; Ren, W.; Vanden-Eijnden, E. String method for the study of rare events. *Phys. Rev. B* **2002**, *66*, 052301.

(7) Maragliano, L.; Vanden-Eijnden, E. On-the-fly string method for minimum free energy paths calculation. *Chem. Phys. Lett.* **2007**, *446*, 182–190.

(8) Swinburne, T. D.; Marinica, M.-C. Unsupervised Calculation of Free Energy Barriers in Large Crystalline Systems. *Phys. Rev. Lett.* **2018**, *120*, 135503.

(9) Lelièvre, T.; Rousset, M.; Stoltz, G. Computation of free energy profiles with parallel adaptive dynamics. *J. Chem. Phys.* **2007**, *126*, 134111.

(10) Bonati, L.; Parrinello, M. Silicon Liquid Structure and Crystal Nucleation from Ab Initio Deep Metadynamics. *Phys. Rev. Lett.* **2018**, *121*, 265701.

(11) Darve, E.; Pohorille, A. Calculating free energies using average force. *J. Chem. Phys.* **2001**, *115*, 9169.

(12) Darve, E.; Wilson, M. A.; Pohorille, A. Calculating Free Energies Using a Scaled-Force Molecular Dynamics Algorithm. *Mol. Simul.* **2002**, *28*, 113.

(13) Darve, E.; Rodríguez-Gómez, D.; Pohorille, A. Adaptive biasing force method for scalar and vector free energy calculations. *J. Chem. Phys.* **2008**, *128*.

(14) Lelièvre, T.; Rousset, M.; Stoltz, G. Long-time convergence of an adaptive biasing force method. *Nonlinearity* **2008**, *21*, 1155.

(15) Noé, F.; Clementi, C. Kinetic Distance and Kinetic Maps from Molecular Dynamics Simulation. *Journal of Chemical Theory and Computation* **2015**, *11*, 5002–5011.

(16) Noé, F.; Olsson, S.; Köhler, J.; Wu, H. *Science* **2019**, *365*, 6457.

(17) Hinton, G. E.; Salakhutdinov, R. R. Reducing the dimensionality of data with neural networks. *Science* **2006**, *313*, 504–507.

(18) Scholz, M.; Fraunholz, M.; Selbig, J. Nonlinear principal component analysis: neural network models and applications. Principal manifolds for data visualization and dimension reduction. 2008; pp 44–67.

(19) Belkacemi, Z.; Gkeka, P.; Lelièvre, T.; Stoltz, G. Chasing Collective Variables using Autoencoders and biased trajectories. e-print arXiv:Physics / Biological Physics:2104.11061, 2021.

(20) Chen, W.; Ferguson, A. L. Molecular enhanced sampling with autoencoders: On-the-fly collective variable discovery and accelerated free energy landscape exploration. *J. Comput. Chem.* **2018**, *39*, 2079–2102.

(21) Wehmeyer, C.; Noé, F. Time-lagged autoencoders: Deep learning of slow collective variables for molecular kinetics. *J. Chem. Phys.* **2018**, *148*, 241703.

(22) Steinley, D.; Brusco, M. J. Initializing K-means batch clustering: A critical evaluation of several techniques. *J. Classif.* **2007**, *24*, 99–121.

(23) Ribeiro, J. M. L.; Bravo, P.; Wang, Y.; Tiwary, P. Reweighted autoencoded variational Bayes for enhanced sampling (RAVE). *J. Chem. Phys.* **2018**, *149*, 072301.

(24) Dinh, L.; Sohl-Dickstein, J.; Bengio, S. Density estimation using Real NVP. *CoRR* **2016**, *abs/1605.08803*.

(25) Ref. 16, Appendix.

TOC Graphic