Learning mappings between equilibrium states of liquid systems using normalizing flows

Alessandro Coretti, Sebastian Falkner, Phillip L. Geissler, Christoph Dellago

Angaben zur Veröffentlichung / Publication details:

Coretti, Alessandro, Sebastian Falkner, Phillip L. Geissler, and Christoph Dellago. 2025. "Learning mappings between equilibrium states of liquid systems using normalizing flows." *The Journal of Chemical Physics* 162 (18): 184102.
https://doi.org/10.1063/5.0253034.

The Journal of Chemical Physics

RESEARCH ARTICLE | MAY 08 2025

Learning mappings between equilibrium states of liquid systems using normalizing flows

Special Collection: Molecular Dynamics, Methods and Applications 60 Years after Rahman

Alessandro Coretti ; Sebastian Falkner ; Phillip L. Geissler ; Christoph Dellago

Check for updates

J. Chem. Phys. 162, 184102 (2025)
https://doi.org/10.1063/5.0253034

AIP Publishing

The Journal
of Chemical Physics

ARTICLE

pubs.aip.org/aip/jcp

Learning mappings between equilibrium states of liquid systems using normalizing flows

Cite as: J. Chem. Phys. 162, 184102 (2025); doi: 10.1063/5.0253034
Submitted: 13 December 2024 • Accepted: 16 April 2025 •
Published Online: 8 May 2025

View Online Export Citation CrossMark

Alessandro Coretti,[1,a)] **Sebastian Falkner,**[1,2] **Phillip L. Geissler,**[3,b)] **and Christoph Dellago**[1]

AFFILIATIONS

[1] Faculty of Physics, University of Vienna, 1090 Vienna, Austria
[2] Institute of Physics, University of Augsburg, Universitätsstraße 1, 86159 Augsburg, Germany
[3] Department of Chemistry, University of California, Berkeley, California 94720, USA

Note: This paper is part of the JCP Special Topic on Molecular Dynamics, Methods and Applications 60 Years After Rahman.
[a)] **Author to whom correspondence should be addressed:** alessandro.coretti@univie.ac.at
[b)] Phillip L. Geissler passed away on July 17, 2022, during the initial stages of this work.

ABSTRACT

Generative models and, in particular, normalizing flows are a promising tool in statistical mechanics to address the sampling problem in condensed-matter systems. In this work, we investigate the potential of normalizing flows to learn a transformation to map different liquid systems into each other while allowing at the same time to obtain an unbiased equilibrium distribution. We apply this methodology to the mapping of a small system of fully repulsive disks modeled via the Weeks–Chandler–Andersen potential into a Lennard-Jones system in the liquid phase at different coordinates in the phase diagram. We obtain an improvement in the relative effective sample size of the generated distribution up to a factor of six compared to direct reweighting. We show that this factor can have a strong dependency on the thermodynamic parameters of the source and target system.

I. INTRODUCTION

The generation of properly weighted samples from physical distributions is a central problem in statistical mechanics and computational physics. Monte Carlo[1] and molecular dynamics[2,3] simulations are the state of the art approaches for producing samples from equilibrium distributions, but they do not come without drawbacks. Being based on sequential updates of configurations, they can be slow in exploring configuration space due to statistical correlations between subsequent configurations. As a result, it is computationally expensive to generate completely decorrelated samples. In complex systems with rare transitions, these correlations can become a limiting factor of the sampling efficiency.

Machine learning and, in particular, generative models[4–6] are a promising alternative to overcome some of these problems, and their use in physics is increasing steadily.[7–9] These models are able, in principle, to generate decorrelated samples from an arbitrary distribution at the cost of having a relatively large pool of examples on which the model can be trained. In the case of statistical physics systems, this may appear as circular reasoning as many (independent) samples are needed to produce many (independent) samples. Nonetheless, it has been shown that knowledge of the analytical form of the target distribution can be used to enhance the training and to compensate for a bias resulting from the limited expressiveness of the machine learning model. These matters have been addressed by Noé and co-workers,[10] who have used, for the first time, normalizing flows to sample equilibrium configurations of statistical mechanics systems. Normalizing flows are a class of generative models particularly well suited to generate physical configurations, as discussed in more detail below. In recent years and following the seminal work of Noé and co-workers, these concepts have been used to study different systems and have produced many interesting methodological developments.[11–14] Applications include lattice field theory,[15–19] free-energy estimation,[20–22] rare event[23,24] and enhanced sampling,[25,26] likelihood estimation in gravitational waves,[27] non-equilibrium,[28] and glassy systems,[29] as well as the calculation of coexistence lines in the isobaric–isothermal ensemble.[30–32] The principle at the basis of these models, also called Boltzmann generators,[33] is to train a network to transform samples from an elementary, easy-to-sample, "prior" distribution, such as a multivariate Gaussian or uniform distribution, into a more complex and physically interesting distribution such as the Boltzmann distribution.

In the case of condensed-matter systems, the concept of an easy-to-sample prior distribution has been stretched to something more complex than uniform or Gaussian. These distributions are sometimes too different from the target one, and training a flow to transform samples between the source and target system can be too hard a challenge, in particular for high numbers of particles. A different approach has been pursued, for example, by Wirnsberger and co-workers[20,21] and by Ahmad and co-workers,[22] where the authors chose a prior that is closer to the desired target physical distribution. For example, a network has been trained to transform a superposition of Gaussians placed at lattice sites of a particular crystal structure into a physical distribution of a crystalline solid. In a more recent example,[32] a specific point in the phase diagram of a liquid system was chosen as the prior distribution for a conditional flow to generate configurations in the whole phase diagram. The encoding of physical information in the prior (the lattice structure, or the target system itself for specific thermodynamic parameters) allows the training of the network to be performed without samples from the target distribution using only its potential energy function.

This idea can be further developed noting that, also among Boltzmann distributions, the concepts of easy-to-sample and hard-to-sample distributions are meaningful and can be connected to the computational cost required to sample configurations via standard methods. One obvious example is the different accuracy of the underlying theory used to represent interactions: Sampling the same system using a force-field simulation or an *ab initio* method requires very different computational resources and can produce very different results. This different demand can classify the former distribution as easier-to-sample compared to the latter. However, even among interactions modeled at the same level of accuracy, the presence or absence of certain features can have strong consequences on the accuracy and computational cost of the simulation. The Weeks–Chandler–Andersen (WCA) and the Lennard-Jones potentials are an interesting example in this sense: Even if they are similar in many aspects, the lack or presence of an attractive region entails some differences in the behavior of the systems in the fluid phase. For example, the WCA phase diagram does not have critical points and, below a certain density, admits only a single homogeneous fluid phase.[34] On the other hand, the lower cutoff of WCA potential compared to the one routinely used for the Lennard-Jones makes the former faster to compute than the latter.

In this paper, we build on these ideas, and we show that an invertible transformation can be learned by a neural network to transform one Boltzmann distribution into another, different, Boltzmann distribution. We also show that the generated configurations can be successfully reweighted to obtain unbiased samples. The approach is successfully tested on a two-dimensional 32-particle system under periodic boundary conditions modeled with two different forms of potential energies at different temperatures and densities in the liquid part of the phase diagram. These calculations indicate that a Boltzmann-to-Boltzmann approach can be applied to disordered phases of matter of systems described by different forms of interactions.

II. THEORETICAL FRAMEWORK

We consider two systems A and B, whose configurations x_A and x_B are distributed according to the probability densities $\rho_A(x_A)$ and $\rho_B(x_B)$. We assume that there exists an invertible mapping Φ that transforms configurations of one system into the ones of the other and vice versa, i.e., $x_B = \Phi(x_A)$ and $x_A = \Phi^{-1}(x_B)$. The direction of the map is arbitrary, and our choice is done without loss of generality. Under the application of Φ, the probability distributions for systems A and B are related by

$$\rho_B(x_B) = \rho_A(x_A)|\det \mathcal{J}(x_A)|^{-1}, \tag{1a}$$

$$\rho_A(x_A) = \rho_B(x_B)|\det \mathcal{J}^{-1}(x_B)|^{-1}, \tag{1b}$$

where $\mathcal{J}$ is the Jacobian matrix of Φ and $\mathcal{J}^{-1}$ is that of the inverse mapping, so that we have $\mathcal{J}^{-1}(x) \equiv \mathcal{J}\big(\Phi^{-1}(x)\big)$.

The analytical form of the transformation Φ is unknown in general. However, an approximation F can be found by means of a machine-learning approach. A class of deep neural networks well suited to this task are normalizing flows,[35] a particular kind of generative models characterized by the invertibility of the transformation and by the ability of transforming probability distributions. The application of the machine-learned transformation F and its inverse F^{-1} to samples from A and B gives the relations

$$P_B(x_B) = \rho_A(x_A)|\det J(x_A)|^{-1}, \tag{2a}$$

$$P_A(x_A) = \rho_B(x_B)|\det J^{-1}(x_B)|^{-1}, \tag{2b}$$

where J and J^{-1} are the Jacobians associated to F and F^{-1}, respectively, and P_B and P_A are the generated probability distributions, which may differ from the *true* probability distributions ρ_B and ρ_A.[59]

Equation (2) enables the definition of a "distance" between the target and the generated distributions from which a loss function for the training process can be derived. The Kullback–Leibler (KL) divergence acts as such a statistical "distance": Given two distributions $f(x)$ and $g(x)$, their KL divergence is given by

$$\mathrm{KL}(g\|f) = \int \mathrm{d}x\, g(x)[\log g(x) - \log f(x)].$$

Different metrics are possible and have been suggested, among others: the symmetrized KL divergence,[10] the Wasserstein distance,[36] and the optimized KL divergence for data-free training,[37] but here we consider only the KL-divergence, the state of the art choice for normalizing flows in statistical mechanics.

We first consider the direct transformation F so that samples x_A are transformed to x_B. The KL divergence between the target and the generated distribution is given by

$$\mathrm{KL}(\rho_B\|P_B) = \int \mathrm{d}x_B \rho_B(x_B)[\log \rho_B(x_B) - \log P_B(x_B)]. \tag{3}$$

Using Eq. (2), we know how the map F acts on the original samples from the source distribution ρ_A, and we can use this information to write Eq. (3) as

$$\mathrm{KL}(\rho_B \| P_B) = \int \mathrm{d}x_B \rho_B(x_B) \log \rho_B(x_B) + \\ - \int \mathrm{d}x_B \rho_B(x_B) \log \left[\rho_\mathrm{A}\left(F^{-1}(x_\mathrm{B})\right) |\det J^{-1}(x_\mathrm{B})|\right],$$

where we used the property that $|\det J(x_\mathrm{A})|^{-1} = |\det J^{-1}(x_\mathrm{B})|$. The first term of the RHS is the negative entropy of the system B and does not depend on the transformation F (nor on its inverse). Therefore, for the purpose of the definition of a training loss function, it is irrelevant. The second term of the RHS can be recast in the form of an average over the configurations sampled from system B. Therefore, we can define the loss function for the direct transformation as

$$L_\mathrm{NLL} = -\left\langle \log \rho_\mathrm{A}\left(F^{-1}(x_\mathrm{B})\right) + \log|\det J^{-1}(x_\mathrm{B})| \right\rangle_\mathrm{B}, \tag{4}$$

where NLL stands for negative log likelihood. Applying the same line of reasoning for the inverse transformation, i.e., starting from $\mathrm{KL}(\rho_A \| P_A)$, we obtain the loss function

$$L_\mathrm{KLD} = -\langle \log \rho_\mathrm{B}(F(x_\mathrm{A})) + \log|\det J(x_\mathrm{A})| \rangle_\mathrm{A}, \tag{5}$$

where KLD stands for Kullback–Leibler divergence. The use of the nomenclature L_NLL (sometimes called maximum likelihood or reverse KL divergence) and L_KLD (sometimes called forward KL divergence) is done in analogy with previous studies on normalizing flows.[37–39] Note that in this framework, these two losses are completely equivalent. The total loss function for the training of the network is $L = \lambda_\mathrm{NLL} L_\mathrm{NLL} + \lambda_\mathrm{KLD} L_\mathrm{KLD}$, where λ_NLL and λ_KLD are tunable parameters used to focus the training on the direct transformation or on its inverse.

The definition of the loss function through Eqs. (4) and (5) highlights the crucial role that the Jacobian of the transformation J has in the training process. An efficient implementation of a computational solution for the mapping from system A to B and vice versa inevitably involves an efficient implementation of the computation of $\log|\det J(x)|$ and its inverse. This can be achieved with neural networks based on a split-coupling flow architecture.[40,41]

In this paper, we focus on molecular systems in the canonical ensemble for which the ground truth distributions ρ_A and ρ_B have the form of the Boltzmann distribution,

$$\rho(x) \propto \exp\left[-\beta U(x)\right],$$

where U is the potential energy function of the particular system and β is the inverse temperature $\beta^{-1} = k_\mathrm{B} T$. It is then interesting to look back at the loss functions derived in Eqs. (4) and (5), which, after substitution of the particular form of ρ, become

$$L_\mathrm{NLL} = \left\langle \beta_\mathrm{A} U_\mathrm{A}\left(F^{-1}(x_\mathrm{B})\right) - \log|\det J^{-1}(x_\mathrm{B})| \right\rangle_\mathrm{B},$$
$$L_\mathrm{KLD} = \langle \beta_\mathrm{B} U_\mathrm{B}(F(x_\mathrm{A})) - \log|\det J(x_\mathrm{A})| \rangle_\mathrm{A},$$

where U_A and U_B are the potential energy functions of the systems A and B at inverse temperatures β_A and β_B, respectively. The logarithms of the partition functions arising from $\log \rho_\mathrm{A}$ and $\log \rho_\mathrm{B}$ are constant with respect to the network parameters and are thus irrelevant for the definition of the training losses.

For the number of degrees of freedom that are relevant for statistical physics systems, even the most powerful network can be subject to a bias in the generated distribution, regardless of the length and the accuracy of the training process. This bias results from the difference between the exact transformation Φ and the machine-learned one F. A systematic procedure to account for this difference becomes particularly important, for example, when no reference data are available and it is not possible to assess the accuracy of the results produced by the network. It is therefore crucial, if we aim to use the generated distributions to compute physical observables, to take into account the bias of the generated distribution and to compensate for it when computing averages. Indeed, given a set of N configurations $\{x^i\}_{i=1}^N$ sampled according to the probability distribution $\rho(x)$, it is possible to approximate the expected value of an observable $\langle O \rangle = \int \mathrm{d}x \rho(x) O(x)$ as

$$\langle O \rangle \approx \frac{1}{N} \sum_{i=1}^{N} O(x^i), \tag{6}$$

where $O(x)$ is the microscopic estimator of the macroscopic observable. However, Eq. (6) no longer yields the correct average if the configurations $\{x^i\}$ are sampled according to a probability $P(x) \neq \rho(x)$. In this case, it is necessary to account for the difference in the relative weights of the sampled configurations. An estimate for the average $\langle O \rangle$ is then given by

$$\langle O \rangle \approx \frac{\sum_{i=1}^N \omega(x^i) O(x^i)}{\sum_{i=1}^N \omega(x^i)}, \tag{7}$$

where $\omega(x^i)$ are the relative weights of the configurations x^i, given by[10]

$$\omega(x^i) = \frac{\rho(x^i)}{P(x^i)}. \tag{8}$$

This expression for the weights can be verified by writing down the ensemble average for the microscopic estimator $O(x)$ computed under the distribution $\rho(x)$ and the reweighted distribution $\omega(x)P(x)$,

$$\langle O \rangle = \int \mathrm{d}x\, \rho(x) O(x) = \int \mathrm{d}x\, \omega(x) P(x) O(x),$$

which is true for $\omega(x)$ given by Eq. (8). Here, we assume $P(x^i) \neq 0$, which is guaranteed by the normalizing flow architecture, as samples of nonzero probability in the source space are mapped to points of nonzero probability in the target space.

Knowledge of how the probability of a sample transforms under the application of the machine-learned map F makes it possible to give an explicit expression for the reweighting factors for the two systems A and B. For instance, considering Eq. (8) for system B and substituting the expression of the generated probability $P_\mathrm{B}(x)$ as written in Eq. (2a), we get

$$\omega_\mathrm{B}\left(F(x_\mathrm{A}^i)\right) = \frac{\rho_\mathrm{B}\left(F(x_\mathrm{A}^i)\right)}{\rho_\mathrm{A}(x_\mathrm{A}^i)|\det J(x_\mathrm{A}^i)|^{-1}}.$$

Introducing now the explicit expressions for the equilibrium distributions $\rho_\mathrm{A}(x)$ and $\rho_\mathrm{B}(x)$, we obtain the reweighting factor for averages computed using configurations transformed from A to B as

$$\omega_{\mathrm{B}}\big(F(x_{\mathrm{A}}^i)\big) \propto \exp\big[-\beta_{\mathrm{B}} U_{\mathrm{B}}\big(F(x_{\mathrm{A}}^i)\big) + \beta_{\mathrm{A}} U_{\mathrm{A}}(x_{\mathrm{A}}^i) + \log\big[|\det J(x_{\mathrm{A}}^i)|\big]\big]. \tag{9}$$

The same kind of approach yields the reweighting factor for averages computed using configurations transformed from B to A as

$$\omega_{\mathrm{A}}\big(F^{-1}(x_{\mathrm{B}}^i)\big) \propto \exp\big[-\beta_{\mathrm{A}} U_{\mathrm{A}}\big(F^{-1}(x_{\mathrm{B}}^i)\big) + \beta_{\mathrm{B}} U_{\mathrm{B}}(x_{\mathrm{B}}^i) + \log\big[|\det J^{-1}(x_{\mathrm{B}}^i)|\big]\big]. \tag{10}$$

Note that the proportionality factors cancel out in Eq. (7). The weights ω can be used either for standard reweighting of the observables computed as averages over the configurations or to directly resample the configurations by existing resampling methods, such as jackknifing or bootstrapping.[42] Moreover, these weights have been shown to be connected to the free energy difference between the two systems and can be used for free energy estimation.[20] Finally, the weights ω provide a way to quantify the efficiency of the sampling procedure via the so-called Kish (relative) effective sample size[43] defined by

$$\mathrm{RESS} = \frac{1}{N}\frac{\left(\sum_{i=1}^{N}\omega_i\right)^2}{\sum_{i=1}^{N}\omega_i^2}, \tag{11}$$

where $\omega_i \equiv \omega(x^i)$. The RESS is a number between N^{-1} and 1, and it is a way of quantifying how well the configurations sampled from one distribution represent samples from another. It is 1 when all the configurations have the same weight, and it is close to N^{-1} when one configuration has a weight that is much larger than all the others.

III. MATERIALS AND METHODS

A. Model systems

We use a normalizing flow to sample a system of 32 Lennard-Jones disks transforming configurations obtained from a simulation using a WCA potential. The pair potentials are given by

$$U_{\mathrm{WCA}}(r) = \begin{cases} U_{\mathrm{LJ}}(r) + \varepsilon & \text{if } r < \sqrt[6]{2}\sigma, \\ 0, & \text{otherwise}, \end{cases} \tag{12a}$$

$$U_{\mathrm{LJ}}(r) = 4\varepsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right]. \tag{12b}$$

Reduced units (indicated by the asterisk) are used throughout the paper, with σ and ε of the Lennard-Jones system chosen as units of length and energy, respectively. Periodic boundary conditions are enforced in all directions, and a cutoff of $r_{\mathrm{cut}}^* = 2.5$ is employed for the Lennard-Jones interactions. The Lennard-Jones potential is also shifted by the value at the cutoff in order to avoid discontinuities in the potential energy function. Long-range corrections are used for the energy.

B. Training

To ensure a good exploration of the source space during training, new configurations are continuously sampled from the source system using MCMC starting from a pool of independent configurations, which gets updated as training goes on. This *on-the-fly* sampling ensures that just the right amount of samples from the source space is produced during training and helps avoid overfitting. The use of multiple parallel walkers improves efficiency and guarantees a better exploration of configurations of the source space. Configurations are represented in internal coordinates (a particle is always centered at $\boldsymbol{r}_0 = (0, 0)$, and positions of other particles are given relative to $\boldsymbol{r}_0$ in $-L/2, L/2$ in each dimension) to remove translation invariance. Training is performed in both directions, i.e., $\lambda_{\mathrm{NNL}} = 1$ and $\lambda_{\mathrm{KLD}} = 1$ using a training set of 9×10^4 target configurations obtained via Markov chain Monte Carlo (MCMC) (details in the supplementary material). This training set is also augmented during training, exploiting other symmetries of the system (details in the supplementary material). As periodic boundary conditions are enforced, the system is invariant under the octahedral symmetry group of transformations. Whenever a configuration from the training set is chosen for loss evaluation, a random particle is centered in $\boldsymbol{r}_0$, and one of the 8 (in two dimensions) octahedral transformations is randomly selected and applied. The combined use of *on-the-fly* sampling in source space and augmentation in target space makes the training process robust with respect to the number of learnable parameters of the network. No overfitting is observed for the training protocol presented here. The same is true for training protocols twice as long and for network sizes twice as big as the ones used to produce the results presented here (see the supplementary material for details about training and additional training runs). The network is always validated in both directions using 10^4 independent fixed configurations from the source and target system. On these configurations, we also monitor the evolution of the RESS along training.

C. Network architecture

The architecture of the network is the same as that introduced by Wirnsberger *et al.* in Ref. 20 and improved in Ref. 21. The input in internal coordinates is reduced by two degrees of freedom by removing the central reference particle and rescaled in $[-1, 1]$. Two of these normalizing layers at each side of the network allow to take into account automatically different volumes (i.e., densities) for source and target systems: whenever the energy must be computed, the central particle is added back, and particles are rescaled to match the correct density of the corresponding space. In the spirit of split coupling flows, the input is then divided between the two spatial components, and a transformation modeled by a periodic rational quadratic spline[44] is applied to one part of the input. A circular encoder converts the input into periodic functions and ensures compatibility with periodic boundary conditions. A circular shift ensures that particles at the boundary of the box do not remain fixed. The parameters of the spline are obtained from an equivariant transformer[45] whose input is the other spatial component (the one that is not transformed). The transformation is thus invertible, and the Jacobian is fast to compute. The architecture guarantees the network to be equivariant with respect to particle permutations. Further details are given in the supplementary material.

D. Computational efficiency

The training procedure is the most computationally expensive step due to the many energy evaluations that are required to compute the loss functions. Training the network for all the examples shown took around 12 h on a NVIDIA GeForce RTX 3080 Ti GPU. Details on training hyperparameters and metrics are given in the supplementary material. Once the training procedure is completed, the transformation between the two systems requires only a minimal computational overhead, mainly linked to the generation of the source configurations and the computation of the weights. Even though the energy of the target system still needs to be computed to obtain the reweighting factors [see Eqs. (9) and (10)], this needs to be done only for a set of independent configurations that have already been decorrelated in the source space. For the specific examples presented, transforming 5×10^4 configurations from system A to system B requires around 5 s on the same hardware used for training.

IV. RESULTS

The thermodynamic parameters for the source and target system are selected such that both the WCA and the Lennard-Jones systems are within the liquid part of the phase diagram (see Refs. 46 and 47 for the phase diagrams of the WCA and Lennard-Jones potentials in two dimensions). We first choose to sample configurations of the target system in the supercritical region of the phase diagram. Here, the attractive part of the Lennard-Jones interaction, which distinguishes it from the WCA potential, is expected to play a minor role. We set a reduced density of $\varrho^* = 0.735$ (corresponding to a box length of $L^* = 6.6$) with a reduced temperature $T^* = 1$ for the Lennard-Jones system (system IB). The prior (system IA) is set to be at the same density $(\varrho^* = 0.735)$ with a higher temperature $(T^* = 2)$ to help the exploration of configuration space and consequently the sampling process. Once the training is complete (see the supplementary material for details), configurations are sampled from system IA and transformed to system IB via the learned transformation F. The inverse transformation is also possible without the need for additional training. A reweighting procedure using the weights of Eq. (8) is applied to the computed averages to obtain correctly weighted physical observables. Error bars are estimated via bootstrapping.

The potential energy distributions and radial distribution functions of the system before and after the transformations in both directions (IA → IB and IB → IA) are shown in Fig. 1 for a sample of 5×10^5 configurations. In addition, the reweighted observables obtained as outlined above [Eqs. (9) and (10)] are shown in the same figure. As a reference, we use the observables computed using standard Markov chain Monte Carlo simulations. We observe that it is harder for the network to transform configurations from a more ordered system $(T^*_{\mathrm{IB}} = 1)$ into a more disordered one $(T^*_{\mathrm{IA}} = 2)$ and to reweight them. This is apparent looking at the potential energy distribution of the samples transformed from system IB to system IA (Fig. 1), where high energy configurations are underrepresented. Nevertheless, the reweighting is successful in correcting for this bias, and average values are in good agreement. The signal to noise ratio of the weights grows with an increasing number of samples. In the supplementary material, we also show the effect of the transformations F and F^{-1} and the corresponding resampling procedure over a subset of configurations.

The relative effective sample size is also computed for both directions before and after applying the learned transformations. The value reported is the average of the RESS obtained from the application of F and F^{-1} to 10 different sets of independent configurations, each consisting of 5×10^4 samples. Before training, for the transformation IA → IB, we have $\mathrm{RESS}_{\mathrm{id}}^{\mathrm{A}\to\mathrm{B}} = 7.15 \times 10^{-3}$, where the subscript "id" indicates the identity transformation. This means that a direct reweighting is not efficient since more than 100 samples must be generated to obtain a single effective sample. After training, we obtain $\mathrm{RESS}^{\mathrm{A}\to\mathrm{B}} = 4.04 \times 10^{-2}$, resulting in an almost sixfold increase, i.e., four samples out of one hundred are meaningful in target space. The same process for the inverse transformation

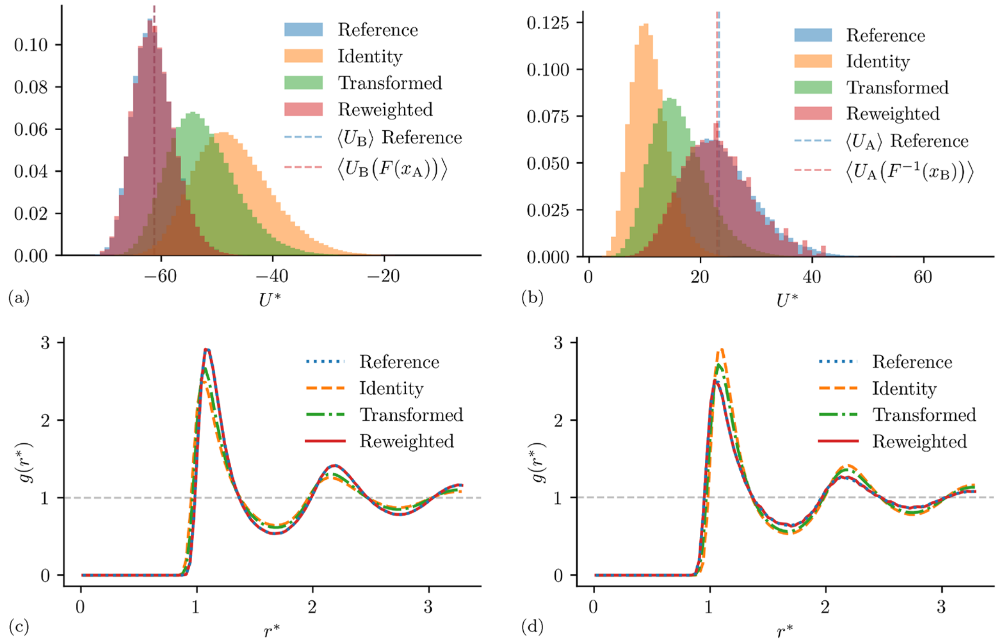

FIG. 1. Results obtained by transforming a two-dimensional system of WCA particles at $\varrho^* = 0.735$ and $T^* = 2$ (IA) into a system of Lennard-Jones particles at $\varrho^* = 0.735$ and $T^* = 1$ (IB). The orange dashed lines refer to the "identity" transformation, i.e., $F(x) = x$. The green dotted-dashed lines refer to the raw result of the transformations after training. The red solid lines are the reweighted results. The blue dotted lines are the reference data obtained via MCMC simulations. [(a) and (b)] Normalized histograms of potential energies of the configurations for the direct (IA → IB) and inverse (IB → IA) transformations, respectively. The dashed lines are the average potential energies computed using the reference configurations (blue) and the transformed (red) ones with the weights obtained via Eq. (8). [(c) and (d)] Radial distribution functions for the direct (IA → IB) and inverse (IB → IA) transformations, respectively.

IB → IA yields $\text{RESS}_{\text{id}}^{\text{B}\to\text{A}} = 9.32 \times 10^{-4}$ and $\text{RESS}^{\text{B}\to\text{A}} = 6.60 \times 10^{-3}$, corresponding to an increase of more than seven times. In all cases, statistical errors are around 20%. Note that the different results for the direct and inverse transformations are consistent with the potential energy distributions: the higher temperature of the source system helps the exploration of configuration space and makes it easier to reweight to a lower temperature.

In a second simulation, we explore the liquid subcritical region of the phase diagram of the Lennard-Jones system. At the same density ($\varrho^* = 0.735$), we set a temperature for the target system (system IIB) of $T^* = 0.5$. In this region of the phase diagram, the attractive part of the Lennard-Jones system is expected to play a more important role in the interactions. It is interesting to investigate if this behavior can be reproduced using normalizing flows with a prior that lacks such an attractive part. Configurations in source space (system IIA) are then sampled using a WCA potential at the same density $\varrho^* = 0.735$ and at a higher temperature of $T^* = 1$ with respect to the target space. We use the same training and generation protocol of the previous simulation. The potential energy distribution and radial distribution functions obtained by transforming samples from system IIA to system IIB via the learned transformation F are shown in Fig. 2. We note that the transformation slightly underrepresents the lowest energy configurations of the reference system, a phenomenon that was not observed in the supercritical region. A difference is also noted in the values of the RESS before and after the training. Before training, for the transformation IIA → IIB, we have $\text{RESS}_{\text{id}}^{\text{A}\to\text{B}} = 1.22 \times 10^{-3}$, while after training, we obtain $\text{RESS}^{\text{A}\to\text{B}} = 6.13 \times 10^{-3}$, resulting in a fivefold increase. The inverse transformation B → A yields $\text{RESS}_{\text{id}}^{\text{B}\to\text{A}} = 8.89 \times 10^{-4}$ and $\text{RESS}^{\text{B}\to\text{A}} = 3.86 \times 10^{-3}$, corresponding to an increase of about four times. We note that while the absolute value of the RESS after training in both directions is lower than what we obtain for the supercritical region, this is compensated by a lower starting value, which highlights a more profound difference between source and target systems in this region of the phase diagram. The transformation modeled by the neural network produces roughly the same increase in both the supercritical and subcritical liquid phases.

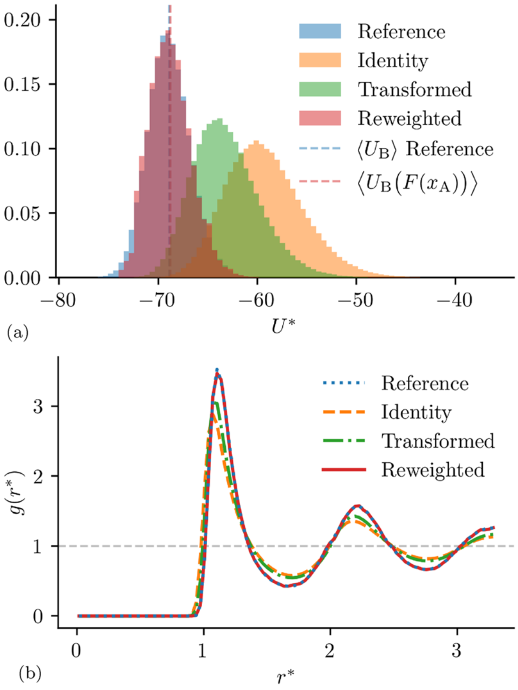

FIG. 2. Results obtained by transforming a two-dimensional system of WCA particles at $\varrho^* = 0.735$ and $T^* = 1$ (IIA) into a system of Lennard-Jones particles at $\varrho^* = 0.735$ and $T^* = 0.5$ (IIB). Labels are the same as in Fig. 1. (a) Normalized histogram of potential energies of the configurations for the direct (IIA → IIB) transformation. (b) Radial distribution functions for the direct (IIA → IIB) transformation.

Taking advantage of the more pronounced difference between the source and target space of this particular simulation, we provide a more detailed analysis of how the network models the transformation, examining in detail the mapping between corresponding configurations in source and target space. In Fig. 3, we show the highest weight configuration in target space generated by the flow, $x_{\text{B}}^{\text{max}}$, and compare it with the corresponding original in source space $x_{\text{A}}^{\text{max}} = F^{-1}(x_{\text{B}}^{\text{max}})$. The generated configuration with the highest weight in target space is the $x_{\text{B}}^{\text{max}}$ corresponding to the weight $\omega_{\text{B}}^{\text{max}} = \max_{\omega_{\text{B}}}\left\{\omega_{\text{B}}\left(F(x_{\text{A}}^{i})\right)\right\}$, where the absolute weight is obtained via Eq. (10). This configuration is then used as a reference to compute the relative weights $\bar{\omega}_{\text{B}} = \omega_{\text{B}}/\omega_{\text{B}}^{\text{max}}$ of all the others. The highest weight configuration is therefore the one that corresponds to a relative weight of 1. In the upper panel of Fig. 3, we plot $x_{\text{B}}^{\text{max}}$ and the one it generates from $x_{\text{A}}^{\text{max}}$ on top of each other (blue: original WCA configuration, source space; orange: Lennard-Jones transformed configuration, target space). In the lower panel, we highlight the transformation modeled by the flow by showing the vectors representing the displacements of the particles colored according to their magnitude.

Although it may seem that the effect of the network transformation is minimal, these small displacements are extremely efficient: In the particular case shown in Fig. 3, the relative weight of the transformed configuration increases by two orders of magnitude with respect to that of the original untransformed configuration. This improvement is also reflected in a lower energy after the transformation $U_{\text{LJ}}^*(F(x_{\text{A}}^{\text{max}})) = -70.6$ compared to its value before $U_{\text{LJ}}^*(\text{id}(x_{\text{A}}^{\text{max}})) = -67.7$. In general, there seems to be a direct relation between the weight of the generated configuration in target space and both the magnitude of the displacement of the particles and the difference in energy between the generated and original configurations [see also Figs. S7(a) and S7(b) in the supplementary material]. We observe that the configurations with the highest weights are on average the ones that have been moved the most and that have a higher difference in energy (in absolute value) with respect to the corresponding original configurations in source space.

While the latter is expected from the particular form of the loss function, we conjecture that the former is due to an easier solution of the many-body displacement problem for the specific configuration. Indeed, to minimize the loss, the transformation must move all particles in a concerted way, avoiding particle overlaps. This is an extremely hard problem to solve for condensed matter systems in presence of periodic boundary conditions. In situations where this problem is, by chance, easier to solve (e.g., a single local rearrangement relaxes a particle overlap), the transformation

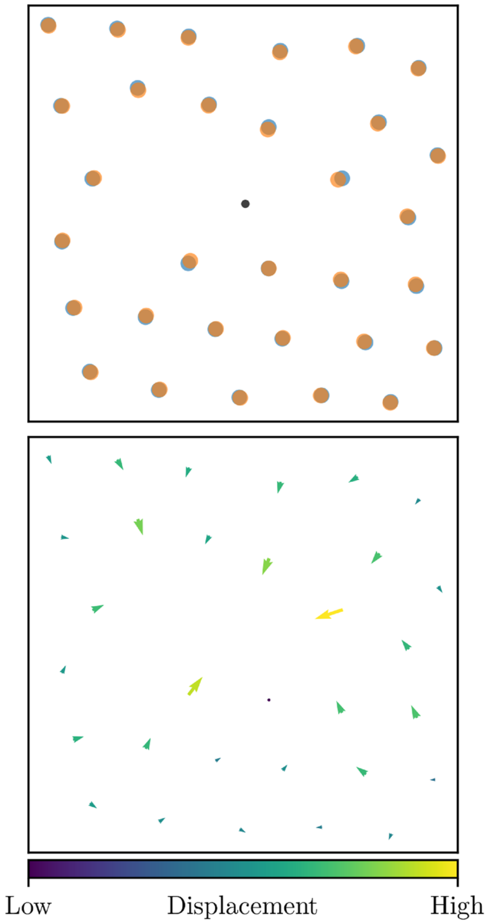

FIG. 3. Highest weight configuration in target space when transforming from WCA ($\varrho^* = 0.735$, $T^* = 1$) to Lennard-Jones ($\varrho^* = 0.735$, $T^* = 0.5$), i.e., simulation II. Upper panel: source configuration (blue) and transformed configuration (orange). The central particle (black) is not involved in the transformation. Lower panel: displacement from source to target configuration plotted on source configurations. The arrows are proportional to $d^i = x_B^i - x_A^i$, computed with the minimum image convention, and are colored according to $|d^i|$. The color scale is logarithmic and ranges from $4.7 \times 10^{-3}\sigma$ to $6.9 \times 10^{-2}\sigma$.

performs well in minimizing the loss by displacing particles and producing high weight (low energy) configurations. In the other cases, the optimal strategy seems to rely on making small adjustments to the particle positions, focusing more on avoiding energy explosions than to find low-energy structures. This is also compatible with the unfavorable scaling of normalizing flows for liquid systems under periodic boundary conditions, as the solution of this problem becomes exponentially harder with increasing number of degrees of freedom. A quantitative discussion of this point goes beyond the scope of this paper and will be subject of future publications.

To uncover possible patterns in the inference process of the flow and to clarify further its working mechanism, we inspect the average displacement between 5×10^5 generated configurations and the corresponding source configurations. Figure 4 shows the average displacement between source and target configurations plotted as a function of the position of the particle with respect to the central particle in the source configuration. In particular, we show in the top panel the average displacement in Cartesian coordinates, while in the bottom panel, we show the same quantity in polar coordinates after the polar angle has been integrated out (for brevity, we refer to this quantity as radial coordinate in Fig. 4). Figure 4(a) indicates how the transformation acts on the particle positions to reproduce

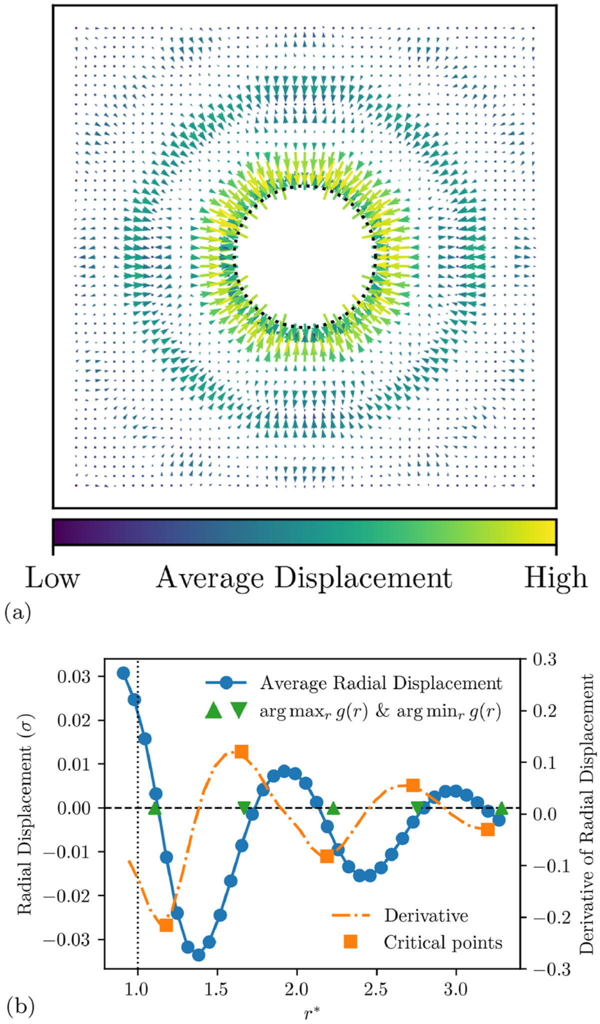

FIG. 4. Average displacement in Cartesian and radial coordinates between original and transformed configurations computed from a sample of 5×10^5 configurations: (a) Average displacement in Cartesian coordinates computed as in the lower panel of Fig. 3 for every generated configuration and averaged on a 50×50 grid. Arrows are colored according to their magnitude, and the color scale (logarithmic) ranges from $1.2 \times 10^{-2}\sigma$ to $4.0 \times 10^{-2}\sigma$. The dashed line indicates the 1σ radius. (b) Average displacement (blue) and its derivative (orange) in radial coordinate. The green triangles indicate maxima (up) and minima (down) of the g(r) in target space [see Fig. 2(b)]. The orange squares are the critical points of the derivative of the average displacement in radial coordinate.

the particle density of the target system. In particular, the peak corresponding to the first shell is steepened by moving particles that are close to the center of the box outward and pushing inward particles that are slightly farther from the center of the box than the main peak. The same pattern can be observed on the second and third shells but with a smaller magnitude. In this case, the inward pushing is more pronounced than the outward push, which is still visible on some of the grid points between the solvation shells. The slight anisotropy apparent in Fig. 4(a) arises from the finite size effects caused by the small system size. In Fig. 4(b), this simulation artifact is removed by computing the average displacement in polar coordinates and integrating out the polar angle. The plot of the average displacement in this radial coordinate shows zeros in correspondence with the radial distribution functions maxima and minima, and the same correspondence is found with the points of maximum and minimum variation of the displacement (highlighted by the relative maxima and minima of the derivative of the radial displacement). This corroborates the qualitative description given above, and it is a direct consequence of the higher temperature of the source distribution compared to that of the target distribution (see also the generated configurations, Fig. S4 of the supplementary material). Note, however, that an exact correspondence between the displacement produced by the transformation and the density of particles in target space is not trivial, as it ultimately comes down to the minimization of the KL divergence between configuration densities in source and target space.

In view of these results, the question arises if the network learns to favor configurations that already have a high weight in source space. In other words: Does the network only increase the relative weights of configurations that are already "good" in target space before the transformation or does it also modify configurations to substantially increase their weights? To answer this question, in the left panel of Fig. 5, we show the logarithm of the relative weight of the transformed configurations sorted by their relative weights in target space (blue). These weights are compared to the relative weights in target space of the original configurations (light orange).

The results shown in Fig. 5 indicate that a weight that is already high before the transformation helps the network in producing good configurations, as expected. This follows from the similar trends observed for the relative weights of the transformed configurations and for the moving average computed over 10^3 samples (deep orange line) of the relative weights of the original configurations. Therefore, we conclude that generated high-weight configurations come, on average, from source configurations that already possess a high weight in target space. On the other hand, we also see that, for specific configurations, the network increases the value of the relative weight by many orders of magnitude. At the same time, there are also cases in which the relative weight of a configuration is decreased by the transformation. Nonetheless, in general, transformed configurations have higher relative weights on average than the original ones, as it can be appreciated from the probability density of the weights shown in the right panel of Fig. 5. This also shows how the effect of the network training is to transform source configurations to maximize their relative weights in target space, a result that can be mathematically proven.[10]

As a final point, it is interesting to investigate the effectiveness of the normalizing flow in improving the sampling efficiency under conditions where the source and target systems are closer to each other. This amounts to finding coordinates in the phase diagram where a direct reweighting of configurations between two systems could be already possible given a sufficient number of samples. Starting from configurations obtained using a WCA potential at

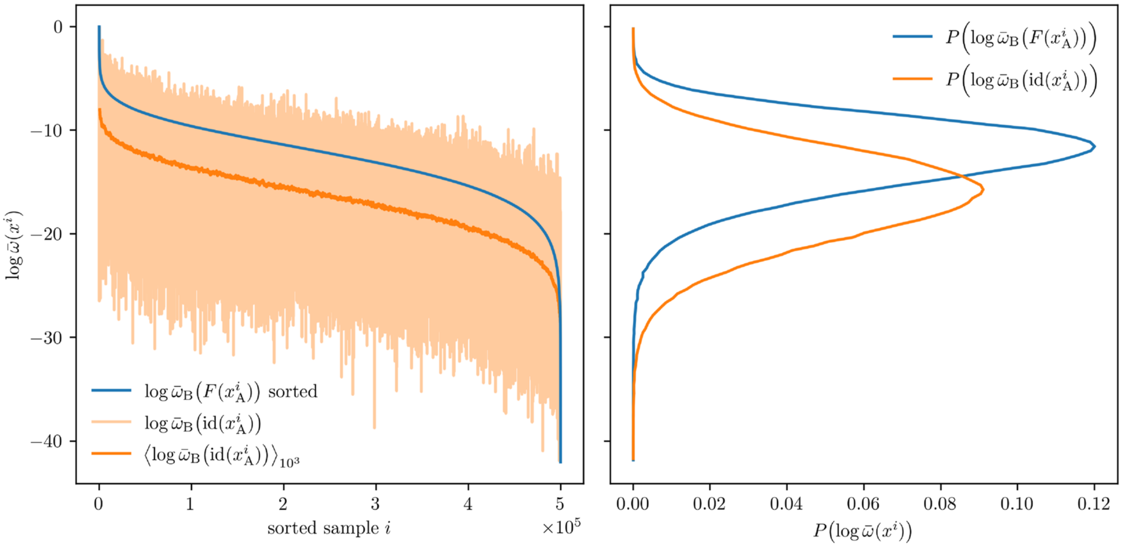

FIG. 5. Distribution of the logarithm of the relative weights of transformed and source configurations in target space. Left panel: logarithm of the relative weights in target space for each sample of transformed (sorted by weights, blue) and corresponding original configurations (light orange) with moving average performed over 10^3 samples (deep orange). Right panel: probability density of the logarithm of the relative weights over the samples. The shift of the average of the distribution toward higher relative weights is an effect of the training of the network, which maximizes the weights of the configurations in target space.[10]

$\varrho^* = 0.735$ and a temperature of $T^* = 1$ (system IIIA), we compute the RESS when attempting to transform directly (without any network) to a system of Lennard-Jones particles in the range of densities and temperatures given by $\varrho^* \in [0.6, 0.75]$ and $T^* \in [0.6, 1]$. Note that, at difference with the previous simulations, we now also take into account the possibility of transforming between thermodynamic coordinates at different densities. The weights associated to the configurations obtained in system A when averages are computed in system B are then given by

$$\omega_\mathrm{B}(sx_\mathrm{A}^i) \propto \exp\left[-\beta_\mathrm{B} U_\mathrm{B}(sx_\mathrm{A}^i) + \beta_\mathrm{A} U_\mathrm{A}(x_\mathrm{A}^i) + k\right], \tag{13}$$

where the scaling parameter s, for a system in d spatial dimensions, can be computed as

$$s = \left(\frac{\varrho_\mathrm{B}}{\varrho_\mathrm{A}}\right)^{\frac{1}{d}} \tag{14}$$

and k is the constant term arising from the Jacobian of the scaling transformation, i.e., $k = dN\log(s)$. Note that this is equivalent to Eq. (9) with an isotropic scaling in place of the learned transformation F. Also note that the term e^k in Eq. (13) is constant, and it simplifies in Eqs. (7) and (11). The result of direct reweighting in the range discussed above is shown in Fig. 6. The figure shows that, just by scaling the coordinates without any network-based transformation, there is a wide range of thermodynamic parameters where the RESS is considerable. At constant density or temperature, the RESS quickly drops to zero, while varying both parameters at the same time, it is possible to slow down the decay of the RESS. This is compatible with the physical intuition that to obtain the same distribution of configurations for certain systems in the liquid phase, one can either increase the temperature and compress the system or expand the system and decrease the temperature. This concept has been formalized rigorously in the framework of isomorphs[48] for interaction potentials that are perfect inverse power laws and with good approximation for the Lennard-Jones potential. For this study, given that we are considering two different interaction forms in source and target systems, we just observe that there is a diagonal line $\varrho_\mathrm{max}(T) = \arg\max_\varrho[\mathrm{RESS}(\varrho, T)]$, where the RESS is maximum for each temperature, as shown in Fig. 6.

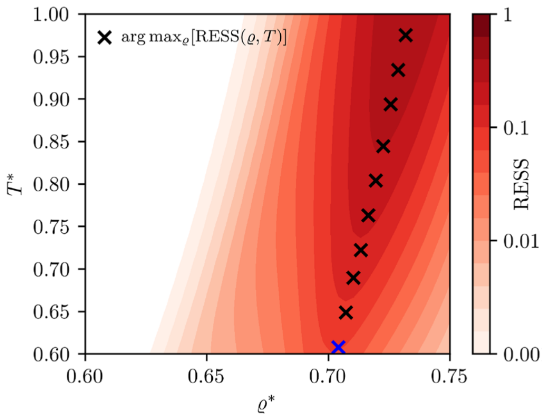

FIG. 6. Direct reweighting of 2000 configurations obtained from a simulation of particles interacting via a WCA potential at $\varrho^* = 0.735$ and $T^* = 1$ to a 50×50 grid of thermodynamic parameters given by $\varrho^* \in [0.6, 0.75]$ and $T^* \in [0.6, 1]$. Weights associated to each configuration are computed via Eq. (13), and RESS is computed via Eq. (11). For a subset of $T^* \in [0.6, 1]$, the corresponding density that yields the highest RESS is computed. These points are shown as symbols in the figure. In blue, we mark the point used as system IIIB.

We choose a point along this line, in particular $\varrho^* = 0.704$, $T^* = 0.608$ (system IIIB), to check how well the network performs in situations where the starting point has already a high RESS for the identity transformation [note that for the rest of the paper, whenever different densities are involved between source and target systems, we refer to the identity transformation as the pure scaling with the factor s given by Eq. (14)]. We again use the same training and sampling protocols as the two previous simulations. We show in Fig. 7 the potential energy distribution and radial distribution functions obtained after training the flow to transform between system IIIA and system IIIB. We observe in this case how the potential energy distribution and the radial distribution functions are close to the reference already before training of the transformation F starts. The training, on the other hand, does not improve significantly the observables computed. This can be noted from the almost perfect superposition between the result of the identity transformation (scaling only) and the transformed observables. In terms of the

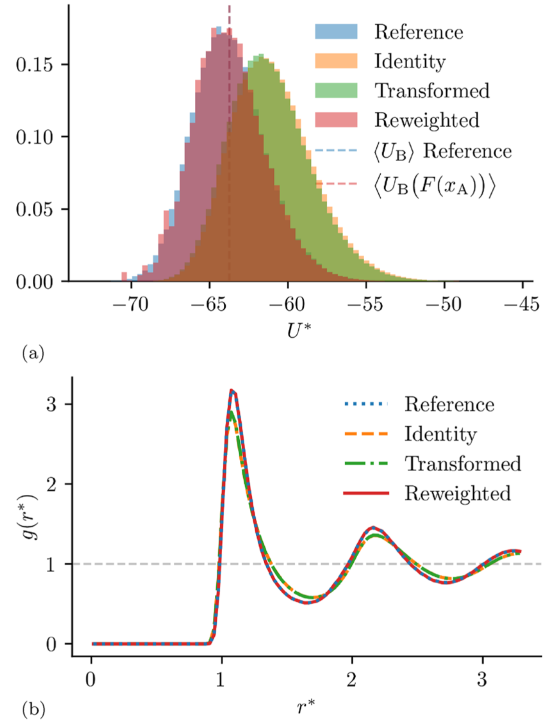

FIG. 7. Results obtained transforming a two-dimensional system of WCA particles at $\varrho^* = 0.735$ and $T^* = 1$ (IIIA) into a system of Lennard-Jones particles at $\varrho^* = 0.704$ and $T^* = 0.608$ (IIIB). Labels are the same as in Fig. 1. Note that the identity transformation includes, in this case, an isotropic scaling computed via Eq. (14). (a) Normalized histogram of potential energies of the configurations for the direct (IIIA → IIIB) transformation. (b) Radial distribution functions for the direct (IIIA → IIIB) transformation.

TABLE I. Summary of the thermodynamic parameters of source and target systems analyzed in this paper. RESS of the identity transformation and the trained transformation, together with their ratios, is also reported for both directions.

	ϱ_A^*	T_A^*	ϱ_B^*	T_B^*	$\text{RESS}_{\text{id}}^{A\to B}$	$\text{RESS}^{A\to B}$	Ratio	$\text{RESS}_{\text{id}}^{B\to A}$	$\text{RESS}^{B\to A}$	Ratio
I	0.735	2	0.735	1	7.15×10^{-3}	4.04×10^{-2}	6	9.32×10^{-4}	6.60×10^{-3}	7
II	0.735	1	0.735	0.5	1.22×10^{-3}	6.13×10^{-3}	5	8.89×10^{-4}	3.86×10^{-3}	4
III	0.735	1	0.704	0.608	2.44×10^{-2}	2.62×10^{-2}	1	9.08×10^{-2}	9.27×10^{-2}	1

calculation of the RESS, we come to the same conclusion: Before training, for the transformation IIIA → IIIB, we have $\text{RESS}_{\text{id}}^{A\to B} = 2.44 \times 10^{-2}$, while after training, we obtain $\text{RESS}^{A\to B} = 2.62 \times 10^{-2}$, resulting in a very modest increase. For the inverse transformation B → A, we have $\text{RESS}_{\text{id}}^{B\to A} = 9.08 \times 10^{-2}$ and $\text{RESS}^{B\to A} = 9.27 \times 10^{-2}$, again resulting in an almost unit ratio. While for these thermodynamic parameters, the value of the RESS after training is the highest among the simulations shown, we see how this result is mostly entirely due to the application of the isotropic scaling factor to the coordinates of the source system. The ratio between the RESS before and after the application of the network-based transformation shows a very modest increase, the lowest among the simulations presented. In Table I, we show a summary of the RESS obtained for the various transformations.

V. CONCLUSION

The results presented in this paper show that normalizing flows can be used to generate independent configurations of liquid systems in the *NVT* ensemble. This approach allows to decorrelate samples in source space using a relatively inexpensive model and then to transform them to configurations distributed according to the probability distribution of the target space using a normalizing flow. This is made possible by the use of another physical system as a prior for the network, therefore encoding a high degree of physical information in the problem already before the training starts. It becomes then simpler and faster for the model to produce a good approximation of the desired distribution after a relatively inexpensive training process. A reweighting procedure ensures that averages computed over the generated configurations are free from any residual bias from the distribution of the source space. An increase in the relative effective sample size compared to the identity transformation is observed to be particularly pronounced along certain directions on the $\rho - T$ phase diagram of Lennard-Jones particles.[60] We note, however, that the increase in relative effective sample size is modest, and for larger systems, it would not compensate for the number of energy evaluations necessary for training the model, also considering the exponential dependence of the RESS with respect to the number of degrees of freedom of source and target systems.

Nevertheless, the numerical simulations presented here show that the network can be trained to generate the correct energy and particle distributions of fluid systems in the case of different functional forms for the interaction potential between the source and target systems. In particular, the reweighted radial distribution function indicates that the structure of the liquid for the two different systems can be well reproduced by the normalizing flow. The use of different interaction models in the source and target system paves the way for the exploitation of normalizing flows in transforming configurations obtained using computationally inexpensive models to more precise but computationally expensive descriptions, as already attempted in different studies.[49–51] For example, a first step in this direction would be the transformation of configurations sampled through coarse-grained or all-atom force-field potential to *first-principles* simulations, drastically reducing, in this way, the time needed for the numerical study of liquid systems at *ab initio* accuracy. In this case, decorrelation of configurations in the target (expensive and fine-grained) system could be achieved at the numerical cost of decorrelating samples in the source (cheap and coarse-grained) system. Note that, for such a study, generation of the training set, the training procedure, and the reweighting step still represent formidable problems to solve due to the high number of energy evaluations in target space necessary in these steps. While for the first and second problems, data-free training[21] and the use of neural network potentials (a first approach in this direction has been attempted in Ref. 50 with promising results) could mitigate the computational cost, for the reweighting procedure, there is no trivial solution to the extensive scaling with system size. When the transformation learned by the network produces a distribution that is very close to the target distribution, a possibility could be to skip the reweighting step, avoiding, in this way, the additional computational burden associated with it. This possibility has been investigated and used when generating solid systems,[21] which have been shown to be an easier problem to solve for generative models such as normalizing flows. Still, skipping the reweighting step is subject to doubts about the statistical ensemble and, more in general, on the probability distribution sampled by the flow, in addition to questioning the use of an exact likelihood model (such as normalizing flows) to solve the sampling problem. Lifting the necessity of training in both directions with data-free training and the requirement of reweighting open up the possibility of using more flexible, non-invertible models (such as diffusion models) to tackle the sampling problem.[52]

In conclusion, while normalizing flows have shown promising results particularly for the simulation of solids, their application to liquids still suffers from low efficiency. Conditional flows, as implemented in Ref. 32 in the isobaric–isothermal ensemble, surely represent a significant improvement for the generation of liquid systems at different thermodynamic parameters. An implementation in the context of the canonical ensemble does not seem to pose any particular additional difficulty. It would be interesting to investigate how this kind of flow performs for the model system discussed in this paper and, in particular, in regions of the phase diagram where the initial sampling efficiency is already high for the identity transformation (for example, along the line in Fig. 6). A different and also promising perspective in the long term could be represented by the latest development in the field of flow-matching,[53–55] diffusion models,[56,57] and their combinations.[58] For the time being, 60 years after the seminal paper from Rahman,[3] molecular dynamics still seems to be the best tool available to scientists to get insights into the equilibrium properties of liquid systems.

SUPPLEMENTARY MATERIAL

The supplementary material contains details on the network architecture and explains how to handle symmetries during training and inference, the training dataset, protocols and metrics, and additional results obtained with different hyperparameters for training. It also contains a qualitative analysis of the relation between high-weight generated configurations and the difference in energy and positions between source and generated configurations.

ACKNOWLEDGMENTS

We acknowledge the financial support of the Austrian Science Fund (FWF) through the SFB TACO, Grant No. 10.55776/F8100. A.C. thanks Benjamin Rotenberg for insightful discussions.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Alessandro Coretti: Conceptualization (equal); Data curation (equal); Investigation (equal); Methodology (equal); Software (equal); Validation (equal); Writing – original draft (equal); Writing – review & editing (equal). **Sebastian Falkner**: Conceptualization (equal); Data curation (equal); Investigation (equal); Methodology (equal); Software (equal); Validation (equal); Writing – original draft (equal); Writing – review & editing (equal). **Phillip L. Geissler**: Conceptualization (equal); Methodology (equal); Supervision (equal). **Christoph Dellago**: Conceptualization (equal); Funding acquisition (equal); Methodology (equal); Supervision (equal); Validation (equal); Writing – original draft (equal); Writing – review & editing (equal).

DATA AVAILABILITY

The data that support the findings of this study together with the source code used for the numerical simulations are available via the project GitHub page CompPhysVienna/paper_boltzmann2boltzmann.

REFERENCES

[1] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," J. Chem. Phys. **21**(6), 1087–1092 (1953).

[2] B. J. Alder and T. E. Wainwright, "Phase transition for a hard sphere system," J. Chem. Phys. **27**(5), 1208–1209 (1957).

[3] A. Rahman, "Correlations in the motion of atoms in liquid argon," Phys. Rev. **136**(2A), A405–A411 (1964).

[4] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, in *Generative Adversarial Nets, Advances in Neural Information Processing Systems Vol. 27*, edited by Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger (Curran Associates, Inc., 2014).

[5] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in 2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14–16, 2014, Conference Track Proceedings, 2013.

[6] G. Papamakarios, E. Nalisnick, D. J. Rezende, S. Mohamed, and B. Lakshminarayanan, "Normalizing flows for probabilistic modeling and inference," J. Mach. Learn. Res. **22**(57), 1–64 (2021).

[7] R. Gómez-Bombarelli, J. N. Wei, D. Duvenaud, J. M. Hernández-Lobato, B. Sánchez-Lengeling, D. Sheberla, J. Aguilera-Iparraguirre, T. D. Hirzel, R. P. Adams, and A. Aspuru-Guzik, "Automatic chemical design using a data-driven continuous representation of molecules," ACS Cent. Sci. **4**(2), 268–276 (2018).

[8] M. E. Tuckerman, "Machine learning transforms how microstates are sampled," Science **365**(6457), 982–983 (2019).

[9] D. Wu, L. Wang, and P. Zhang, "Solving statistical mechanics using variational autoregressive networks," Phys. Rev. Lett. **122**, 080602 (2019).

[10] F. Noé, S. Olsson, J. Köhler, and H. Wu, "Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning," Science **365**(6457), eaaw1147 (2019).

[11] K. A. Nicoli, S. Nakajima, N. Strodthoff, W. Samek, K.-R. Müller, and P. Kessel, "Asymptotically unbiased estimation of physical observables with neural samplers," Phys. Rev. E **101**(2), 023304 (2020).

[12] H. Wu, J. Köhler, and F. Noé, "Stochastic normalizing flows," Adv. Neural Inf. Process. Syst. **33**, 5933–5944 (2020).

[13] M. Dibak, L. Klein, and F. Noé, "Temperature steerable flows and Boltzmann generators," arXiv:2108.01590 [cond-mat] (2021).

[14] J. Köhler, A. Krämer, and F. Noé, "Smooth normalizing flows," arXiv:2110.00351 [physics, stat] (2021).

[15] M. Albergo, G. Kanwar, and P. Shanahan, "Flow-based generative models for Markov chain Monte Carlo in lattice field theory," Phys. Rev. D **100**(3), 034515 (2019).

[16] K. A. Nicoli, C. J. Anders, L. Funcke, T. Hartung, K. Jansen, P. Kessel, S. Nakajima, and P. Stornati, "Estimation of thermodynamic observables in lattice field theories with deep generative models," Phys. Rev. Lett. **126**(3), 032001 (2021).

[17] E. Cellini, M. Caselle, A. Nada, and M. Panero, "Stochastic normalizing flows for lattice field theory," arXiv:2210.03139 (2022).

[18] M. Gabrié, G. M. Rotskoff, and E. Vanden-Eijnden, "Adaptive Monte Carlo augmented with normalizing flows," Proc. Natl. Acad. Sci. U. S. A. **119**(10), e2109420119 (2022).

[19] A. Singha, D. Chakrabarti, and V. Arora, "Conditional normalizing flow for Markov chain Monte Carlo sampling in the critical region of lattice field theory," Phys. Rev. D **107**(1), 014512 (2023).

[20] P. Wirnsberger, A. J. Ballard, G. Papamakarios, S. Abercrombie, S. Racanière, A. Pritzel, D. Jimenez Rezende, and C. Blundell, "Targeted free energy estimation via learned mappings," J. Chem. Phys. **153**(14), 144112 (2020); arXiv:2002.04913.

[21] P. Wirnsberger, P. George, B. Ibarz, S. Racanière, A. J. Ballard, A. Pritzel, and C. Blundell, "Normalizing flows for atomic solids," arXiv:2111.08696 [cond-mat, physics:physics, stat] (2021).

[22] R. Ahmad and W. Cai, "Free energy calculation of crystalline solids using normalizing flows," Modell. Simul. Mater. Sci. Eng. **30**(6), 065007 (2022).

[23] S. Falkner, A. Coretti, S. Romano, P. L. Geissler, and C. Dellago, "Conditioning Boltzmann generators for rare event sampling," Mach. Learn.: Sci. Technol. **4**(3), 035050 (2023).

[24] S. Asghar, Q.-X. Pei, G. Volpe, and R. Ni, "Efficient rare event sampling with unsupervised normalizing flows," Nat. Mach. Intell. **6**, 1370–1381 (2024).

[25] M. Invernizzi, A. Krämer, C. Clementi, and F. Noé, "Skipping the replica exchange ladder with normalizing flows," J. Phys. Chem. Lett. **13**(50), 11643–11649 (2022).

[26] S. Tamagnone, A. Laio, and M. Gabrié, "Coarse-grained molecular dynamics with normalizing flows," J. Chem. Theory Comput. **20**(18), 7796–7805 (2024).

[27] M. J. Williams, J. Veitch, and C. Messenger, "Nested sampling with normalizing flows for gravitational-wave inference," Phys. Rev. D **103**(10), 103006 (2021).

[28] M. Caselle, E. Cellini, A. Nada, and M. Panero, "Stochastic normalizing flows as non-equilibrium transformations," J. High Energy Phys. **2022**(7), 15.

[29] G. Jung, G. Biroli, and L. Berthier, "Normalizing flows as an enhanced sampling method for atomistic supercooled liquids," Mach. Learn.: Sci. Technol. **5**(3), 035053 (2024).

[30] P. Wirnsberger, B. Ibarz, and G. Papamakarios, "Estimating Gibbs free energies via isobaric-isothermal flows," Mach. Learn.: Sci. Technol. **4**(3), 035039 (2023).

[31] S. van Leeuwen, A. P. de Alba Ortíz, and M. Dijkstra, "A Boltzmann generator for the isobaric-isothermal ensemble," arXiv:2305.08483 (2023).

[32] M. Schebek, M. Invernizzi, F. Noé, and J. Rogal, "Efficient mapping of phase diagrams with conditional Boltzmann generators," Mach. Learn.: Sci. Technol. **5**(4), 045045 (2024).

[33] A. Coretti, S. Falkner, J. Weinreich, C. Dellago, and O. Anatole von Lilienfeld, "Boltzmann generators and the new Frontier of computational sampling in many-body systems," arXiv:2404.16566 (2024).

[34] S. S. Khali, D. Chakraborty, and D. Chaudhuri, "Two-step melting of the Weeks–Chandler–Anderson system in two dimensions," Soft Matter **17**(12), 3473–3485 (2021).

[35] D. Rezende and S. Mohamed, "Variational inference with normalizing flows," in *Proceedings of the 32nd International Conference on Machine Learning, Volume 37 of Proceedings of Machine Learning Research*, edited by F. Bach and D. Blei (PMLR, Lille, France, 2015), pp. 1530–1538.

[36] F. Coeurdoux, N. Dobigeon, and P. Chainais, "Sliced-wasserstein normalizing flows: Beyond maximum likelihood training," arXiv:2207.05468 (2022).

[37] L. Felardos, J. Hénin, and G. Charpiat, "Designing losses for data-free training of normalizing flows on Boltzmann distributions," arXiv:2301.05475 (2023).

[38] J. Mosiński, P. Biliński, T. Merritt, A. Ezzerg, and D. Korzekwa, "Ae-flow: Autoencoder normalizing flow," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (IEEE, 2023), pp. 1–5.

[39] M. Hassanaly, B. A. Perry, M. E. Mueller, and S. Yellapantula, "Uniform-in-phase-space data selection with iterative normalizing flows," Data-Centric Eng. **4**, e11 (2023).

[40] L. Dinh, D. Krueger, and Y. Bengio, "NICE: Non-linear independent components estimation," arXiv:1410.8516 [cs] (2015).

[41] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using Real NVP," arXiv:1605.08803 [cs, stat] (2017).

[42] B. Efron, *The Jackknife, the Bootstrap and Other Resampling Plans* (SIAM, 1982).

[43] L. Kish, *Survey Sampling* (John Wiley & Sons, Nashville, TN, 1966).

[44] C. Durkan, A. Bekasov, I. Murray, and G. Papamakarios, "Neural spline flows," in *Advances in Neural Information Processing Systems*, edited by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (Curran Associates, Inc., 2019), Vol. 32, https://proceedings.neurips.cc/paper_files/paper/2019/file/7ac71d433f282034e088473244df8c02-Paper.pdf.

[45] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, edited by I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Curran Associates, Inc., 2017), Vol. 30, https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[46] F. Cuadros and A. Mulero, "A validity test of the WCA perturbation theory for two-dimensional Lennard-Jones fluids," Chem. Phys. **160**(3), 375–382 (1992).

[47] J. A. Barker, D. Henderson, and F. F. Abraham, "Phase diagram of the two-dimensional Lennard-Jones system; evidence for first-order transitions," Physica A **106**(1–2), 226–238 (1981).

[48] T. B. Schrøder, N. Gnan, U. R. Pedersen, N. P. Bailey, and J. C. Dyre, "Pressure-energy correlations in liquids. V. Isomorphs in generalized Lennard-Jones systems," J. Chem. Phys. **134**(16), 164505 (2011).

[49] A. Rizzi, P. Carloni, and M. Parrinello, "Free energies at qm accuracy from force fields via multimap targeted estimation," Proc. Natl. Acad. Sci. U. S. A. **120**(46), e2304308120 (2023).

[50] A. Molina-Taborda, P. Cossio, O. Lopez-Acevedo, and M. Gabrié, "Active learning of Boltzmann samplers and potential energies with quantum mechanical accuracy," J. Chem. Theory Comput. **20**(20), 8833–8843 (2024).

[51] A. H. Mahmoud, M. Masters, S. J. Lee, and M. A. Lill, "Accurate sampling of macromolecular conformations using adaptive deep learning and coarse-grained representation," J. Chem. Inf. Model. **62**(7), 1602–1617 (2022).

[52] G. M. Rotskoff, "Sampling thermodynamic ensembles of molecular systems with generative neural networks: Will integrating physics-based models close the generalization gap?," Curr. Opin. Solid State Mater. Sci. **30**, 101158 (2024).

[53] J. Köhler, Y. Chen, A. Krämer, C. Clementi, and F. Noé, "Flow-matching: Efficient coarse-graining of molecular dynamics without forces," J. Chem. Theory Comput. **19**(3), 942–952 (2023).

[54] L. Klein and F. Noé, "Transferable Boltzmann generators," arXiv:2406.14426 (2024).

[55] L. Klein, A. Krämer, and F. Noé, "Equivariant flow matching," in Advances in *Neural Information Processing Systems*, edited by A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (Curran Associates, Inc., 2023), Vol. 36, pp. 59886–59910, https://proceedings.neurips.cc/paper_files/paper/2023/file/bc827452450356f9f558f4e4568d553b-Paper-Conference.pdf.

[56] M. Arts, V. Garcia Satorras, C.-W. Huang, D. Zügner, M. Federici, C. Clementi, F. Noé, R. Pinsler, and R. van den Berg, "Two for one: Diffusion models and force fields for coarse-grained molecular dynamics," J. Chem. Theory Comput. **19**(18), 6151–6159 (2023).

[57] Y. Wang, L. Guo, H. Wu, and T. Zhou, "Energy based diffusion generator for efficient sampling of Boltzmann distributions," arXiv:2401.02080 (2024).

[58] N. M. Boffi, M. S. Albergo, and E. Vanden-Eijnden, "Flow map matching," arXiv:2406.07507 (2024).

[59] The mapping F and its Jacobian J, together with their inverse, are also functions of the network parameters θ as in, for example, $F(x) \equiv F(x;\ \theta)$. We omit this dependence in the rest of this paper to keep the notation as light as possible.

[60] High values of the relative effective sample size $(\sim 70\%)$ for liquid systems have been obtained in Ref. 32 in the isobaric-isothermal ensemble. In this case the presence of a learnable scaling parameter improves the sampling and helps transforming liquid configurations between different points of the $T - P$ diagram.