Learning Efficient, Collective Monte Carlo Moves with Variational Autoencoders

Jacob I. Monroe* and Vincent K. Shen*

Chemical Sciences Division, National Institute of Standards and Technology, Gaithersburg, Maryland 20899-8320, USA

E-mail: jacob.monroe@nist.gov; vincent.shen@nist.gov

Abstract

Discovering meaningful collective variables for enhancing sampling, via applied biasing potentials or tailored MC move sets, remains a major challenge within molecular simulation. While recent studies identifying collective variables with variational autoencoders (VAEs) have focused on the encoding and latent space discovered by a VAE, the impact of the decoding and its ability to act as a generative model remains unexplored. We demonstrate how VAEs may be used to learn (on-the-fly and with minimal human intervention) highly efficient, collective Monte Carlo moves that accelerate sampling along the learned collective variable. In contrast to many machine learning-based efforts to bias sampling and generate novel configurations, our methods result in exact sampling in the ensemble of interest and do not require re-weighting. In fact, we show that the acceptance rates of our moves approach unity for a perfect VAE model. While this is never observed in practice, VAE-based Monte Carlo moves still enhance sampling of new configurations. We demonstrate, however, that the form of the encoding and decoding distributions, in particular the extent to which the decoder reflects the underlying physics, greatly impact the performance of the trained VAE.

1 Introduction

Designing and engineering soft materials requires a fundamental understanding of underlying microstructures and how their formation is driven by molecular-level interactions. While molecular simulation provides a promising avenue for studying self-assembly and microstructure formation, its brute-force application to such problems is often intractable due to insurmountable length and timescales associated with assembly processes. Coarse-graining[1] and enhanced sampling[2,3] techniques present potential solutions. In the former, degrees of freedom are removed from the system by combining atoms or molecules into coarse-grained sites with effective interactions, allowing for more rapid configurational sampling. However, even with recent advances in coarse-grained potentials[4] and training protocols,[5] capturing solvent-mediated interactions, like hydrophobic associations, remains challenging due to their collective, many-body nature.[6] Capturing such many-body interactions involving both solutes and solvent becomes especially relevant when extrapolating coarse-grained models trained on fine-grained interactions between two peptides, for example, to simulations of hundreds of interacting peptides. More generally, it is difficult to assess the applicability of a coarse-grained model to regions of configuration space not observed in the fine-grained training set, despite such exploration often being the motivation behind developing the coarse-grained model. Enhanced sampling instead preserves atomistic detail, driving sampling along crucial, low-dimensional coordinates. Unfortunately, it remains difficult to identify collective variables and appropriate biases to drive sampling, particularly for assembling biomolecules that may significantly change their structure upon association.

Many of the most recent innovations in collective variable discovery are based on machine learning, specifically neural networks with autoencoding structures. Chen et al.[7] proposed an effective strategy for iteratively identifying low-dimensional coordinates of biomolecules with autoencoders followed by biased sampling along these coordinates. Tiwary and coworkers[8] extended these techniques by using variational autoencoders (VAEs),[9] a state-of-the-art machine learning method for dimensionality reduction and generative modeling, to simultaneously learn the collective variable and bias to drive sampling. Related techniques use autoencoding structures to predict

time series,[10–14] which identify coordinates along which dynamics are slow. Unlike VAEs utilized here and by Ribeiro et al.[8], these methods focus on dynamics and do not directly learn the probability density, or equivalently free energy, along a meaningful latent coordinate, a distinction that is critical to this work.

Here, we demonstrate how training a VAE not only learns a low-dimensional collective variable and its probability density, but also efficient Monte Carlo (MC) moves that pass into and out of that latent space, accelerating sampling. In other words, we also make use of the learned decoding, whereas past efforts with autoencoders have focused solely on the encoding. Crucial to our methods is the fact that VAEs can predict probability densities for the encoding, latent coordinate, and decoding, in addition to learning mappings to and from a low-dimensional latent space. The ability to simultaneously learn dimensionality reductions and probability densities is unique to VAEs and is not present in other modern machine learning techniques for density estimation, such as normalizing flows[15] and generative adversarial networks.[16]

Generative modeling to produce new configurations typically takes the form of MC or molecular dynamics simulations. Recent work, however, has utilized machine learning for this task, either through normalizing flows,[17,18] application of standard autoencoders with a modified loss,[19,20] or by combining diverse techniques for dimensionality reduction, time propagation, and decoding.[21] These techniques, however, only result in approximate sampling of the ensemble of interest and require re-weighting to recover correct statistics. Re-weighting must be applied carefully, however, as biased or limited sampling can still result in inaccurate estimates, as is often observed in free energy perturbation methods.[22,23] In stark contrast, the VAE-based MC moves presented here satisfy detailed balance and *exactly* recover the ensemble of interest. This exact sampling is also enhanced by the VAE's knowledge of a highly informative latent coordinate, with MC moves passing into, along, and out of the latent space resulting in rapid configurational transitions.

With exact sampling, our method is related to, and in fact inspired by, resolution exchange[24,25] and nested MC.[26,27] This includes recent variants based on machine learning.[28–31] By learning the probability density of the latent space, however, our methods directly draw uncorrelated sam-

ples from this distribution rather than perform long simulations at lower resolution and/or with a cheaper Hamiltonian. This results in drastically reduced autocorrelation times and more efficient sampling. In fact, we show that acceptance probabilities of VAE-based MC moves approach unity in the limit of a VAE model that perfectly learns the statistics of the ensemble used for training. Training a VAE thus represents a general method of developing collective MC moves with high acceptance rates, which further extends and complements existing approaches, such as Rosenbluth sampling,[32] hybrid MC,[33] and cluster MC moves.[34]

We organize this paper as follows: Section 2 establishes the mathematical underpinnings of our methods and derives the main theoretical contribution of this work, Section 3 describes the computational implementation, and Section 4 assesses the applicability of our technique to a variety of systems and discusses opportunities for improvements.

2 Theory

2.1 VAE Objectives and Probabilistic Models

VAEs consist of an encoding/decoding structure that is probabilistic in nature, shown schematically in Fig. 1. Given a model for the probability $P(\mathbf{x};\boldsymbol{\lambda})$ of a configuration $\mathbf{x}$, our objective is to maximize the expected log-likelihood $\langle \ln P(\mathbf{x};\boldsymbol{\lambda})\rangle_{\mathcal{P}(\mathbf{x})} = \int \mathcal{P}(\mathbf{x}) \ln P(\mathbf{x};\boldsymbol{\lambda})\, \mathrm{d}\mathbf{x}$ under sampling over the true ensemble probability distribution $\mathcal{P}(\mathbf{x})$ by modifying the parameters $\boldsymbol{\lambda}$. For instance, one-dimensional data x distributed according to an unknown probability density $\mathcal{P}(x)$ might be modeled with a Gaussian probability density for $P(x;\boldsymbol{\lambda})$, with likelihood maximization achieved by determining the mean and variance of x, which would be the model parameters $\boldsymbol{\lambda}$. If the training ensemble probability does not depend on $\boldsymbol{\lambda}$, then maximizing the likelihood of $P(\mathbf{x};\boldsymbol{\lambda})$ is

equivalent to minimizing the relative entropy (or Kullback-Leibler divergence)

$$\mathrm{KL}\left(P\left(\mathbf{x};\boldsymbol{\lambda}\right),\mathcal{P}\left(\mathbf{x}\right)\right) = \int \mathcal{P}\left(\mathbf{x}\right)\ln\frac{\mathcal{P}\left(\mathbf{x}\right)}{P\left(\mathbf{x};\boldsymbol{\lambda}\right)}\mathrm{d}\mathbf{x} \tag{1a}$$

$$= -\left\langle \ln P\left(\mathbf{x};\boldsymbol{\lambda}\right)\right\rangle_{\mathcal{P}(\mathbf{x})} + \left\langle \ln \mathcal{P}\left(\mathbf{x}\right)\right\rangle_{\mathcal{P}(\mathbf{x})} \tag{1b}$$

The first term on the right is the negative log-likelihood and the second is not involved in optimizations over the parameters $\boldsymbol{\lambda}$. Introducing a set of lower-dimensional (latent) coordinates $\mathbf{z}$ and using Bayes' rule we obtain the following objective to minimize

$$-\left\langle \ln P\left(\mathbf{x};\boldsymbol{\lambda}\right)\right\rangle_{\mathcal{P}(\mathbf{x})} = \Big\langle -\ln P\left(\mathbf{x}|\mathbf{z};\boldsymbol{\lambda}\right) - \ln P\left(\mathbf{z};\boldsymbol{\lambda}\right) + \ln P\left(\mathbf{z}|\mathbf{x};\boldsymbol{\lambda}\right)\Big\rangle_{\mathcal{P}(\mathbf{x})} \tag{2}$$

At this point, Eq. 2 is a completely general Bayesian modeling objective. To make it useful for specific applications, the form of the model probability densities must be specified.

Typically, and as implemented in this work, the decoding distribution $P\left(\mathbf{x}|\mathbf{z};\boldsymbol{\lambda}\right)$ of a VAE for continuous degrees of freedom is modeled as a multivariate normal $\mathcal{N}\left(\boldsymbol{\mu}\left(\mathbf{z}\right),\boldsymbol{\sigma}\left(\mathbf{z}\right)\right)$ with no covariance between degrees of freedom (i.e., the covariance matrix is solely defined by a vector of its diagonal elements).[35] Note that, in the notation defined above, the mean and variance are part of the model parameters $\boldsymbol{\lambda}$. For discrete degrees of freedom, such as for a lattice gas, the decoding is represented by a Bernoulli distribution $\mathrm{Bern}\left(\boldsymbol{\mu}\left(\mathbf{z}\right)\right)$, also with probabilistic independence between sites. To be mathematically precise, the decoding distribution for these models may be decomposed as

$$P\left(\mathbf{x}|\mathbf{z}\right) = \prod_{i=1} P\left(x_i|\mathbf{z}\right) \tag{3}$$

For molecules and dense fluids, the assumption of independence is not appropriate, which we make clear in Section 4. Instead, we require an autoregressive model, as employed in state-of-the-art

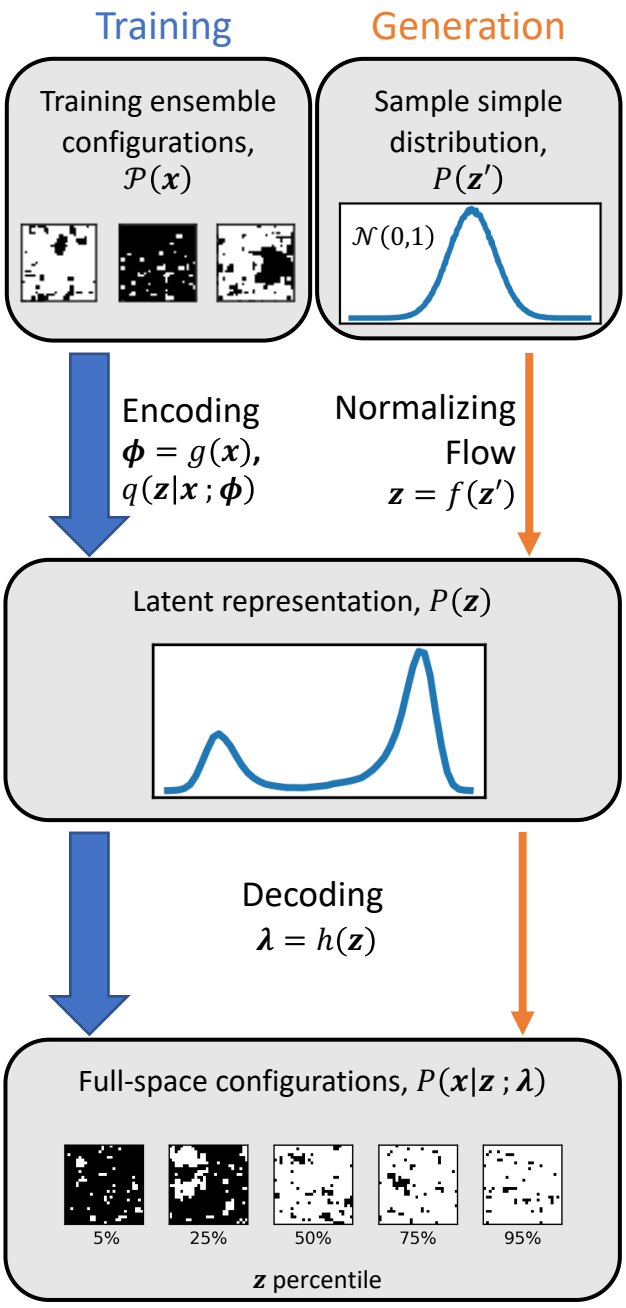

Figure 1: Autoencoding structure involves an encoding to a lower-dimensional space and a decoding to the original space. During training, parameters are adjusted so that configurations sampled from the ensemble of interest have high probability under $P(\mathbf{x}|\mathbf{z};\boldsymbol{\lambda})$ after being encoded and decoded. To generate new structures, we draw from a simple distribution that is transformed into the latent distribution to be decoded.

architectures for generating images[36]

$$P(\mathbf{x}|\mathbf{z}) = P(x_1|\mathbf{z}) \prod_{i=2} P(x_i|\mathbf{z}, \mathbf{x}_{<i}) \tag{4}$$

The meaning of Eq. 4 is that degree of freedom i is sampled with full knowledge of the sampled values of all $i-1$ degrees of freedom. As a result, autoregressive models are capable of capturing even long-range correlations. In practice, sampling then proceeds iteratively, updating the model probability density for the next degree of freedom that will be sampled based on those already drawn, i.e., $\mu_i(\mathbf{z}, \mathbf{x}_{<i})$ and $\sigma_i(\mathbf{z}, \mathbf{x}_{<i})$. Autoregressive decoders will be used throughout this work, except where otherwise indicated.

It is important to note that, even without autoregression, the decoding model is remarkably flexible due to the dependence of the parameters $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ on the latent coordinates $\mathbf{z}$. In fact, this dependence is approximated by a highly nonlinear, powerful neural network taking the latent coordinates as inputs, i.e., $(\boldsymbol{\mu}, \boldsymbol{\sigma}) = h(\mathbf{z})$. Any information encoded in the latent space may thus be communicated to the decoding distribution to build in correlations between degrees of freedom. Autoregressive structure, then, is most useful for situations where the latent space dimensionality is lower than that necessary for a lossless encoding, which is desirable for accelerating molecular simulations.

Latent space distributions, or priors, should be simple enough to facilitate generative modeling, but also complicated enough to allow for information-rich encodings. This presents a dilemma that is resolved by placing normalizing flows on the prior.[37–41] Normalizing flows[15] are bijective mappings defined by neural networks for which the Jacobian determinant $R_{\mathbf{z}' \to \mathbf{z}}(\mathbf{z}')$ is exactly known. This allows for sampling from the classic choice of a standard normal $\mathcal{N}(\mathbf{0}, \mathbf{1})$ without overly simple latent distributions arising upon training of the overall VAE objective.[40,42,43] By training the parameters of the neural networks, we learn a mapping $f(\mathbf{z}')$ from a simple distribution in coordinates $\mathbf{z}'$, here a standard normal, to an arbitrarily complicated distribution in $\mathbf{z}$ while simultaneously gaining the ability to estimate its probability density. Once a VAE is trained, new

configurations may be generated as shown in Fig. 1 by drawing from a standard normal, passing through the flow, and decoding.

With the decoding and latent distributions specified, and training data specifying the target for $P(\mathbf{x};\boldsymbol{\lambda})$, Bayes' rule indicates that there is no freedom left to arbitrarily specify the encoding distribution. To reclaim this ability and simplify training, we introduce a variational distribution $q(\mathbf{z}|\mathbf{x};\boldsymbol{\phi})$ defined by its own parameters $\boldsymbol{\phi}$. Adding $\langle \ln q(\mathbf{z}|\mathbf{x};\boldsymbol{\phi})\rangle_{\mathcal{P}(\mathbf{x})}$ to both sides of Eq. 2, rearranging, and averaging in $\mathbf{z}$ over this distribution, we obtain the VAE objective function to minimize[9,35]

$$
\begin{aligned}
&-\left\langle \ln P(\mathbf{x};\boldsymbol{\lambda}) - \langle \ln q(\mathbf{z}|\mathbf{x};\boldsymbol{\phi}) - \ln P(\mathbf{z}|\mathbf{x};\boldsymbol{\lambda})\rangle_{q(\mathbf{z}|\mathbf{x};\boldsymbol{\phi})} \right\rangle_{\mathcal{P}(\mathbf{x})} \\
&= \left\langle \langle -\ln P(\mathbf{x}|\mathbf{z};\boldsymbol{\lambda}) - \ln P(\mathbf{z};\boldsymbol{\lambda}) + \ln q(\mathbf{z}|\mathbf{x};\boldsymbol{\phi})\rangle_{q(\mathbf{z}|\mathbf{x};\boldsymbol{\phi})} \right\rangle_{\mathcal{P}(\mathbf{x})} \qquad (5) \\
&-\langle \ln P(\mathbf{x};\boldsymbol{\lambda}) - \mathrm{KL}\left(q(\mathbf{z}|\mathbf{x};\boldsymbol{\phi}), P(\mathbf{z}|\mathbf{x};\boldsymbol{\lambda})\right)\rangle_{\mathcal{P}(\mathbf{x})} \\
&= \left\langle \langle -\ln P(\mathbf{x}|\mathbf{z};\boldsymbol{\lambda})\rangle_{q(\mathbf{z}|\mathbf{x};\boldsymbol{\phi})} + \mathrm{KL}\left(q(\mathbf{z}|\mathbf{x};\boldsymbol{\phi}), P(\mathbf{z};\boldsymbol{\lambda})\right) \right\rangle_{\mathcal{P}(\mathbf{x})} \qquad (6)
\end{aligned}
$$

The right-hand side of Eq. 6 is an upper bound on the negative log likelihood. The bound is tight when $q(\mathbf{z}|\mathbf{x};\boldsymbol{\phi})$ is a close approximation to $P(\mathbf{z}|\mathbf{x};\boldsymbol{\lambda})$, which is greatly facilitated by a flexible latent distribution,[42] such as one defined by a normalizing flow. We follow typical convention in VAEs by using a multivariate normal distribution with zero covariance for $q(\mathbf{z}|\mathbf{x};\boldsymbol{\phi})$, with the means and variances represented by $\boldsymbol{\phi}$ and determined by the neural network $g(\mathbf{x})$.

Before continuing to define VAE-based MC moves, it is worthwhile to comment on the generality of the Bayesian framework for probabilistic modeling, noting relative entropy coarse-graining[44] as a special case. Deterministic mappings, such as those used in coarse-graining, are specific cases of probabilistic mappings, where the probability density is represented by a delta function defined by mapping function $\mathbf{z} = g(\mathbf{x})$. If this is assumed for the encoder distribution $P(\mathbf{z}|\mathbf{x};\boldsymbol{\lambda})$, then

inserting this into the objective in Eq. 2 and averaging over $P(\mathbf{z}|\mathbf{x};\boldsymbol{\lambda})$ on both sides yields

$$
\begin{aligned}
-\left\langle \ln P(\mathbf{x};\boldsymbol{\lambda})\right\rangle_{\mathcal{P}(\mathbf{x})} = \Big\langle & -\ln P(\mathbf{x}|g(\mathbf{x});\boldsymbol{\lambda}) \\
& -\ln P(g(\mathbf{x});\boldsymbol{\lambda}) \\
& +\int \delta(\mathbf{z}-g(\mathbf{x}))\ln\delta(\mathbf{z}-g(\mathbf{x}))\,\mathrm{d}\mathbf{z}\Big\rangle_{\mathcal{P}(\mathbf{x})}
\end{aligned}
\tag{7}
$$

Though the integral in Eq. 7 diverges, it does not involve any optimizable parameters if the mapping is fixed (i.e., not modified during training), and may thus be considered a constant for the purposes of minimizing the left-hand side. With such a fixed mapping, statistical mechanics supplies that the distribution in latent, or coarse-grained, space will be a Boltzmann distribution with the coarse-grained potential U_{CG} defined by parameters $\boldsymbol{\lambda}_U$ approximating the potential of mean force. Inserting this into Eq. 7 and neglecting terms that do not depend on optimizable parameters, we obtain the original relative entropy objective in Ref. 44 with one critical distinction

$$
\min_{(\boldsymbol{\lambda},\boldsymbol{\lambda}_U)}\left\langle -\ln P(\mathbf{x}|g(\mathbf{x});\boldsymbol{\lambda}) + \beta U_{\mathrm{CG}}(g(\mathbf{x});\boldsymbol{\lambda}_U) - \beta A_{\mathrm{CG}}\right\rangle_{\mathcal{P}(\mathbf{x})}
\tag{8}
$$

The distribution $P(\mathbf{x}|\mathbf{g}(\mathbf{x});\boldsymbol{\lambda})$ is related to the mapping entropy defined by Shell[44], who, focusing on the development of coarse-grained models, assumed this distribution to be uniform and hence a constant of the optimization. Decoding models presented in this work may instead be dropped into Eq. 8 and trained simultaneously with coarse-grained potentials to obtain a back-mapping, opening up the possibility of multi-scale models that switch efficiently between resolutions, which we briefly explore in the Supporting Information (SI). Such ideas have previously been investigated by Wang and Gómez-Bombarelli[45], though the authors focus on using an innovative encoder design to learn coarse-grained mappings and potentials without exploring different probabilistic decoder models or sampling from the decoding distribution. In the next section, we describe how explicit use of the decoding probability distribution enables generation of configurations that, along with appropriate acceptance criteria, exactly sample the fine-grained ensemble of interest.

2.2 VAE-based MC moves

We are now ready to derive VAE-based MC moves, the main theoretical contribution of this paper. Because all of the model probability distributions in a VAE are known, they can be used as proposal probabilities for MC moves passing into and out of the latent space. We show in Fig. 2a how we use the encoder distribution $q(\mathbf{z}|\mathbf{x};\boldsymbol{\phi})$ to propose a configuration in latent space $\mathbf{z}_1$, the prior $P(\mathbf{z})$ to sample a new latent configuration $\mathbf{z}_2$, and the decoder distribution $P(\mathbf{x}|\mathbf{z};\boldsymbol{\lambda})$ to propose a new full-space configuration $\mathbf{x}_2$. With the following ratio of acceptance probabilities, we impose super-detailed balance[46] (i.e., requiring that detailed balance be satisfied for all possible paths through latent space and back)

$$\frac{\alpha_{1\to2}}{\alpha_{2\to1}} = \frac{\mathcal{P}(\mathbf{x}_2)\,q(\mathbf{z}_2|\mathbf{x}_2;\boldsymbol{\phi})\,P(\mathbf{z}_1)\,P(\mathbf{x}_1|\mathbf{z}_1;\boldsymbol{\lambda})}{\mathcal{P}(\mathbf{x}_1)\,q(\mathbf{z}_1|\mathbf{x}_1;\boldsymbol{\phi})\,P(\mathbf{z}_2)\,P(\mathbf{x}_2|\mathbf{z}_2;\boldsymbol{\lambda})} \tag{9}$$

Subscripts 1 and 2 indicate the starting and newly proposed configurations, respectively. In the event that the model is well-trained and the variational distribution is sufficiently flexible, $q(\mathbf{z}|\mathbf{x};\boldsymbol{\phi}) \approx P(\mathbf{z}|\mathbf{x};\boldsymbol{\lambda})$. Bayes' rule then provides that

$$P(\mathbf{x};\boldsymbol{\lambda}) = \frac{P(\mathbf{z})\,P(\mathbf{x}|\mathbf{z};\boldsymbol{\lambda})}{P(\mathbf{z}|\mathbf{x};\boldsymbol{\lambda})} \tag{10}$$

Substitution into Eq. 9 leads to

$$\begin{aligned}\frac{\alpha_{1\to2}}{\alpha_{2\to1}} &= \frac{\mathcal{P}(\mathbf{x}_2)\,P(\mathbf{x}_1;\boldsymbol{\lambda})}{\mathcal{P}(\mathbf{x}_1)\,P(\mathbf{x}_2;\boldsymbol{\lambda})} \\ &\approx 1\end{aligned} \tag{11}$$

As indicated in Eq. 11, the acceptance ratio approaches unity for a well-trained VAE for which the encoding is lossless. In practice, this limit is difficult to reach, but it is remarkable that, through training a VAE, we may discover highly efficient MC moves in addition to defining information-rich latent coordinates. With our formulation, VAE-based MC moves can significantly accelerate

sampling while satisfying detailed balance. The extent to which sampling is accelerated is only dependent on how well the VAE model approximates the true distribution, with the theoretical limit being perfectly uncorrelated sampling of the ensemble of interest.

We may additionally conduct biased VAE-based MC moves in order to achieve more uniform sampling within the learned CG space. For a given $\mathbf{x}$, a trained VAE provides an estimate of the biasing potential $W(\mathbf{x})$ that is the negative of the free energy along the latent coordinate and can be applied to achieve more uniform sampling in $\mathbf{z}$ space

$$\beta W(\mathbf{x}) = -\ln \int P(\mathbf{z})\, q(\mathbf{z}|\mathbf{x};\boldsymbol{\phi})\, \mathrm{d}\mathbf{z} \tag{12}$$

Intuitively, the integral is the average latent space probability associated with $\mathbf{x}$. Based on the analytically known transformed prior $P(\mathbf{z}')$ (a standard normal) and the definition of the flow $f(\mathbf{z}')$ with an easily calculated Jacobian determinant $R_{\mathbf{z}'\to\mathbf{z}}(\mathbf{z}')$, the probability $P(\mathbf{z})$ for D_l latent dimensions is

$$P(\mathbf{z}) = \frac{1}{(2\pi)^{D_l/2}} \mathrm{e}^{-\frac{1}{2}\left(\sum_i f^{-1}(z_i)^2\right) + \ln R_{\mathbf{z}\to\mathbf{z}'}(\mathbf{z})} \tag{13}$$

While $q(\mathbf{z}|\mathbf{x};\boldsymbol{\phi})$ is a known Gaussian, the bijective transformation $f^{-1}(\mathbf{z})$ appearing in Eq. 13 involves neural networks and is not easily integrated over. In practice, we draw many $\mathbf{z}$ from the distribution $q(\mathbf{z}|\mathbf{x};\boldsymbol{\phi})$ and estimate the average of Eq.13 over this sample. Since this is a stochastic estimate of the biasing potential, the resulting MC simulation will not be strictly equilibrium, but will satisfy detailed balance on average, and will come closer to doing so as the number of sampled $\mathbf{z}$ increases.

In the work of Noé et al. [17], an MC sampling scheme based on the model distribution learned by normalizing flows is also proposed for enhancing sampling. In that scheme, however, the probability determining MC sampling is that of the latent distribution (note that for a normalizing flow as utilized in Ref. 17, the "latent" $\mathbf{z}$ is of the same dimensionality as $\mathbf{x}$) under mapping from

the ensemble of training configurations

$$P(\mathbf{z}) = \frac{\mathcal{P}(\mathbf{x})}{R_{\mathbf{x}\to\mathbf{z}}(\mathbf{x})} \tag{14}$$

This is different from the probability of the ensemble of interest by a factor of the Jacobian determinant $R_{\mathbf{x}\to\mathbf{z}}(\mathbf{x})$, which will not cancel as it changes for different full-space coordinates, so that the acceptance ratio is

$$\frac{\alpha_{1\to2}}{\alpha_{2\to1}} = \frac{\mathcal{P}(\mathbf{x}_2) R_{\mathbf{x}\to\mathbf{z}}(\mathbf{x}_1)}{\mathcal{P}(\mathbf{x}_1) R_{\mathbf{x}\to\mathbf{z}}(\mathbf{x}_2)} \tag{15}$$

As a result, the MC sampling scheme of Noé et al.[17] samples the learned latent space, but will not necessarily produce a sample from the ensemble of interest. It will in the limit of a perfect normalizing flow model, but otherwise reproduces the distribution that is learned through training on limited samples. While this does accelerate sampling, it does not result in exact sampling, as our VAE-based MC moves do, and requires re-weighting.

To achieve exact sampling with normalizing flows, one can propose configurations by sampling from the latent space (e.g., a standard normal over all degrees of freedom) and mapping back to the model distribution $P(\mathbf{x})$ learned from training data drawn from $\mathcal{P}(\mathbf{x})$. As shown by Nicoli et al.[47], the acceptance ratio that satisfies detailed balance in the ensemble of interest with this proposal scheme is

$$\frac{\alpha_{1\to2}}{\alpha_{2\to1}} = \frac{\mathcal{P}(\mathbf{x}_2) P(\mathbf{z}_1) R_{\mathbf{z}\to\mathbf{x}}(\mathbf{z}_2)}{\mathcal{P}(\mathbf{x}_1) P(\mathbf{z}_2) R_{\mathbf{z}\to\mathbf{x}}(\mathbf{z}_1)} = \frac{\mathcal{P}(\mathbf{x}_2) P(\mathbf{x}_1)}{\mathcal{P}(\mathbf{x}_1) P(\mathbf{x}_2)} \tag{16}$$

Wu et al.[48] developed a similar MC scheme based on non-equilibrium candidate MC moves[49] that also satisfies detailed balance for non-bijective flows with stochastic sampling layers. While such MC moves based on normalizing flows may also approach acceptance ratios of unity, no dimensionality reduction is performed and hence no information-rich latent space is defined. Since VAEs focus sampling along the learned low-dimensional manifold, they are expected to result in improved sampling. With our formulation of proposal probabilities for VAE-based MC moves, sampling is unbiased and exact, contrary to the claims of Nicoli et al.[47].

Sampling is also expected to be significantly improved over resolution exchange[24,25] or nested

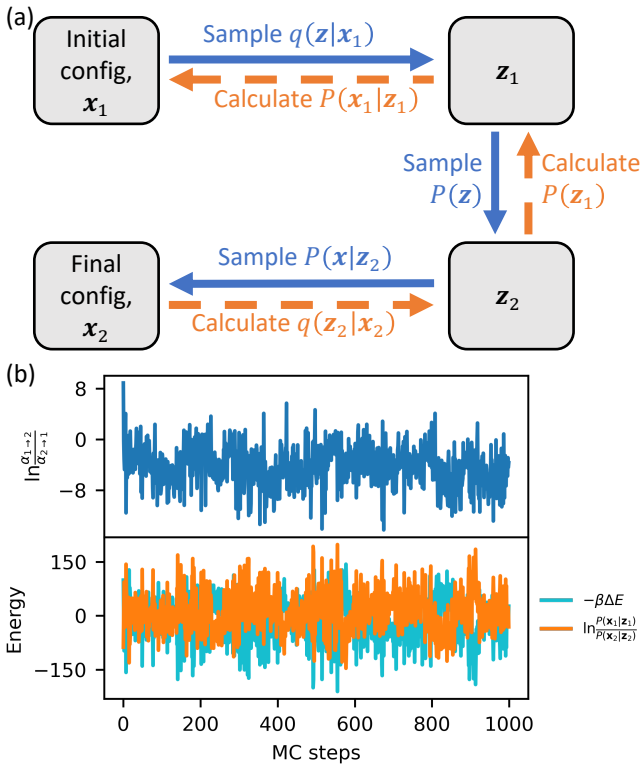

Figure 2: (a) MC moves are performed by passing through the learned latent space, naturally leading to efficient MC moves for well-trained VAE models. (b) We show instantaneous values of the logarithm of the acceptance probability ratio in Eq.9 (top panel) and the largest contributions to this on the right-hand-side of Eq. 9 (bottom panel). Specifically, $-\beta\Delta E$ is the difference in the lattice gas total energy (particle-particle and chemical potential terms) between configurations $\mathbf{x}_2$ and $\mathbf{x}_1$, and it is also the logarithm of the ratio of $\mathcal{P}(\mathbf{x}_2)$ to $\mathcal{P}(\mathbf{x}_2)$. Nearly offsetting this term is the logarithm of the ratio of probabilities output by the decoder. The VAE model used for MC sampling is autoregressive with a single latent coordinate and trained on the full set of lattice gas data.

MC.[26,27] In each of those methods, a coarse-grained, or simply computationally cheaper, Hamiltonian governs MC moves or molecular dynamics in the latent space. New configurations at full resolution or with the expensive Hamiltonian are then drawn from these simulations. In contrast, samples from VAE-based MC moves come directly from the probability density of the prior that is learned during training. Direct sampling from the latent distribution in this way results in perfectly uncorrelated samples, which is the limit of infinitely long simulations in the latent space. If a specific coarse-grained model is desirable, decoders may be trained as presented here and applied to similar MC moves with latent configurations generated by coarse-grained simulations. An example of such training is provided in the SI.

3 Methods

3.1 Neural Network Architecture and VAE Model Training

All models are implemented in TensorFlow[50] (version 2.4.0) and trained on an NVIDIA RTX 2080TI GPU. Stochastic encoders are modeled by fully-connected neural networks with 2 hidden layers of 1200 units each for the lattice gas and 300 units for all other systems. A separate dense network layer maps the output of the hidden layers to the means and log-variances of the coarse-grained (CG) space. RELU activation is used for all but the last layer.

Flows follow the RealNVP structure of Dinh et al.[37] but utilize monotonic rational quadratic splines parametrized by neural networks, as introduced by Durkan et al.[38], which enables powerful transformations for both many- and single-dimensional spaces. Neural networks to predict the spline parameters consist of a single fully-connected layer mapping the latent coordinates $\mathbf{z}$ to a hidden dimension of 200 followed by three dense layers mapping to the spline parameters, where 32 bins on the interval $[-10, 10]$ are used to partition spline points. This interval is arbitrary as long as it encompasses the region of $\mathbf{z}$ space with non-zero probability density.

Conditionally independent decoders contain fully-connected networks with tanh activations and hidden dimensions of 1200 for the lattice gas and 300 for all other systems. A fully-connected

network with no activation then predicts the logits associated with the probabilities of each lattice site containing a particle, or the means and log-variances of degrees of freedom in continuous systems. The autoregressive decoder similarly starts with two fully-connected neural networks with tanh activations and hidden dimensions of 1200 (or 300 for all but the lattice gas). Another fully-connected layer then maps to a set of starting logit values for the lattice, or mean and log-variance values for other systems. A masked network[51] with identical input and output dimensions predicts additive shifts to the outputs predicted by the previous layer. Masking is used to enforce autoregressive structure, with each degree of freedom only influenced by degrees of freedom with a lower index. For example, the base logit value predicted for the first degree of freedom by the previous layer will not be modified by this masked network, while the second degree of freedom will be shifted by an amount that depends only on the *sampled* value for the first degree of freedom, as well as the latent space since skip connections[52] are utilized. When generating new structures, this means that a loop over the dimensionality of the full configurational space is performed. Each degree of freedom is sampled before passing that and all previously sampled values to the masked network in order to create the distribution for the next degree of freedom, thus requiring as many evaluations of the masked neural network as dimensions in the space.

The decoding model distribution $P(\mathbf{x}|\mathbf{z};\boldsymbol{\lambda})$ is highly system-dependent. For all systems studied and VAE models implemented, it is important to note that we have not applied convolutional schemes or considered permutational invariances (e.g., lattice sites or particle identities) of system degrees of freedom. As such, each trained VAE model is specific to the ordering of degrees of freedom and coordinate system chosen for the training data. Lattice gas configurations are discrete, and so a Bernoulli distribution is chosen for the decoder $P(\mathbf{x}|\mathbf{z};\boldsymbol{\lambda}) \sim \text{Bern}(\boldsymbol{\mu}(\mathbf{z}))$, where $\boldsymbol{\mu}$ is the vector of probabilities for each site to be occupied (or, equivalently, mean occupancy). The parameters $\boldsymbol{\lambda}$ are the logits $\boldsymbol{\alpha}$ associated with the probabilities for each site, with, for site i, $\mu_i = \frac{1}{1+e^{-\alpha_i}}$. Logits are output by the decoder's neural network $h(\mathbf{z})$ because their values are unconstrained on the real line, but their definition guarantees probabilities between 0 and 1. For continuous, non-periodic degrees of freedom, $P(\mathbf{x}|\mathbf{z};\boldsymbol{\lambda}) \sim \mathcal{N}\left(\boldsymbol{\mu}(\mathbf{z}),\boldsymbol{\sigma}^2(\mathbf{z})\right)$. Directly sampling continuous,

periodic degrees of freedom, such as dihedral angles of molecules, requires a von Mises distribution $P(\mathbf{x}|\mathbf{z};\boldsymbol{\lambda}) \sim \text{vonMises}(\boldsymbol{\mu}(\mathbf{z}),\boldsymbol{\kappa}(\mathbf{z}))$. If the decoder alternatively predicts sine-cosine pairs, we use a Gaussian distribution instead, though, as we show in the SI, this tends to perform more poorly than a von Mises distribution (Figs. S3-5, Table S1). Notice that $\boldsymbol{\sigma}^2$ and $\boldsymbol{\kappa}$ must be positive, and so their natural logarithms are output by $h(\mathbf{z})$ and transformed.

Models are all trained with an Adam optimizer[53] with a learning rate of 0.0001 and other parameters following their default values in TensorFlow. For VAE models of the lattice gas, the encoder and decoder are first trained with a weight on the Kullback-Leibler (KL) divergence term annealed from 0 to 1 over 20 epochs. This helps prevent initial domination of the KL divergence term and collapse of the encoding to an overly simple, uninformative distribution.[54] We subsequently optimize with the full ELBO objective for 80 epochs over shuffled data with a batch size of 200, which achieves converged loss functions and reproducible results (Figs. S6-8). When training on mid- and high-density data, the training time with the full loss is extended to 180 epochs since these datasets contain fewer samples (Fig. S6). For molecular systems, we instead anneal for 40 epochs and train manually until improvement in the 5% of the training set left out for validation becomes small (see Figs. S9-11). For *n*-eicosane, this is 320 epochs (Fig. S9). Alanine required 480 epochs (Fig. S10), and training of the 2D fluid dimer system required 160 epochs after annealing (Fig. S11).

3.2 Simulations

Simulations of a 2D lattice gas are executed with translation, insertion, and deletion MC moves, as described in detail in the SI. Reference data for training VAE models comes from two simulations, one starting at low density and one starting at high density, of 100 000 000 steps each with configurations saved every 1 000 steps, with trajectories in particle number shown in Fig. S15. While validation during training involves a randomly selected 5% of the reference data, an independent test set was used to assess VAE models. The independent test data set comes from initiating 100 independent MC simulations (with only translation, insertion, and deletion moves) from random

configurations drawn from the trajectory generated by VAE-based MC moves using an autoregressive model with a 1D latent space trained on the reference data set. We run each simulation for 2 000 000 steps with configurations again saved every 1 000 steps. In all MC simulations without VAE-based moves, we selected translation randomly with 50% probability and insertion and deletion each with 25% probability.

For model polymers and alanine dipeptide, we use the OpenMM package[55] to generate configurations with molecular dynamics simulations. Eicosane in vacuum is modeled with the TraPPE united atom representation.[56] We model alanine dipeptide in vacuum with the AMBER99-SB force field.[57] All molecular systems are simulated for 1 μs with configurations saved every 1 ps. Training data for VAEs is sampled uniformly from every tenth configuration to yield 100 000 samples. For all molecular systems, bond-angle-torsion coordinates (computed using the BAT analysis module of MDAnalysis[58–60]) are used with rigid translation and rotation removed for the purposes of training VAEs. Bonds with constrained lengths are also removed from the degrees of freedom provided for training, leaving only non-rigid bonds, angles, and dihedrals. During the encoding process, all periodic degrees of freedom (i.e., dihedral angles) are converted to sine-cosine pairs, which provides coordinates that keep similar angles close to each other regardless of the presence of periodic boundaries.[7] Unlike Chen et al.[7], we are not necessarily interested in directly utilizing the discovered latent space for biasing molecular dynamics simulations, so its smoothness only affects the ease with which the flow on the prior converts this distribution to a standard normal. Additionally, those authors only considered autoencoders with a reconstruction term in the loss (first term on the right-hand-side of Eq. 6). Introduction of the KL divergence term pushes the encoding to be smooth to facilitate the prior flow. As such, we do not consider explicit addition of circular degrees of freedom to the latent space.[7]

The 2D Lennard-Jones fluid system with a single highly attractive dimer is described in Ref. 17. We conduct simulations using the software[61] published in association with the work of Noé et al.[17]. Simulations starting from both open and closed states are run for 10 000 000 steps with configurations saved every 100 steps after a burn-in period of 10 000 steps. As crossing between

open and closed states is never observed, this generates 100 000 configurations from each of the open and closed states that are used for training VAE models. Before training, all particle identities are permuted to keep each particle trajectory localized to a small region within the simulation box[17] rather than passing through all parts of the box as particles diffuse past each other. Permuted particle trajectories are also centered and whitened before training, which improves VAE performance (Fig. S3, Table S1). Full details for simulations of molecular systems and the 2D fluid are provided in the SI.

4 Results and discussion

4.1 Lattice gas

As a first test of VAE-based MC moves, we turn to a lattice gas, which is a realistic model of a fluid under conditions of vapor-liquid phase equilibria and exhibits fluctuating microstructures. In two dimensions, this model may be represented by a binary image, with each pixel assuming a 0 where no particle exists and a 1 where a particle is present. All simulations are performed in the grand canonical ensemble, with the particle-particle interaction strength and chemical potential chosen to make low- and high-density configurations equally probable at equilibrium, though the data used for training below does not necessarily reflect this due to finite sampling. Further details are provided in Section 3 and the SI.

Training a VAE with a single latent dimension results in a learned latent distribution (Fig. 1) that, aside from translation and scaling, is similar to the distribution of particle numbers used for training shown as a dashed black line in Fig. 3. From samples generated at various percentiles of $P(\mathbf{z})$ in Fig. 1, it is clear that this coordinate also encodes particle density. These results are for a VAE with an autoregressive decoder, which successfully generates configurations with particle number, energy, and cluster size distributions matching the training data (Fig. 3). The only notable difference from the training data is a slightly broader high-energy tail in the distribution of total energies. In contrast, Fig. 3 also shows that conditionally independent decoders are only capable of

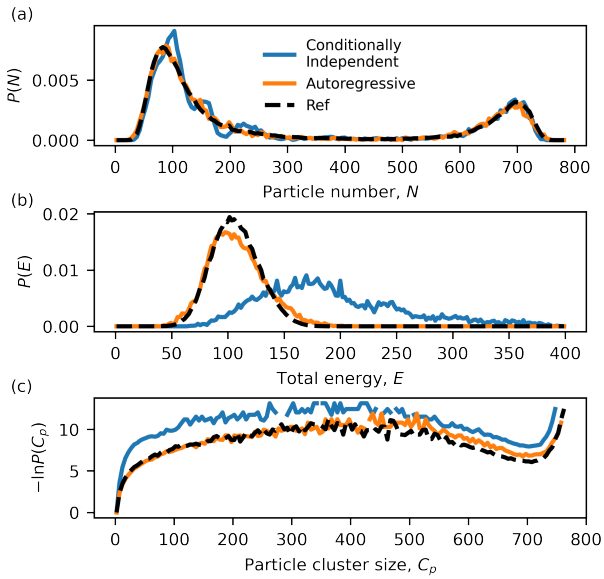

Figure 3: Distributions of (a) numbers of particles, (b) total energy (particle-particle and chemical potential contributions), and (c) particle cluster size are shown for various VAE models with a single latent dimension trained on lattice gas data. All distributions are reproduced well with autoregressive decoders, but poorly for conditionally independent decoder models. Black dashed lines represent normalized fine-grained ensemble histograms over 200 000 samples from the reference simulations, while model curves represent 10 000 draws from standard normal samples passed through a flow and decoded.

accurately capturing distributions in particle number. Such models ignore long-range dependencies and fail to capture the correct cluster size distribution, which results in higher potential energies.

We note that the ability of a VAE model to reproduce inputs is sensitive to its overall structure. For instance, increasing the latent dimension allows for more expressive latent spaces and improved conditionally independent models. Reconstructions (passing configurations through the encoder and decoder and resampling) from an autoregressive model with a single latent dimension correctly capture the size and number of clusters, but do not place clusters in the same locations as the original image (Fig. S12). Moving up to 10 latent dimensions provides enough information for both autoregressive and conditionally independent models to match the locations of clusters. However, the conditionally independent model still includes too many small clusters that lead to high energies (Fig. S12).

Because a VAE learns a distribution based on a training data set, it will also learn inaccuracies or biases associated with this data. This is clear from Fig. 3 in that the distribution in particle number generated by the VAE matches that of the original dataset, which, due to limited sampling, is asymmetric rather than equally weighted to high and low densities. Unfortunately, this means that we expect trained VAE models to be only sparingly transferable, with relevance only to the data on which they are trained. Fig. 4 shows that this is indeed the case. When trained on low-density data alone, a VAE recognizes that high-density configurations should be placed far away in both latent space and configurational space, but drastically underpredicts the density. Due to this, the resulting configurations are of high energy, with their likelihood over-represented compared to the ground-truth ensemble. Given that high-density configurations were never observed during training, this might be anticipated, but we even observe a lack of transferability when the training data does include configurations relevant to the new conditions.

High-temperature data include configurations that are also likely at low temperatures, but reconstructing low-temperature data from a VAE trained on high-temperature information results in particle number distributions significantly shifted and broadened (Fig. 4). This is related to the fact that the trained VAE has no sense of temperature and attempts to reproduce the training *dis-*

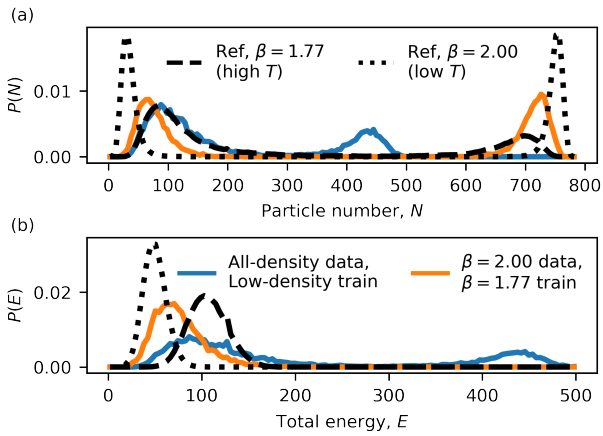

Figure 4: Particle number (a) and total energy distributions (b) are shown for models trained at specific density ranges or temperatures and applied to reconstructing data at a different density or temperature. The black dashed and dotted curves represent data at high and low temperatures. Blue curves are from reconstructions (i.e., passing through the encoder and decoder) of data from all densities with the VAE trained only on low densities (N less than 200). Orange curves are for a VAE trained on high-temperature ($\beta = 1.77$) data and applied to reconstructing low-temperature ($\beta = 2.00$) data.

tribution, which is broader at high temperatures, rather than exact configurations. Even though low-temperature configurations have been observed at a high temperature, their distribution of encodings in latent space and subsequent full-space decodings, are permissively broadened compared to what would need to be learned at a lower temperature. This acts to also broaden the reconstructed grand canonical energy distribution and shift it to higher energies. Figs. S13-14 of the SI demonstrate that we can build in explicit temperature dependence by training with datasets from multiple temperatures and scaling the variance of the prior distribution based on reduced temperatures, as described by Noé et al.[17]. While this enables the generation of configurations specific to a given temperature, it requires more extensive training data and, as we show later, does not limit the use of VAEs trained at one temperature to propose moves at another. Note that an inability to reconstruct distributions of data at different temperatures does not necessarily mean that the produced configurations are unphysical. Reconstructed configurations are likely to be physically relevant, but with the probability of their occurrence overestimated. As such, we demonstrate below (Fig. 6c) that VAE-based MC moves are in fact capable of accelerating sampling at temperatures different from

those used for VAE training.

Even if a VAE model is inaccurate due to limited training data, VAE-based MC moves exactly sample from the ensemble of interest and accelerate sampling in two ways: (1) increase acceptance rates by modeling probabilities used for proposals that offset probabilities governing sampling, as shown in Eq. 11 and (2) drive rapid conformational transitions by generating new configurations from a learned distribution in a lower-dimensional space (i.e., samples are drawn directly from $P(\mathbf{z})$) in Eq. 9). We observe the first mechanism in the lower panel of Fig. 2, which displays "time" series of the acceptance probability ratio in Eq. 9 for VAE-based MC moves. The data show that the ratio of $P(\mathbf{x}|\mathbf{z};\boldsymbol{\lambda})$ terms are highly correlated to, and largely offset by, total energy differences. Logarithms of other probabilities appearing in Eq. 9 are two orders of magnitude smaller, indicating that the model places the largest uncertainty in the decoding distribution. This is consistent with the idea that the encoding involves information loss and thus moving back, from a lower to a higher dimension, involves some guesswork. Probabilities in Eq. 9 do not exactly cancel, however, and the fraction of accepted MC moves (i.e., acceptance rate) based on a Metropolis criterion f_{acc} is 0.17 for the VAE model displayed in Fig. 2. Though significantly less than the theoretical upper bound of unity, this acceptance rate is approximately the same as that observed for MC simulations of a lattice gas with only translation, insertion, and deletion moves. Compared to these reference MC simulations, however, VAE-based moves accelerate sampling along the latent dimension, which is related to the particle density. This second mechanism for enhancing sampling is clear from comparing the time series of particle numbers in Fig. 5b and Fig. S15. VAE-based moves lead to many transitions between low- and high-density states, whereas only 6 such transitions were observed in 5 orders of magnitude more simulation steps with the standard move set. Autocorrelation functions in Fig. 5a provide further evidence that VAE-based MC moves drastically enhance sampling.

Despite reduced autocorrelation times, VAE-based moves are limited in their sampling based on the data used in training. Fig. 5 shows that a VAE trained only on low-density data will only increase sampling within the low-density regime. This is directly tied to the lack of transfer-

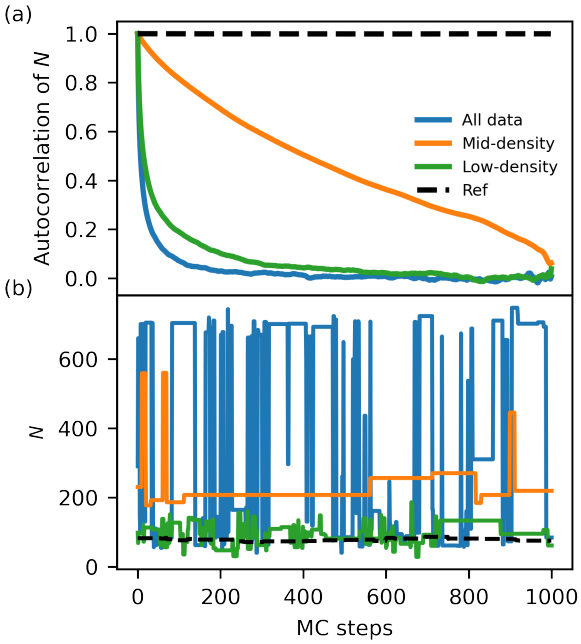

Figure 5: Autocorrelation times are shown for various trained VAE models in (a) with associated sample trajectories in particle number shown in (b). Black-dashed curves represent the reference MC with translation, insertion, and deletion moves, while colored curves are MC simulation based on VAE models. Blue curves are models trained with all lattice gas data (low-, mid-, and high-densities), green with only low-density data (below 200 particles), and orange with only mid-density data (between 200 and 584 particles). All VAE models are trained with autoregressive decoders and a latent space with 1 dimension.

ability of the model — if a specific configurational space is never sampled, the VAE will not place any support of its probability distributions in this region. When trained on only mid-density (low-probability) data, the VAE still proposes moves with a shorter autocorrelation time than the reference (orange curves in Fig. 5), but with a low acceptance rate (Table 1). To efficiently drive sampling beyond that used for training, a biasing function based on the learned latent distribution may be applied. This is demonstrated in Fig. 6b for VAE-based MC moves, using a VAE trained on all of the lattice gas training data. In cases of extremely poor sampling, such as the VAE trained on mid-density data, a combination of VAE-based and traditional MC moves is likely required to achieve improved sampling under the influence of the VAE-learned biasing potential.

Sampling from VAE-based MC moves is far less limited by a VAE's knowledge of temperature than the configurations seen during training. While we demonstrated earlier that our VAE models have no sense of temperature, VAE-based MC moves will still satisfy detailed balance at any temperature, regardless of the model chosen for proposing new configurations. It is likely that the latent space learned at one temperature is still meaningful at another, resulting in rapid barrier crossing, which is borne out in Fig. 6c. Acceptance probabilities, however, are decreased by an order of magnitude to 0.01 when applying a VAE trained at $\beta = 1.77$ to an MC simulation at $\beta = 2.00$, and to 0.006 for simulations at $\beta = 1.60$. Fig. 6c shows, though, that even when started from a poorly sampled and biased distribution of configurations at a different temperature than the VAE-based MC simulation, the correct distributions at other temperatures may be obtained. At the lower temperature in particular, sampling is significantly accelerated compared to MC simulations with standard move sets. Iterative VAE-based MC simulations and training of VAEs at progressively lower temperatures is thus a potentially efficient route for obtaining difficult to sample, low-temperature distributions of configurations.

Regardless of the quality of the training data or resulting VAE model, the nature of the VAE-based MC moves will still preserve detailed balance and produce configurations consistent with the ensemble of interest. This is apparent in Fig. 6a where the input distribution of lattice gas configurations is asymmetric in particle number density due to limited sampling (i.e., the free energy

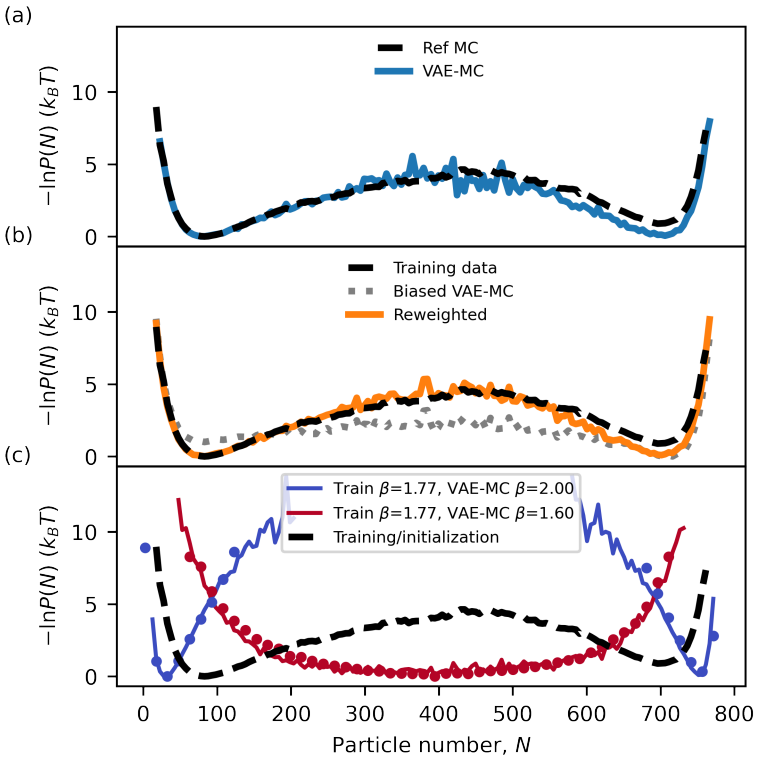

Figure 6: (a) Free energy profiles along particle number for a lattice gas are shown for the reference (training) MC data (black, dashed) and for VAE-based MC moves with an autoregressive model trained on all of the data with a latent dimension of 1 (blue). (b) Uncorrected profile based on the visited states histogram (dotted gray) exhibiting flattened sampling when a bias according to Eq. 12 is added to VAE-based moves. The reweighted profile, correcting for the bias, is shown in orange. (c) VAE-based MC moves can accelerate simulations at temperatures other than that at which they were trained. The dashed curve represents training data at $\beta = 1.77$, which was used to initialize a VAE-based MC simulations at both lower ($\beta = 2.00$) and higher ($\beta = 1.60$) temperatures (blue and red curves), below and above the critical temperature. Correct distributions are recovered at these temperatures, as shown by the circles in each corresponding color, which come from running 200 000 configurations pulled uniformly from 100 independent MC simulations (translation, insertion, and deletion moves for 2 000 000 steps) initiated from random configurations drawn from distributions generated by VAE-based MC at each temperature.

profile is incorrectly tilted), but the results of a VAE-based MC simulation reflect a symmetric distribution, which is expected for the chosen particle-particle interaction and chemical potential. Even though the VAE model generates configurations that closely reproduce the unbalanced sampling of the training data (Fig. 3), detailed balance is enforced through Eq. 9. However, inconsistencies in the training data reflected in the VAE will reduce the acceptance rate, as seen for the case of training on only unlikely, mid-density data. Stated differently, f_{acc} should be indicative of the quality of the VAE model, as it reflects how well the underlying loss objective $\mathcal{P}(\mathbf{x})$ is matched.

Judging model quality based on f_{acc}, however, must be caveated with the fact that MC moves only consider relative probabilities between proposed configurations. Inaccuracies associated with relative frequencies of globally defined macrostates, such as low and high densities of the lattice gas, are not reflected in this metric unless configurations from both such states are proposed. For instance, MC moves based on a VAE trained with only low-density configurations have an acceptance rate of 0.14 (Table 1). This is artificially high because the model only proposes new configurations with low densities and the relative probabilities that the model has learned are accurate for that region of configuration space. Global accuracy across configurations is better probed via the structural metrics presented in Fig. 3, and is generally hard to assess without some prior knowledge of the system at hand. Values of loss functions are specific to the training data set, and only provide estimates of global accuracy if all phase-space is correctly represented.

Despite only tracking relative accuracy of a VAE model, it is interesting to compare f_{acc} for different models and training data sets for the lattice gas in Table 1. Incomplete or biased training data drives the acceptance rate down, as most notably observed by training on mid-density data. Interestingly, increasing the latent dimensionality also lowers acceptance. Reasons for this are unclear. It may be that with 10 latent dimensions, the VAE model becomes capable of memorizing the training data. We noted this earlier as an ability to more accurately capture not only the approximate density, but also the shapes and locations of particle or vacancy clusters (Fig. S12). Subtle overfitting, evident as preferences for specific sizes and locations of particle clusters, is then possible since the training data is generated from a single trajectory that does not fully sample

configurational space. While such bias undoubtedly exists, Table 1 does not reveal large changes in the loss, in particular the reconstruction contribution, between the training data set and an independently generated test set. While the increase in the loss is larger compared to a VAE with a single latent dimension, the reconstruction loss remains lower for the test set and the acceptance rate drops by nearly half.

Table 1: Acceptance rates and loss information for VAE-based MC simulations of a lattice gas. Loss information for training involves the data set used to train each model, while the test data set was generated independently (200 000 configurations were pulled uniformly from 100 trajectories initiated from random samples of the 1D VAE-based MC trajectory, each involving 2 000 000 steps with only translation, insertion, and deletion moves) and is used in all models to generate the test loss information.

Model	Training Data	Latent Dim, D_l	f_{acc}	Train Loss	Train Recon	Train KL	Test Loss	Test Recon	Test KL
Ref MC	N/A	N/A	0.16	N/A	N/A	N/A	N/A	N/A	N/A
Full VAE	all data	1	0.17	225.4	224.0	1.4	231.9	230.3	1.6
Full VAE	low-density	1	0.14	217.9	216.5	1.4	363.2	354.9	8.3
Full VAE	mid-density	1	0.0043	267.2	264.6	2.6	265.8	262.8	3.0
Full VAE	high-density	1	0.048	216.6	215.7	0.9	348.8	341.9	6.9
Full VAE	all data	10	0.091	224.5	213.9	10.6	234.3	222.7	11.6
Full VAE	1D VAE-based MC	1	0.16	225.3	223.9	1.4	231.8	230.4	1.4
Full VAE	10D VAE-based MC	10	0.018	216.6	204.7	11.9	242.5	229.5	13.0

While it is difficult to fully understand the subtle dependence of f_{acc} on training data and latent dimensionality, trajectories generated by VAE-based MC moves do satisfy detailed balance and typically more thoroughly sample phase space due to shorter autocorrelation times (Fig. 5). Using configurations generated by VAE models, we can train new VAEs that are expected to exhibit enhanced probabilistic models and hence improved acceptance rates. Indeed, we find that the resulting models generate more symmetric distributions of particle densities (Fig. S16). However, f_{acc} is not observed to increase over previous models. If configurations over the entire phase space are proposed and detailed balance is observed, then inaccuracies in relative frequencies of global states should result in lower acceptance. We find, however, that the performance of the VAEs trained on VAE-based MC trajectories are highly susceptible to the frequency with which configurations are saved. The number of steps between snapshots should be much longer than the autocorrelation time to prevent multiple of the same configuration entering the training set. More optimal MC procedures may involve both VAE-based moves and local translation, insertion, and

deletion, though we do not pursue such fine-tuning of MC move sets to enhance f_{acc} in this work.

4.2 More realistic molecular and fluid systems

While a lattice gas is discrete and represented well by Bernoulli distributions in the decoder, VAEs can also handle systems with continuous degrees of freedom through autoregressive decoders that predict the means and variances of Gaussian (or other continuous) distributions. We examine the performance of such VAEs trained on various systems, including united-atom *n*-eicosane in vacuum, alanine dipeptide in vacuum, and a 2D Lennard-Jones (LJ) fluid containing a single dimer pair interacting via a double-well potential.[17,49] Model decoder probability distributions and data preprocessing are discussed in Section 3, while full simulation details are provided in the SI. All VAE models employ autoregressive decoders and for all systems we compare VAEs trained with either 1 or 20 latent dimensions.

Table 2: Acceptance rates and loss information for VAE-based MC simulations of various systems over 1 000 000 total attempts. Note that, while the KL divergence term is always positive, the sign and magnitude of the reconstruction loss depend on the form of the decoder probabilistic model and the specific training data, leading to differences between systems.

System	Latent Dim, D_l	f_{acc}	Loss	Recon	KL
n-Eicosane	1	0.000 10	−4.2	−5.3	1.0
	20	0.0093	−10.2	−22.6	12.5
Dialanine	1	0.14	−52.3	−53.3	1.0
	20	0.12	−52.0	−58.2	6.2
2D fluid with dimer	1	0.000 005	65.2	61.7	3.5
	20	0.000 78	52.5	15.6	36.9

Results from VAE-based MC simulations for all of the more realistic systems explored are shown along with training loss information in Table 2. For all systems but dialanine, f_{acc} increases in moving from 1 to 20 latent dimensions. This improvement is always coincident with a lower value of the reconstruction loss function. However, for dialanine, we see that better reconstruction loss does not necessarily result in improved acceptance of VAE-based MC moves. The reduced reconstruction loss with lower f_{acc} and higher total loss is indicative that 20 latent dimensions for dialanine results in slight overfitting. This points to a subtle balance between reconstruction and

KL terms in the loss that will depend on the latent dimensionality and decoding model among other choices of the VAE architecture. Indeed, increasing the weight on the KL term,[62] or specific decompositions of it,[42,63–66] is thought to improve training in some cases and result in more independent and informative latent space coordinates. The reconstruction term drives the model to reproduce input configurations faithfully, while the KL divergence ensures that this is done by encoding a minimal amount of information. An example of this tradeoff is provided in Fig. S17, where it is clear that a VAE with a conditionally independent decoder trained on lattice gas data produces a more complicated latent distribution compared to an autoregressive decoder with a more expressive decoding model.

It is encouraging, however, that the acceptance rates for alanine dipepeptide presented in Table 2 are similar to those for a lattice gas. While VAE-based MC moves enhance sampling in all studied systems through their proposal of non-local transitions, f_{acc} is much lower for *n*-eicosane in vacuum and the 2D LJ fluid. Examining various configurational distributions associated with each system in Fig. 7, we find that low acceptance rates in VAE-based MC moves are driven mainly by an inability of decoding probabilistic models to capture the physics of the system at hand. This was clear in our discussion of switching from conditionally independent to autoregressive models for a lattice gas, and we also show in the SI (Figs. S3-S5, Table S1) that direct prediction of dihedral angles via von Mises distributions, rather than predicting sine-cosine pairs with Gaussians, improves f_{acc}.

In the first column of Fig. 7, we present distributions of the potential energy U, the radius of gyration R_g, and the dihedral angles ϕ for united atom *n*-eicosane. Moving from 1 to 20 latent dimensions results in significant shifting of potential energies to lower values and more accurate reproduction of dihedrals. Even with 20 latent dimensions, however, the trained VAE only accurately reproduces a subset of the marginal distributions and correlations between dihedrals (Fig. 7 and Fig. S4). Since marginal distributions over angles are all approximately Gaussian and independent of other angles and dihedrals (Fig. S4), the decoder should be able to model angular distributions without any latent information and focus instead on dihedrals.

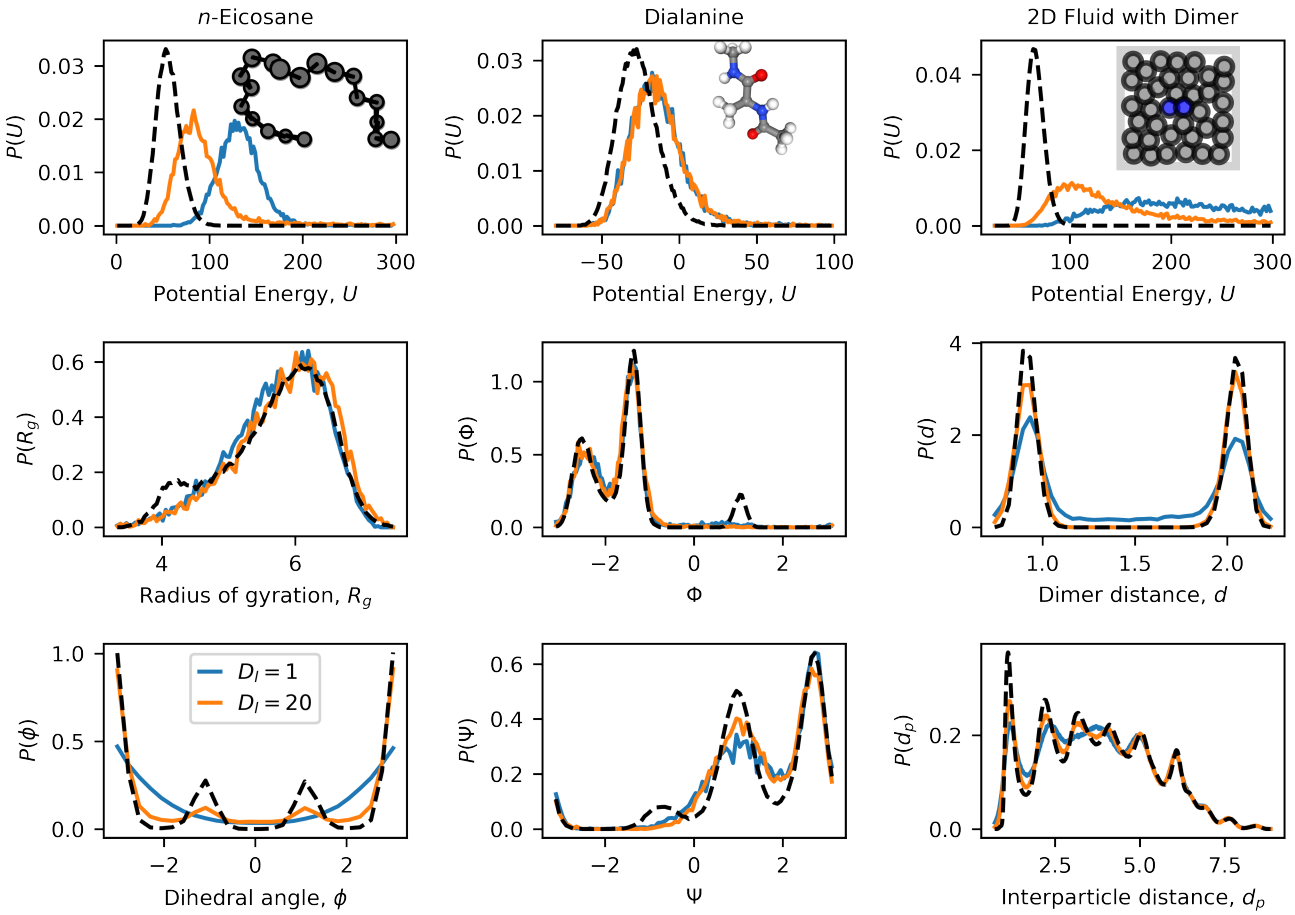

Figure 7: VAE performance in generating potential energy distributions and various structural metrics is shown for (from left to right) *n*-eicosane in vacuum modeled with the TraPPE united atom representation,[56] alanine dipeptide in vacuum modeled with the AMBER99-SB force field,[57] and finally a 2D Lennard-Jones fluid with a single highly attractive dimer as described in Ref. 17. Black dashed curves represent the distribution of the training data for reference. Blue curves are for latent spaces with 1 dimension and orange are for those with 20 dimensions. For *n*-eicosane, the last two rows show distributions for the radius of gyration R_g and all dihedral angles ϕ. Dialanine structural features include the Φ and Ψ backbone dihedrals, and for the 2D fluid we show the distribution of the dimer distance as well as all interparticle distances. Insets in the first row show sample configurational snapshots of each system.

Rather than precisely encoding dihedral values, however, the VAE is driven to learn their underlying distributions by the KL divergence term. That there is difficulty in doing this indicates that, even with a large latent space and autoregressive decoding structure, the probabilistic models may not be able to capture all of the correlations in the high-dimensional dihedral space. Further evidence for this comes in the form of much lower probabilities predicted for configurations with low R_g values. Such collapsed states are induced by attractive LJ interactions between particles[67] that undoubtedly create longer-range, more subtle correlations that complicate training of the decoder model. An inability of the VAE to capture collapsed states, while still capturing some local correlations between dihedrals, suggests that the decoder particularly neglects correlations arising from non-bonded interactions. In contrast, dialanine is much more rigid, with interatomic (LJ and electrostatic) interactions playing a minor role compared to dihedral potentials on the Φ and Ψ torsions. With less opportunities for atomic overlaps and fewer long-range, collective correlations, VAE-generated potential energy and configurational distributions for dialanine better match the training data (Fig. 7, column 2) and f_{acc} is also much higher (Table 2).

In the last column of Fig. 7, we consider a 2D Lennard-Jones fluid with a single particle dimer of differing interactions, as described in Ref.[17]. The single particle pair (in blue in Fig. 7) interacts with a double-well potential with highly attractive basins separated by a high barrier such that dimer dissociation/association represents a slow transition. As such, data is generated by sampling for equal amounts of time in each dimer state, though these two states are not actually equivalent in free energy. The fluid is contained within repulsive walls and is free to move within these confines, though particle trajectories are permuted, centered, and whitened before VAE training, as described in Section 3. A single latent dimension captures distinct associated and dissociated populations, but produces distributions more dissimilar to the reference than twenty-dimensional latent spaces, which only exhibit slight broadening of the low-distance peak. For 20 latent dimensions, the distribution of all interparticle distances (including the dimer) mostly matches the reference. However, closer inspection reveals that small interparticle distance peaks, representing nearest or next-nearest neighbors, are broadened. Such inaccuracies represent particle overlaps that lead to

large potential energies.

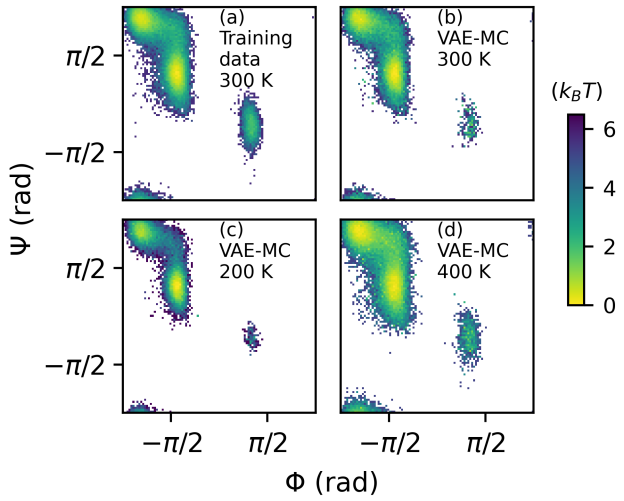

Figure 8: Free energy profiles along Φ and Ψ dihedrals of dialanine in vacuum for (a) training data from MD simulations at 300 K and VAE-based MC with a 1D latent space at (b) 300 K, (c) 200 K, and (d) 400K. VAE-based MC simulations (1000 parallel simulations of 10 000 steps each with configurations saved every 100 steps) recover the expected ensembles at the training temperature and at much lower and higher temperatures.

While results for *n*-eicosane and the 2D LJ fluid demonstrate the difficulties in training VAEs to avoid unrealistic atomic overlaps and to respect collective attractions, VAE-based MC moves can still prove useful in accelerating sampling. Fig. 8 reveals that the free energy profile in the space of the Φ and Ψ backbone dihedrals generated by VAE-based MC matches that from the MD simulation producing training data. Without any additional training, VAE-based MC simulations are also able to generate ensembles at temperatures 100 K lower and higher than the training dataset. Configurations in the small basin around $[\Phi, \Psi] = [\pi/2, -\pi/4]$ are sampled more sparsely compared to the training data due to their low probability in the VAE model and the absence of additional MC moves that sample locally. However, probabilities of these configurations relative to better-sampled free energy basins are only slightly lower than from MD sampling. This is likely attributable to deficiencies in the sampling of the single, long MD simulation, which only experienced 2 transitions to and from the more unlikely states. In contrast, the VAE-based MC simulations experienced many such transitions, achieving better relative global probabilities of

configurations at the expense of less local exploration within the high free energy basin.

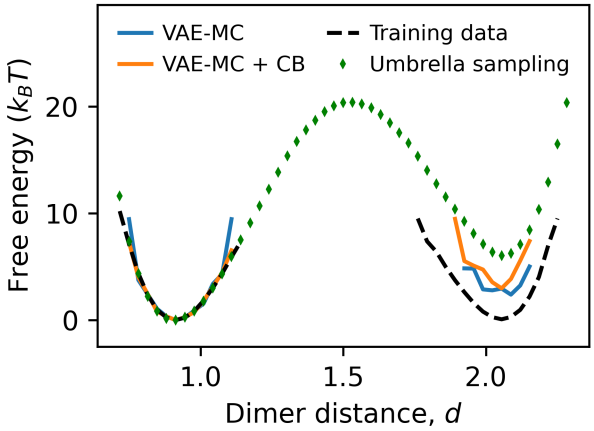

Figure 9: VAE-based MC simulations starting from the training data distribution (dashed black curve) approach the true equilibrium distribution (green diamonds from umbrella sampling, as described in Ref. [17]) for the distance between dimer particles, which places the open state 6 k_BT higher in free energy. Free energy profiles are generated from the second half of 1000 VAE-based MC simulations (20 latent dimensions) run in parallel for 100 000 steps with configurations saved every 1000 steps. Increased acceptance rates in VAE-based MC simulations with configurational bias selection (orange) allow for a closer approach to equilibrium for the same number of MC steps.

Global sampling is accelerated in the 2D LJ fluid in a manner reminiscent of VAE-based MC simulations of the lattice gas. Specifically, the distributions based on equal sampling of the closed and open dimer states used for training are transformed into a preference for the closed state. After many VAE-based MC moves that globally transition the system, we see in Fig. 9 that the free energy profile along dimer distance d approaches that from umbrella sampling. Due to low acceptance rates, this takes many MC trials, though, as described in Appendix A, we find that generating many configurations for each trial and combining VAE-based proposals with configurational bias selection increases acceptance by orders of magnitude. Fig. 9 and Fig. S18 show that these moves speed convergence to the equilibrium distribution. Despite low acceptance rates, VAE-based MC moves are still computationally efficient for recovering correct, equilibrium sampling compared to MD or standard MC simulations, which have difficulty crossing the 20 k_BT barrier separating open and closed states. For systems with high free energy barriers, especially those for which low-

dimensional descriptions are unknown or nonintuitive, VAE-based MC is thus a promising route to accelerate global sampling and refine equilibrium ensembles generated by molecular simulation.

5 Conclusions

The VAE-based MC moves developed here provide a new tool for accelerating molecular simulations. Moving within the VAE-learned information-rich latent space leads to highly collective transitions. By encoding and decoding these transitions, we learn efficient proposal moves for MC simulations. Since the objective of training a VAE is to learn the probability density of the ensemble of interest, the proposal probabilities approximate the configurational probability. As a VAE model more closely matches the true probability density, the acceptance rate of VAE-based MC moves gets closer to unity. However, even if the model is poor, the VAE-based MC moves still satisfy detailed balance and sample the ensemble of interest exactly. This means that we can still fruitfully employ a VAE model to perform VAE-based MC moves that rapidly correct inaccuracies in the training data due to finite sampling. VAE-based MC moves may also be used to efficiently sample at temperatures different from that at which training data was generated. This is particularly helpful when applied to lower-temperature simulations where VAE-based MC moves propose transitions bypassing high free energy barriers. To increase acceptance rates of VAE-based moves, we have clearly shown that a VAE's probabilistic models should more closely match the underlying physics.

It is comforting to know that, regardless of how poorly the VAE model performs, the VAE-based MC moves will sample the intended ensemble — and the degree to which sampling is accelerated depends on the creativity and power put into the probabilistic modeling effort. This is of course true of any MC simulation effort for which the proposed moves satisfy detailed balance. However, we present a framework by which a computer may be used to learn such moves independently. Human creativity and physical understanding is still needed, however, in designing VAE models, in particular decoding distributions. We envision the iterative process of training and

redesigning VAE models as a powerful future paradigm in developing MC move sets.

Developing probabilistic decoding models that correctly account for intermolecular interactions and prevent particle-particle overlaps will be an essential element of transitioning VAE-based MC moves to molecular and fluid systems. We expect that this will not only entail more sophisticated, tailored decoder models, but also involve augmentation of the data input and output by a VAE (e.g., devising probabilistic models that include information about pairwise atomic distances). Neural networks designed to respect translational, rotational, and permutational invariance (or equivariance) of molecular interactions[68,69] also hold promise for improving VAE-based MC moves. In particular, recent innovations in convolutional, autoregressive networks for backmapping with conditional generative adversarial networks[70,71] might be fruitfully employed to enhance decoders in future work. Efficient convolution schemes, applied to both the encoder and decoder, will also be necessary to make models system-size independent. However, convolutions and deep learning on point clouds remain active areas of research in the machine learning community at large.[72–75] Unique modifications, such as the proposed incorporation of equivariant convolutions into VAEs for molecules,[76] will be required to adapt current methods to highly structured, correlated point clouds representing molecular fluids. With such tools in hand, it may become possible to efficiently switch between implicit and explicit solvent models within a single simulation, enabling large-scale investigations of self-assembling systems while recovering the precise role of solvent in driving the assembly process. Many similarly exciting applications exist, presenting unique opportunities to extend the VAE-based MC framework we have developed in this work.

Appendix A Configurational bias selection

For the same VAE model, acceptance rates may be significantly improved by using configurational bias selection of trial configurations. In the case of of the 2D LJ fluid system, acceptance for a VAE model with a 20D latent space can be increased from 0.00078 to 0.0082, an order of magnitude increase. With configurational bias selection for MC moves,[46] M trial configurations are

proposed, where $M = 100$ for the results shown in Fig. 9 and acceptance rates just cited. We generate configurations based on a trained VAE model as described in Section 2.2, passing into, along, and out of the latent space. In reverse, we generate $M-1$ configurations and augment this set with the current configuration to produce M backwards moves. Due to the highly parallel nature of the VAE code running on a GPU, generating many configurations comes virtually free in terms of computational time, only incurring additional memory costs. If potential energy calculations are similarly parallelized, as implemented for the 2D LJ fluid[61] VAE-based proposals with configurational bias selections run at nearly the same speed as standard VAE-based MC moves. We assign weights to each configuration by first grouping together terms in Eq. 9 to approximate the marginal probability of the i^{th} forward or backward configuration under the VAE model

$$P(\mathbf{x}_i;\boldsymbol{\lambda}) \approx \frac{P(\mathbf{x}_i|\mathbf{z}_i;\boldsymbol{\lambda})\,P(\mathbf{z}_i)}{q(\mathbf{z}_i|\mathbf{x}_i;\boldsymbol{\phi})} \tag{17}$$

This keeps the weight on each configuration independent of other configurations and the specific path followed though latent space — in other words, all combinations of forward and backward configurations are possible. Each forward configuration is assigned the configurational bias weight

$$w_{\mathbf{F},i} = \frac{\mathrm{e}^{-\beta U\left(\mathbf{x}_{\mathrm{F},i}\right)}/P(\mathbf{x}_{\mathrm{F},i};\boldsymbol{\lambda})}{W_{\mathrm{F}}} \tag{18}$$

The subscript F indicates that the configuration is from the set generated for the forward trials (B will be used for backward). β is the reciprocal temperature and U is the potential energy. The normalization factor W_{F}, or Rosenbluth factor,[46] is given by

$$W_{\mathrm{F}} = \sum_{i}^{M} \mathrm{e}^{-\beta U\left(\mathbf{x}_{\mathrm{F},i}\right)}/P(\mathbf{x}_{\mathrm{F},i};\boldsymbol{\lambda}) \tag{19}$$

We efficiently select a new configuration in the forward direction based on these weights via the Gumbel-max trick.[77] In the reverse direction, the current configuration is automatically selected,

which leads to the ratio of acceptance probabilities

$$\frac{\alpha_{1\to 2}}{\alpha_{2\to 1}} = \frac{W_{\mathrm{F}}}{W_{\mathrm{B}}} \tag{20}$$

We arrive at this result by noting that the proposal probability in the forward direction involves the probability of generating all forward *and* backward configurations *except* the initial configuration, and in the reverse direction we consider all but the probability of generating the selected new configuration. The probabilities for generating configurations cancel except for the probabilities of generating the current and newly selected configurations. The Boltzmann weight and proposal probabilities of the current and new configurations cancel those in the numerator of the selection weights, resulting in the above acceptance ratio. More involved derivations and explanations of configurational bias moves may be found in Ref.[46], with the use of arbitrary proposal distributions addressed in Ref.[78].

Acknowledgement

This work was performed while J.I.M. held an NRC Postdoctoral Fellowship at the National Institute of Standards and Technology.

Supporting Information Available

Example of training and performing MC moves with relative entropy models with decodings, additional simulation details, additional supporting figures and tables.

References

(1) Noid, W. G. Perspective: Coarse-grained Models for Biomolecular Systems. *The Journal of Chemical Physics* **2013**, *139*, 090901.

(2) Spiwok, V.; Sucur, Z.; Hosek, P. Enhanced Sampling Techniques in Biomolecular Simulations. *Biotechnology Advances* **2015**, *33*, 1130–1140.

(3) Yang, Y. I.; Shao, Q.; Zhang, J.; Yang, L.; Gao, Y. Q. Enhanced Sampling in Molecular Dynamics. *The Journal of Chemical Physics* **2019**, *151*, 070902.

(4) Sanyal, T.; Shell, M. S. Transferable Coarse-Grained Models of Liquid–Liquid Equilibrium Using Local Density Potentials Optimized with the Relative Entropy. *The Journal of Physical Chemistry B* **2018**, *122*, 5678–5693.

(5) Shen, K.; Sherck, N.; Nguyen, M.; Yoo, B.; Köhler, S.; Speros, J.; Delaney, K. T.; Fredrickson, G. H.; Shell, M. S. Learning Composition-Transferable Coarse-Grained Models: Designing External Potential Ensembles to Maximize Thermodynamic Information. *The Journal of Chemical Physics* **2020**, *153*, 154116.

(6) Ben-Amotz, D. Water-Mediated Hydrophobic Interactions. *Annual Review of Physical Chemistry* **2016**, *67*, 617–638.

(7) Chen, W.; Tan, A. R.; Ferguson, A. L. Collective Variable Discovery and Enhanced Sampling Using Autoencoders: Innovations in Network Architecture and Error Function Design. *The Journal of Chemical Physics* **2018**, *149*, 072312.

(8) Ribeiro, J. M. L.; Bravo, P.; Wang, Y.; Tiwary, P. Reweighted Autoencoded Variational Bayes for Enhanced Sampling (RAVE). *The Journal of Chemical Physics* **2018**, *149*, 072301.

(9) Kingma, D. P.; Welling, M. Auto-Encoding Variational Bayes. *arXiv:1312.6114 [cs, stat]* **2014**,

(10) Hernández, C. X.; Wayment-Steele, H. K.; Sultan, M. M.; Husic, B. E.; Pande, V. S. Variational Encoding of Complex Dynamics. *Physical Review E* **2018**, *97*, 062412.

(11) Sultan, M. M.; Wayment-Steele, H. K.; Pande, V. S. Transferable Neural Networks for Enhanced Sampling of Protein Dynamics. *Journal of Chemical Theory and Computation* **2018**, *14*, 1887–1894.

(12) Chen, W.; Sidky, H.; Ferguson, A. L. Capabilities and Limitations of Time-Lagged Autoencoders for Slow Mode Discovery in Dynamical Systems. *The Journal of Chemical Physics* **2019**, *151*, 064123.

(13) Wehmeyer, C.; Noé, F. Time-Lagged Autoencoders: Deep Learning of Slow Collective Variables for Molecular Kinetics. *The Journal of Chemical Physics* **2018**, *148*, 241703.

(14) Wang, D.; Tiwary, P. State Predictive Information Bottleneck. *The Journal of Chemical Physics* **2021**, *154*, 134111.

(15) Papamakarios, G.; Nalisnick, E.; Rezende, D. J.; Mohamed, S.; Lakshminarayanan, B. Normalizing Flows for Probabilistic Modeling and Inference. *Journal of Machine Learning Research* **2021**, *22*, 1–64.

(16) Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Nets. 9.

(17) Noé, F.; Olsson, S.; Köhler, J.; Wu, H. Boltzmann Generators: Sampling Equilibrium States of Many-Body Systems with Deep Learning. *Science* **2019**, *365*, eaaw1147.

(18) Wu, D.; Wang, L.; Zhang, P. Solving Statistical Mechanics Using Variational Autoregressive Networks. *Physical Review Letters* **2019**, *122*, 080602.

(19) Lemke, T.; Peter, C. EncoderMap: Dimensionality Reduction and Generation of Molecule Conformations. *Journal of Chemical Theory and Computation* **2019**, *15*, 1209–1215.

(20) Lemke, T.; Berg, A.; Jain, A.; Peter, C. EncoderMap(II): Visualizing Important Molecular Motions with Improved Generation of Protein Conformations. *Journal of Chemical Information and Modeling* **2019**, *59*, 4550–4560.

(21) Sidky, H.; Chen, W.; Ferguson, A. L. Molecular Latent Space Simulators. *Chemical Science* **2020**, *11*, 9459–9467.

(22) Kofke, D. A. Free Energy Methods in Molecular Simulation. *Fluid Phase Equilibria* **2005**, *228–229*, 41–48.

(23) Monroe, J. I.; Hatch, H. W.; Mahynski, N. A.; Shell, M. S.; Shen, V. K. Extrapolation and Interpolation Strategies for Efficiently Estimating Structural Observables as a Function of Temperature and Density. *The Journal of Chemical Physics* **2020**, *153*, 144101.

(24) Lyman, E.; Zuckerman, D. M. Resolution Exchange Simulation with Incremental Coarsening. *Journal of Chemical Theory and Computation* **2006**, *2*, 656–666.

(25) Lyman, E.; Ytreberg, F. M.; Zuckerman, D. M. Resolution Exchange Simulation. *Physical Review Letters* **2006**, *96*, 028105.

(26) Bandyopadhyay, P. Increasing the Efficiency of Monte Carlo Simulation with Sampling from an Approximate Potential. *Chemical Physics Letters* **2013**, *556*, 341–345.

(27) Gelb, L. D. Monte Carlo Simulations Using Sampling from an Approximate Potential. *The Journal of Chemical Physics* **2003**, *118*, 7747–7750.

(28) Huang, L.; Wang, L. Accelerated Monte Carlo Simulations with Restricted Boltzmann Machines. *Physical Review B* **2017**, *95*, 035105.

(29) Liu, J.; Qi, Y.; Meng, Z. Y.; Fu, L. Self-Learning Monte Carlo Method. *Physical Review B* **2017**, *95*, 041101.

(30) Nagai, Y.; Okumura, M.; Tanaka, A. Self-Learning Monte Carlo Method with Behler-Parrinello Neural Networks. *arXiv:1807.04955 [cond-mat, physics:physics]* **2018**,

(31) Shen, H.; Liu, J.; Fu, L. Self-Learning Monte Carlo with Deep Neural Networks. *Physical Review B* **2018**, *97*, 205140.

(32) Rosenbluth, M. N.; Rosenbluth, A. W. Monte Carlo Calculation of the Average Extension of Molecular Chains. *The Journal of Chemical Physics* **1955**, *23*, 356–359.

(33) Duane, S.; Kennedy, A.; Pendleton, B. J.; Roweth, D. Hybrid Monte Carlo. *Physics Letters B* **1987**, *195*, 216–222.

(34) Wu, D.; Chandler, D.; Smit, B. Electrostatic Analogy for Surfactant Assemblies. *The Journal of Physical Chemistry* **1992**, *96*, 4077–4083.

(35) Doersch, C. Tutorial on Variational Autoencoders. *arXiv:1606.05908 [cs, stat]* **2016**,

(36) van den Oord, A.; Kalchbrenner, N.; Vinyals, O.; Espeholt, L.; Graves, A.; Kavukcuoglu, K. Conditional Image Generation with PixelCNN Decoders. *arXiv:1606.05328 [cs]* **2016**,

(37) Dinh, L.; Sohl-Dickstein, J.; Bengio, S. Density Estimation Using Real NVP. *arXiv:1605.08803 [cs, stat]* **2017**,

(38) Durkan, C.; Bekasov, A.; Murray, I.; Papamakarios, G. Neural Spline Flows. *arXiv:1906.04032 [cs, stat]* **2019**,

(39) Kingma, D. P.; Salimans, T.; Jozefowicz, R.; Chen, X.; Sutskever, I.; Welling, M. Improving Variational Inference with Inverse Autoregressive Flow. *arXiv:1606.04934 [cs, stat]* **2017**,

(40) Rezende, D. J.; Mohamed, S. Variational Inference with Normalizing Flows. *arXiv:1505.05770 [cs, stat]* **2016**,

(41) Chen, X.; Kingma, D. P.; Salimans, T.; Duan, Y.; Dhariwal, P.; Schulman, J.; Sutskever, I.; Abbeel, P. Variational Lossy Autoencoder. *arXiv:1611.02731 [cs, stat]* **2017**,

(42) Hoffman, M. D.; Johnson, M. J. ELBO Surgery: Yet Another Way to Carve up the Variational Evidence Lower Bound. 30th Conference on Neural Information Processing Systems. Barcelona, Spain, 2016; p 4.

(43) Tomczak, J. M.; Welling, M. VAE with a VampPrior. *arXiv:1705.07120 [cs, stat]* **2018**,

(44) Shell, M. S. The Relative Entropy Is Fundamental to Multiscale and Inverse Thermodynamic Problems. *The Journal of Chemical Physics* **2008**, *129*, 144108.

(45) Wang, W.; Gómez-Bombarelli, R. Coarse-Graining Auto-Encoders for Molecular Dynamics. *npj Computational Materials* **2019**, *5*, 125.

(46) Frenkel, D.; Smit, B. *Understanding Molecular Simulation*; Computational Science Series 1; Academic Press, 2002.

(47) Nicoli, K. A.; Nakajima, S.; Strodthoff, N.; Samek, W.; Müller, K.-R.; Kessel, P. Asymptotically Unbiased Estimation of Physical Observables with Neural Samplers. *Physical Review E* **2020**, *101*, 023304.

(48) Wu, H.; Köhler, J.; Noé, F. Stochastic Normalizing Flows. *arXiv:2002.06707 [physics, stat]* **2020**,

(49) Nilmeier, J. P.; Crooks, G. E.; Minh, D. D. L.; Chodera, J. D. Nonequilibrium Candidate Monte Carlo Is an Efficient Tool for Equilibrium Simulation. *Proceedings of the National Academy of Sciences* **2011**, *108*, E1009–E1018.

(50) Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G. S.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Goodfellow, I.; Harp, A.; Irving, G.; Isard, M.; Jia, Y.; Jozefowicz, R.; Kaiser, L.; Kudlur, M.; Levenberg, J.; Mané, D.; Monga, R.; Moore, S.; Murray, D.; Olah, C.; Schuster, M.; Shlens, J.; Steiner, B.; Sutskever, I.; Talwar, K.; Tucker, P.; Vanhoucke, V.; Vasudevan, V.; Viégas, F.; Vinyals, O.; Warden, P.; Wattenberg, M.; Wicke, M.; Yu, Y.; Zheng, X. TensorFlow: Large-scale Machine Learning on Heterogeneous Systems. 2015.

(51) Germain, M.; Gregor, K.; Murray, I.; Larochelle, H. MADE: Masked Autoencoder for Distribution Estimation. *arXiv:1502.03509 [cs, stat]* **2015**,

(52) Dieng, A. B.; Kim, Y.; Rush, A. M.; Blei, D. M. Avoiding Latent Variable Collapse With Generative Skip Models. *arXiv:1807.04863 [cs, stat]* **2019**,

(53) Kingma, D. P.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]* **2017**,

(54) Sønderby, C. K.; Raiko, T.; Maaløe, L.; Sønderby, S. K.; Winther, O. Ladder Variational Autoencoders. *arXiv:1602.02282 [cs, stat]* **2016**,

(55) Eastman, P.; Swails, J.; Chodera, J. D.; McGibbon, R. T.; Zhao, Y.; Beauchamp, K. A.; Wang, L.-P.; Simmonett, A. C.; Harrigan, M. P.; Stern, C. D.; Wiewiora, R. P.; Brooks, B. R.; Pande, V. S. OpenMM 7: Rapid Development of High Performance Algorithms for Molecular Dynamics. *PLOS Computational Biology* **2017**, *13*, e1005659.

(56) Martin, M. G.; Siepmann, J. I. Transferable Potentials for Phase Equilibria. 1. United-Atom Description of n-Alkanes. *J. Phys. Chem. B* **1998**, *102*, 2569–2577.

(57) Hornak, V.; Abel, R.; Okur, A.; Strockbine, B.; Roitberg, A.; Simmerling, C. Comparison of Multiple Amber Force Fields and Development of Improved Protein Backbone Parameters. *Proteins: Structure, Function, and Bioinformatics* **2006**, *65*, 712–725.

(58) Michaud-Agrawal, N.; Denning, E. J.; Woolf, T. B.; Beckstein, O. MDAnalysis: A Toolkit for the Analysis of Molecular Dynamics Simulations. *Journal of Computational Chemistry* **2011**, *32*, 2319–2327.

(59) Minh, D. D. L. Alchemical Grid Dock (AlGDock): Binding Free Energy Calculations between Flexible Ligands and Rigid Receptors. *Journal of Computational Chemistry* **2020**, *41*, 715–730.

(60) Gowers, R. J.; Linke, M.; Barnoud, J.; Reddy, T. J. E.; Melo, M. N.; Seyler, S. L.; Domański, J.; Dotson, D. L.; Buchoux, S.; Kenney, I. M.; Beckstein, O. MDAnalysis: A Python

Package for the Rapid Analysis of Molecular Dynamics Simulations. Proceedings of the 15th Python in Science Conference. 2016; pp 98–105.

(61) Noé, F.; Olsson, S.; Kohler, J.; Hao, W. Boltzmann Generators – Sampling Equilibrium States of Many-Body Systems with Deep Learning. http://doi.org/10.5281/zenodo.3242635, 2021; Accessed 02/25/2021.

(62) Higgins, I.; Matthey, L.; Pal, A.; Burgess, C.; Glorot, X.; Botvinick, M.; Mohamed, S.; Lerchner, A. β-VAE: LEARNING BASIC VISUAL CONCEPTS WITH A CONSTRAINED VARIATIONAL FRAMEWORK. **2017**, 22.

(63) Chen, R. T. Q.; Li, X.; Grosse, R.; Duvenaud, D. Isolating Sources of Disentanglement in Variational Autoencoders. *arXiv:1802.04942 [cs, stat]* **2019**,

(64) Kim, H.; Mnih, A. Disentangling by Factorising. *arXiv:1802.05983 [cs, stat]* **2019**,

(65) Kumar, A.; Sattigeri, P.; Balakrishnan, A. VARIATIONAL INFERENCE OF DISENTANGLED LATENT CONCEPTS FROM UNLABELED OBSERVATIONS. **2018**, 16.

(66) Zhao, S.; Song, J.; Ermon, S. InfoVAE: Information Maximizing Variational Autoencoders. *arXiv:1706.02262 [cs, stat]* **2018**,

(67) Ferguson, A. L.; Debenedetti, P. G.; Panagiotopoulos, A. Z. Solubility and Molecular Conformations of *n* -Alkane Chains in Water. *The Journal of Physical Chemistry B* **2009**, *113*, 6405–6414.

(68) Gebauer, N. W. A.; Gastegger, M.; Schütt, K. T. Generating Equilibrium Molecules with Deep Neural Networks. *arXiv:1810.11347 [physics, stat]* **2018**,

(69) Schütt, K. T.; Kindermans, P.-J.; Sauceda, H. E.; Chmiela, S.; Tkatchenko, A.; Müller, K.-R. SchNet: A Continuous-Filter Convolutional Neural Network for Modeling Quantum Interactions. *arXiv:1706.08566 [physics, stat]* **2017**,

(70) Stieffenhofer, M.; Wand, M.; Bereau, T. Adversarial Reverse Mapping of Equilibrated Condensed-Phase Molecular Structures. *Machine Learning: Science and Technology* **2020**, *1*, 045014.

(71) Stieffenhofer, M.; Bereau, T.; Wand, M. Adversarial Reverse Mapping of Condensed-Phase Molecular Structures: Chemical Transferability. *APL Materials* **2021**, *9*, 031107.

(72) Hui, L.; Xu, R.; Xie, J.; Qian, J.; Yang, J. Progressive Point Cloud Deconvolution Generation Network. *arXiv:2007.05361 [cs]* **2020**,

(73) Kim, J.; Yoo, J.; Lee, J.; Hong, S. SetVAE: Learning Hierarchical Composition for Generative Modeling of Set-Structured Data. *arXiv:2103.15619 [cs]* **2021**,

(74) Qi, C. R.; Su, H.; Mo, K.; Guibas, L. J. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. *arXiv:1612.00593 [cs]* **2017**,

(75) Wu, W.; Qi, Z.; Fuxin, L. PointConv: Deep Convolutional Networks on 3D Point Clouds. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* **2019**, 9621–9630.

(76) Wang, W.; Xu, M.; Cai, C.; Miller, B. K.; Smidt, T.; Wang, Y.; Tang, J.; Gómez-Bombarelli, R. Generative Coarse-Graining of Molecular Conformations. *arXiv:2201.12176 [physics]* **2022**,

(77) Gumbel, E. J. Statistical Theory of Extreme Values and Some Practical Applications: A Series of Lectures. *National Bureau of Standards Applied Math Series* **1954**, *33*.

(78) Martin, M. G.; Frischknecht, A. L. Using Arbitrary Trial Distributions to Improve Intramolecular Sampling in Configurational-Bias Monte Carlo. *Molecular Physics* **2006**, *104*, 2439–2456.

Graphical TOC Entry