Adaptive Monte Carlo augmented with normalizing flows

Marylou Gabrié[a,b,1], Grant M. Rotskoff[c], and Eric Vanden-Eijnden[d,1]

[a]Center for Computational Mathematics, Flatiron Institute, New York, NY 10010; [b]Center for Data Science, New York University, New York, NY 10011; [c]Department of Chemistry, Stanford University, Stanford, CA 94305; and [d]Courant Institute of Mathematical Sciences, New York University, New York, NY 10012

Edited by Frank Noé, Mathematics and Computer Science, Freie Universitat Berlin, Berlin, Germany; received May 24, 2021; accepted December 19, 2021 by Editorial Board Member Daan Frenkel

Many problems in the physical sciences, machine learning, and statistical inference necessitate sampling from a high-dimensional, multimodal probability distribution. Markov Chain Monte Carlo (MCMC) algorithms, the ubiquitous tool for this task, typically rely on random local updates to propagate configurations of a given system in a way that ensures that generated configurations will be distributed according to a target probability distribution asymptotically. In high-dimensional settings with multiple relevant metastable basins, local approaches require either immense computational effort or intricately designed importance sampling strategies to capture information about, for example, the relative populations of such basins. Here, we analyze an adaptive MCMC, which augments MCMC sampling with nonlocal transition kernels parameterized with generative models known as normalizing flows. We focus on a setting where there are no preexisting data, as is commonly the case for problems in which MCMC is used. Our method uses 1) an MCMC strategy that blends local moves obtained from any standard transition kernel with those from a generative model to accelerate the sampling and 2) the data generated this way to adapt the generative model and improve its efficacy in the MCMC algorithm. We provide a theoretical analysis of the convergence properties of this algorithm and investigate numerically its efficiency, in particular in terms of its propensity to equilibrate fast between metastable modes whose rough location is known a priori but respective probability weight is not. We show that our algorithm can sample effectively across large free energy barriers, providing dramatic accelerations relative to traditional MCMC algorithms.

Monte Carlo | normalizing flows | free energy calculations | phase transitions

Monte Carlo approximations are the method of choice to extract information from high-dimensional probability distributions encountered in the description of natural systems and statistical models. One generic feature of these distributions that is particularly challenging for sampling is multimodality (or metastability): that is, when low-probability regions separate high-probability regions (or basins) of the state space. Markov Chain Monte Carlo (MCMC) algorithms, which are driven primarily by local dynamics such as Hamiltonian Monte Carlo or Langevin dynamics, typically struggle to transition between metastable basins, leading to either extremely long correlation times along the chains and few effective independent samples or even failure to equilibrate at all. As a result, slow relaxation and metastability plague sampling problems that arise in chemistry and biophysics (1).

On the other hand, generative models, which have garnered much attention in the machine learning literature, seem to efficiently sample complicated high-dimensional distributions, such as collections of images. Most of these generative models, including generative adversarial networks (2) and variational autoencoders (3), rely on the transformation of samples from a simple and tractable base distribution through a map parametrized with neural networks. After learning, the map transforms samples from the base to mimic samples of a given empirical distribution. This formulation allows for drawing independent samples from the model at a negligible cost. However, the conventional strategy for training a generative model requires an extensive dataset of samples. Arguably, these models have succeeded most dramatically in domains where the cost of generating and curating data are comparatively low (e.g., image recognition) (2, 4, 5).

In scientific computing applications, obtaining data from the distribution is the primary goal. Furthermore, the quality metrics used in traditional machine learning applications are a priori quite different from the efficacy and precision in sampling a target distribution. Hence, it is natural to ask whether traditional MCMC methods and generative models can be successfully combined to accelerate sampling of complicated high-dimensional distributions?

The prospect of enhancing sampling with suitable generative models is an active area of inquiry (4, 6–11). In particular, sampling via Metropolis–Hastings MCMC requires the computation of each transition generation probability and its inverse. As a result, the model architectures on which most generative neural networks rely are not conducive to Metropolis–Hasting MCMC. However, specific classes of neural networks have been designed with this in mind, allowing for efficient estimates of the probability of a generated sample, including autoregressive models (12) and normalizing flows (NFs), which are expressive

Significance

Monte Carlo methods, tools for sampling data from probability distributions, are widely used in the physical sciences, applied mathematics, and Bayesian statistics. Nevertheless, there are many situations in which it is computationally prohibitive to use Monte Carlo due to slow "mixing" between modes of a distribution unless hand-tuned algorithms are used to accelerate the scheme. Machine learning techniques based on generative models offer a compelling alternative to the challenge of designing efficient schemes for a specific system. Here, we formalize Monte Carlo augmented with normalizing flows and show that, with limited prior data and a physically inspired algorithm, we can substantially accelerate sampling with generative models.

Author contributions: M.G., G.M.R., and E.V.-E. designed research, performed research, and wrote the paper.

The authors declare no competing interest.

This article is a PNAS Direct Submission. F.N. is a guest editor invited by the Editorial Board.

[1]To whom correspondence may be addressed. Email: marylou.gabrie@polytechnique.edu or eve2@cims.nyu.edu.

This article contains supporting information online at https://www.pnas.org/lookup/suppl/doi:10.1073/pnas.2109420119/-/DCSupplemental.

Published March 2, 2022.

invertible function representations (13, 14). At this point, NFs have been investigated as transition operators in MCMC algorithms and variational ansatz in a variety of contexts in the physical sciences and Bayesian applications (13, 15–20). These methods offer a promising speedup for sampling unimodal distributions without requiring preexisting data samples by relying on a *self-training* objective for the map (described in detail in *SI Appendix*, section S1). However, the multimodal case requires prior knowledge about either the symmetries of the systems generating the degeneracy of the modes (20) or the location of the metastable basins (17). This necessity was noted in the influential work of Noé et al. (21) that proposes a training strategy for NFs to generate low-energy configurations, which are subsequently reweighted.

The aim of this paper is to propose an alternative class of adaptive MCMC algorithms augmented with an NF trained on the fly with the generated samples and also, to carefully assess the prospects of these algorithms for accelerating sampling in cases where no extensive preexisting dataset is available. Our main contributions are as follows.

- We introduce an adaptive Metropolis–Hastings MCMC algorithm that augments a chain performing local steps with nonlocal resampling steps proposed by an NF. The corresponding proposal distribution is adapted along sampling by training the NF via optimization of a forward Kullback–Leibler divergence estimated on the generated data.
- As the adaptation of the map depends on the history of the chains, the convergence of the proposed algorithm, where training and sampling happen simultaneously, is not trivial. We show that the adaptive algorithm is akin to a nonlinear MCMC scheme (22), which we analyze in the continuous-time limit. In this limit, we show that the algorithm converges asymptotically with an exponential rate that can be explicitly estimated.
- We test this adaptive MCMC approach on complex examples in high dimension (random fields, transition paths, and interacting particle systems at phase coexistence) and show that it dramatically accelerates the sampling. In particular, we estimate the relative statistical weights of metastable states efficiently without constructing a specific pathway between the basins of interest.

Our results also emphasize some key determinants for the success of sampling augmented with learning.

- One representative configuration per mode of interest in the target distribution must be known beforehand to initialize the chains. We critically assess the ability of using generative model proposals to discover unknown metastable states and show that this prospect is statistically unlikely without prior information about these states.
- Blending generative sampling with a standard MCMC strategy is typically required to guarantee good sampling of the target distribution; in particular, we show that relying on generative sampling alone may not be sufficient because it requires that we learn the generative model to a degree of accuracy that is hard to achieve in practice, especially in high-dimensional examples.
- Finally, our analysis and numerical experiments show how scaling to high dimensions is also facilitated by parametrizations of NFs that incorporate known structures of the target distributions, such as short-scale correlations. The possibility to inform the map, or the base distribution, with physical intuition alleviates the curse of dimensionality that would prevent general-purpose parametrizations from reaching the required level of precision with reasonably sized models as the dimension grows.

Design Challenges in MCMC Methods

The goal of sampling is to generate configurations $x \in \Omega \subset \mathbb{R}^d$ in proportion to some probability measure $\nu_*(dx) = \rho_*(x)dx$, which we assume has probability density function ρ_*. In physical systems, we typically write this in Boltzmann form:

$$\rho_*(x) = Z_*^{-1} e^{-U_*(x)}, \qquad [1]$$

where $U_* \propto -\log \rho_*$ is the potential energy function for the system. We assume that we have an explicit model for U_* and can efficiently evaluate this energy function, although we may have little a priori information about the distribution of configurations associated with this energy and in general, do not know the normalization constant Z_*.

MCMC algorithms avoid computing Z_* by generating a sequence $\{x(k)\}_{k\in\mathbb{N}}$ of configurations with a transition kernel $\pi(x, y)$ with $\int_\Omega \pi(x, y)dy = 1$ for all $x \in \Omega$, which quantifies the conditional probability density of a transition from state x into state y. Assume that the kernel $\pi(x, y)$ is irreducible and aperiodic (23) and satisfies the detailed balance relation

$$\rho_*(x)\pi(x, y) = \rho_*(y)\pi(y, x). \qquad [2]$$

Then, the sequence $\{x(k)\}_{k\in\mathbb{N}}$ will sample the target density ρ_* in the sense that the empirical average of any suitable observable ϕ converges to its expectation over ρ_*: that is,

$$\lim_{N\to\infty} \frac{1}{N} \sum_{k=1}^{N} \phi(x(k)) = \int_\Omega \phi(x)\rho_*(x)dx. \qquad [3]$$

Designing a transition kernel π leading to fast convergence of the series in **[3]** is a generically challenging task for MCMC algorithms. In Metropolis–Hastings MCMC, one constructs a proposal distribution that creates new samples that are then accepted or rejected according to a criterion that maintains **[2]**. For example, in the Metropolis-adjusted Langevin algorithm (MALA) (24), new configurations are proposed by approximating the solution of the Langevin equation propagated on a fixed time interval.

Metropolis–Hastings MCMC algorithms, however, involve a trade-off between two requirements that are hard to fulfill simultaneously. Proposal distributions using local dynamics like MALA suffer from long decorrelation times when there is metastability in the target density ρ_*. At the same time, seeking faster mixing times with nonlocal proposal distributions requires careful design to avoid high rejection rates. Recent work in the machine learning literature has suggested a *data-driven* approach to constructing the transition kernel (4, 8, 9) that aids in this design challenge; these approaches originally were pioneered in the context of adaptive and nonlinear MCMC algorithms (22, 25–27). Here, we explore the use of NFs to adaptively parameterize the transition kernel.

MCMC Sampling with NFs

An NF is an invertible map $T : \Omega \to \Omega$ that is optimized to transport samples from a base measure $\nu_B(dx) = \rho_B(x)dx$ (for example, a Gaussian with unit variance) to a given target distribution (14). The goal is to produce a map T_* with inverse $\bar{T}_*$ such that an expectation of an observable with respect to ρ_* can be estimated by transforming samples from the base density to the target: that is, if x_B is drawn from $\rho_B(x)$, then $T_*(x_B)$ is a sample from $\rho_*(x)$ so that for any suitable observable $\mathcal{O}$, we have

$$\int_\Omega \mathcal{O}(T_*(x))\rho_B(x)dx = \int_\Omega \mathcal{O}(x)\rho_*(x)dx. \qquad [4]$$

The existence of such a map T_* is guaranteed under the general conditions on ρ_* and ρ_B investigated (e.g., in the context of

optimal transport theory) (28, 29). Of course, in practice we do not have direct access to this ideal map T_*. Next, we discuss how any approximation T of T_* can in principle be used to perform exact sampling of the target via Metropolis–Hastings MCMC and how the map T can be improved via training.

Metropolis–Hastings MCMC with NF. Throughout, we denote the push forward of ρ_{B} under the map T simply by $\hat{\rho}$; it has the explicit form

$$\hat{\rho}(x) = \rho_{\mathrm{B}}(\bar{T}(x)) \det \left| \nabla_x \bar{T} \right|, \qquad [5]$$

where $\bar{T}$ denotes the inverse map [i.e., $\bar{T}(T(x)) = T(\bar{T}(x)) = x$]. In practice, the parametrization of the map T must be designed carefully to evaluate this density efficiently, requiring easily estimable Jacobian determinants and inverses. This issue has been one of the main foci in the NF literature (14) and is, for instance, solved using coupling layers (30, 31). Even if the map T is not the optimal T_* [i.e., $\hat{\rho}(x) \neq \rho_*(x)$], as long as $\hat{\rho}$ and ρ_* are either both positive or both zero at any point $x \in \Omega$, we can still generate configurations using T with the correct statistical weight in the target distribution by using a Metropolis–Hastings MCMC algorithm with an accept–reject step; a proposed configuration $y = T(x_{\mathrm{B}})$ from a given configuration x is accepted with probability

$$\mathrm{acc}(x, y) = \min \left[1, \frac{\hat{\rho}(x)\rho_*(y)}{\rho_*(x)\hat{\rho}(y)} \right]. \qquad [6]$$

This procedure is equivalent to using the transition kernel

$$\pi_T(x, y) = \mathrm{acc}(x, y)\hat{\rho}(y) + \big(1 - r(x)\big)\delta(x - y), \qquad [7]$$

where $r(x) = \int_\Omega \mathrm{acc}(x, y)\hat{\rho}(y)\, dy$. The formula in Eq. **6** for the acceptance probability emphasizes that if the generated configurations do not have appreciable statistical weight in the target distribution [i.e., $\rho_*(y)$ is very small], few configurations will be accepted. This problem can become fundamental in high-dimensional spaces because unless care is taken to ensure otherwise, the push-forward measure and the target will not overlap (a discussion of this issue and a precise measure-theoretic formulation of MCMC with NF are in *SI Appendix*, section S2). In contrast, when the map yields an appreciable acceptance rate, the flow-based proposals may mix much faster than proposals based on local moves as independent configurations y can be directly sampled from $\hat{\rho}$. We illustrate these features in numerical experiments presented below.

Map Training. Improving the map T requires that we optimize some objective function measuring the discrepancy between the $\hat{\rho}(x)$ and $\rho_*(x)$: for example, the Kullback–Leibler (KL) divergence of ρ_* with respect to $\hat{\rho}$ that is given by an expectation over ρ_*,

$$D_{\mathrm{KL}}(\rho_* \| \hat{\rho}) = C_* - \int_\Omega \log \hat{\rho}(x)\rho_*(x)\, dx, \qquad [8]$$

where $C_* = \int_\Omega \log \rho_*(x)\rho_*(x)\, dx$ is a constant irrelevant for the optimization of this objective over T. Typically, this procedure is used in situations where a dataset from ρ_* is available beforehand (4, 5) and can be used to construct an empirical approximation of Eq. **8**; in contrast, we are focused on situations where only limited data exist initially. In this context, it has been suggested (13, 15, 21) to use the reverse KL divergence of $\hat{\rho}$ with respect to ρ_* since it can be expressed as an expectation over $\hat{\rho}$:

$$D_{\mathrm{KL}}(\hat{\rho} \| \rho_*) = -\log Z_* + \int_\Omega [U_*(x) + \log \hat{\rho}(x)]\hat{\rho}(x)\, dx. \qquad [9]$$

The (unknown) constant $\log Z_*$ is irrelevant for the optimization of this objective over T. This approach seems to alleviate altogether the need of preexisting samples from ρ_*; however, it rests on the possibility to discover relevant regions on ρ_* via sampling $\hat{\rho}$. In practice, this may be very hard to achieve unless we have a good estimate of the ideal T_* to begin with, which is typically not the case; for this reason, here we will resort to optimizing an approximation of the direct KL in Eq. **8**. This procedure, described in the next section, relies on a dynamical estimate of the forward KL divergence that uses data generated via an adaptive MCMC that synergistically takes advantage of the learning to produce samples of the target ρ_* efficiently.

We stress that once the map T becomes accurate enough, Eqs. **8** and **9** can also be combined for further training, as was done, for example, in the related context of Boltzmann generators (21) (for a road map of the different possible strategies to train T, we refer the reader to *SI Appendix*, section S1). We also stress that trainable generative models other than NFs can be used as well, as long as they offer an easy way to sample some $\hat{\rho}$ that can be adapted to the target ρ_*; this feature is illustrated in the numerical examples presented below.

Adaptive MCMC: Concurrent Sampling and Training

The adaptive MCMC we propose concurrently acquires data by combining a local sampler with a nonlocal one based on an NF and uses these data to further optimize the flow. This procedure is summarized in *Algorithm 1* with MALA as the local MCMC algorithm, and it involves the following components.

Sampling. Our algorithm combines MCMC steps using a local kernel π with those obtained using the NF kernel π_T in Eq. **7**. Assuming for simplicity that we make consecutive steps with each kernel, the algorithm uses the compounded kernel

$$\hat{\pi}(x, y) = \int_\Omega \pi(x, z)\pi_T(z, y)\, dz, \qquad [10]$$

which satisfies the detailed balance relation Eq. **2** because the transitions kernels π and π_T individually do. While the flow-based kernel π_T allows global mixing between modes once T is sufficiently optimized, alternating with the local kernel π improves the robustness of the scheme by ensuring that sampling proceeds in places within the modes where the map is not optimal. This is useful during the first iterations of the scheme

Algorithm 1 Adaptive MCMC: concurrent MCMC sampling and map training.

SAMPLETRAIN(U_*, T, $\{x_i(0)\}_{i=1}^n$, τ, $k_{\max}$, k_{Lang}, ϵ)
Inputs: U_* target energy, T initial map, $\{x_i(0)\}_{i=1}^n$ initial data, $\tau > 0$ time step, $k_{\max} \in \mathbb{N}$ total duration, $k_{\mathrm{Lang}} \in \mathbb{N}$ number of Langevin steps per resampling step, $\epsilon > 0$ map training time step
$k = 0$
while $k < k_{\max}$ **do**
  for $i = 1, \ldots, n$ **do**
    if $k \mod k_{\mathrm{Lang}} + 1 = 0$ **then**
      $x'_{\mathrm{B},i} \sim \rho_{\mathrm{B}}$
      $x'_i = T(x'_{\mathrm{B},i})$ ▷ push-forward via T
      $x_i(k+1) = x'_i$ with probability $\mathrm{acc}(x_i(k), x'_i)$, otherwise $x_i(k+1) = x_i(k)$ ▷ resampling step
    else
      $x'_i = x_i(k) - \tau \nabla U_*(x_i(k)) + \sqrt{2\tau}\,\eta_i$ with $\eta_i \sim \mathcal{N}(0_d, I_d)$ ▷ discretized Langevin step
      $x_i(k+1) = x'_i$ with MALA acceptance probability or ULA, otherwise $x_i(k+1) = x_i(k)$
  $k \leftarrow k + 1$
  $\mathcal{L}[T] = -\frac{1}{n}\sum_{i=1}^n \log \hat{\rho}(x_i(k))$ ▷ evaluate $D_{\mathrm{KL}}(\rho_t \| \hat{\rho})$ on sampled data
  $T \leftarrow T - \epsilon \nabla \mathcal{L}[T]$ ▷ Update the map
return: $\{x_i(k)\}_{k=0,i=1}^{k_{\max},n}$, T

when the map T is almost untrained as well as once training has converged, if the expressiveness of the map parametrization is not sufficient to capture all the features of the target distribution. In *SI Appendix*, section S7.2, we demonstrate numerically the benefit of retaining local components to the sampling scheme (*SI Appendix*, Fig. S4). Let us also note that the convergence rate of a chain using $\hat{\pi}(x, y)$ is necessarily faster than that of MCMC using $\pi(x, z)$ or $\pi_T(z, y)$ individually; if we assume the existence of a spectral gap for both π and π_T and denote the leading eigenvalues of these kernels by $\hat{\lambda} < 1$, $\lambda < 1$, and $\lambda_T < 1$, respectively, we have $\hat{\lambda} \leq \lambda\lambda_T$. While we employ MALA here, any detailed balance MCMC method could be used in Eq. **10**. The transition kernel π does not need to be local; it should, however, have satisfactory acceptance rates. Note that in the experiments that follow, we used ULA because the time steps were sufficiently small to ensure a high acceptance rate.

Adaptation. The kernel π_T of *Algorithm 1* is adapted by using the newly sampled configurations as data to optimize the parameters of the NF T. Denoting by ρ_k the probability density of the chain with kernel $\hat{\pi}$ after $k \in \mathbb{N}$ steps from initialization ρ_0, we minimize the KL divergence of ρ_k with respect to $\hat{\rho}$, $D_{\mathrm{KL}}(\rho_k \| \hat{\rho})$, instead of the KL divergence of the unknown ρ_* with respect to $\hat{\rho}$ as in Eq. **8**. Denoting by $\{x_i(k)\}_{i=1}^n$ the sample of n chains after $k \in \mathbb{N}$ steps of MCMC, this amounts to using the following consistent estimator for $D_{\mathrm{KL}}(\rho_k \| \hat{\rho})$ up to an irrelevant constant:

$$\begin{aligned}\mathcal{L}_n[T] &= -\frac{1}{n}\sum_{i=1}^{n} \log \hat{\rho}(x_i(k)) \\ &= \frac{1}{n}\sum_{i=1}^{n} \big(U_{\mathrm{B}}(\bar{T}(x_i(k)) - \log\det|\nabla \bar{T}(x_i(k))|\big).\end{aligned} \qquad [11]$$

In practice, we use stochastic gradient descent on this loss function to update the parameters of the NF (*Algorithm 1*, line 11). While the expression for the loss is written at iteration k, we can average gradients over multiple MCMC steps. Details of the map parametrizations and training procedures for the experiments presented in the next sections are described in *SI Appendix*, section S6.

Initialization. To start the MCMC chains, we assume that we have configurations $\{x_i(0)\}_{i=1}^n$ in the different modes of the target, but they are not necessarily drawn from ρ_*. We emphasize that the method, therefore, applies in situations where the locations of the metastable states of interest are known a priori, and one should not expect the procedure to find states in basins distinct from initialization. We demonstrate that it is unlikely that the adaptive MCMC will discover new metastable basins without any initial information about their location in *SI Appendix*, section S7.1 on the example of a Gaussian mixture model (*SI Appendix*, Fig. S1) and in *SI Appendix*, section S5 for the random-field example discussed below.

We initialize the map T as the identity transformation and propagate the initial data using $\hat{\pi}$. The initial sampling is essentially driven by the local MCMC, here Langevin dynamics, as the map is not adapted to the target. As the map improves, nonlocal moves start to be accepted, the autocorrelation time drops, and the Markov chains reallocate mass in proportion to the statistical weights of the different basins. These features are illustrated in *SI Appendix*, Fig. S1 in the context of the Gaussian mixture model discussed in *SI Appendix*, section S7.1 and in Figs. 1 and 2 in the context of the random-field example discussed below.

Convergence. Two important questions arise regarding *Algorithm 1*. First, does this scheme produce samples that converge in distribution toward the target, and if so, does the adaptive training of the map T improve the rate of convergence to the target distribution? To analyze the properties of a transition operator that combines nonlocal moves with the NF and a local MCMC algorithm, we consider our approach in the continuous-time limit. In this limit, when using Langevin dynamics as local sampler, the density of the evolving ρ_t with respect to the target ρ_*, defined as $g_t = \rho_t/\rho_*$, satisfies

$$\begin{aligned}\partial_t g_t = &-\nabla U_* \cdot \nabla g_t + \Delta g_t \\ &+ \alpha \int_\Omega \min(\hat{g}_t(x), \hat{g}_t(y))\,(g_t(y) - g_t(x))\,\rho_*(y)\,dy'\end{aligned} \qquad [12]$$

where $\hat{g}_t = \hat{\rho}_t/\rho_*$ and $\alpha \geq 0$ is an adjustable parameter that measures the balance between the Langevin and the resampling parts of the dynamics. Setting $\alpha = 0$ amounts to using the Langevin dynamics alone; in that case, for any initial condition ρ_0, we have that $\rho_t \to \rho_*$ (i.e., $g_t \to 1$) as $t \to \infty$, but this convergence will be exponentially slow in general (32). The situation changes if we include the resampling step (i.e., consider Eq. **12** with $\alpha > 0$). In *SI Appendix*, section S3, under various assumptions about $\hat{g}_t$, we derive convergence rates under the dynamics in Eq. **12** for the Pearson χ^2 divergence of ρ_t with respect to ρ_*, which we denote as

$$D_t = \int_\Omega \frac{\rho_t^2}{\rho_*}\,dx - 1 = \int_\Omega g_t^2\,\rho_*\,dx - 1 \geq 0. \qquad [13]$$

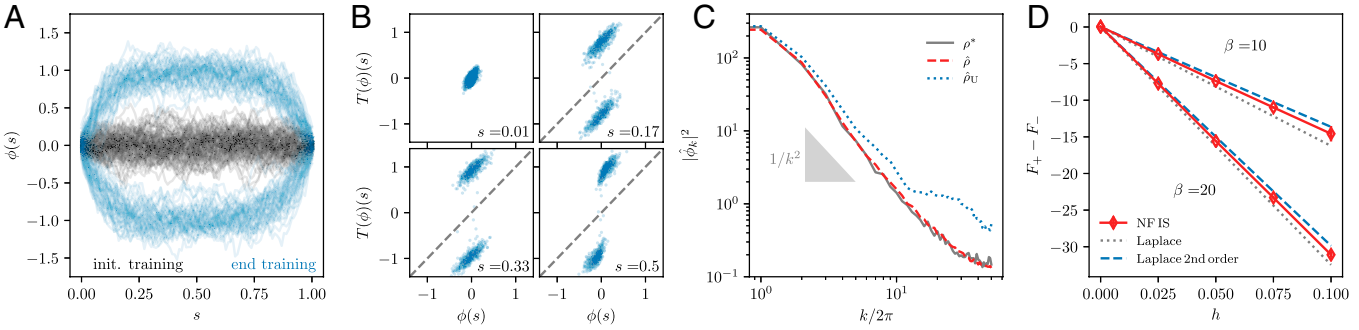

Fig. 1. Sampling metastable states of the stochastic Allen–Cahn model with Langevin dynamics augmented with an NF. (*A*) Configurations obtained by pushing independent samples from the informed base measure Eq. **18** through the flow T at the beginning (black) and at the end of training (blue). Around $\sim 60\%$ of generated configurations are accepted according to the Metropolis–Hasting criteria. (*B*) The learned map T is local in space. (*C*) Fourier spectrum of the target samples: samples from a flow with informed base measure and uniformed base measure. An informed base measure is necessary to capture the higher-frequency features of the target density. (*D*) Computation of the free energy differences between positive and negative modes with importance sampling (IS) from the NF as a function of a local biasing field added in the Hamiltonian Eq. **19**. Results are reported for inverse temperature $\beta = 20$, as in the rest of the plots, and for the same experiment repeated at temperature $\beta = 10$. Errors bars computed from estimator variance are smaller than the marker.

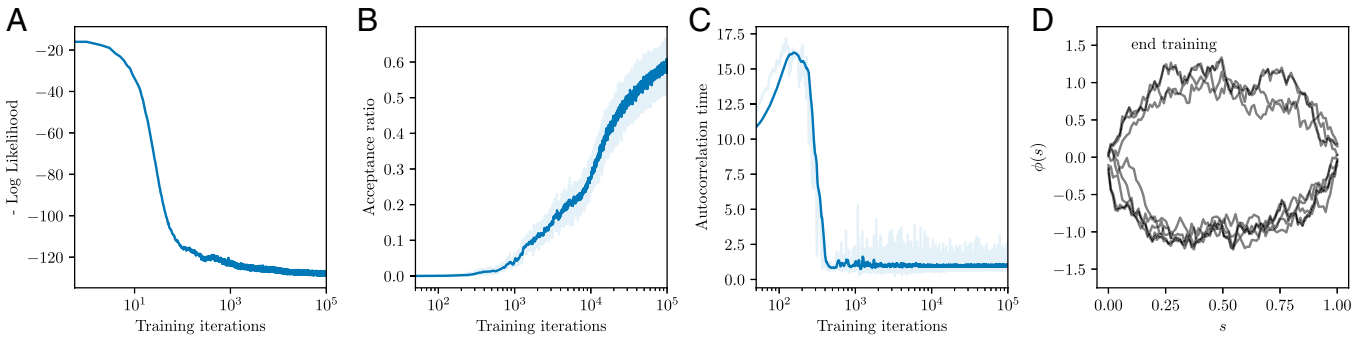

Fig. 2. Concurrent training and sampling of the stochastic Allen–Cahn model with a real-non volume preserving (real-NVP) NF. (*A*) The stochastic gradient descent using samples generated by the procedure decreases the negative log likelihood gradually. (*B*) As the training progresses, the acceptance rate in the Metropolis–Hasting using proposals from the NF improves gradually, reaching levels well beyond 50%. The rolling average over the last 50 time steps is plotted in darker color. (*C*) As independent proposals from the flow start getting accepted, the Markov Chain autocorrelation times drops abruptly. (*D*) Fast mixing is illustrated by looking at the consecutive states of one walker updated with the transition kernel combining local Langevin updates and resampling with the push forward. In 10 steps, the single walker has jumped between ϕ_+ and ϕ_-.

In particular, we study the situation where T learns the instantaneous distribution at all times: that is, $\hat{\rho} = \rho_t$ (and hence, $\hat{g}_t = g_t$) for all $t \geq 0$. While this is certainly a significant approximation, we observe in numerical experiments that there is a dramatic improvement in sampling once there is some mixing between metastable basins, which motivates this limiting scenario. In this case, under the assumptions that there exists some $t_0 \geq 0$ such that $D_{t_0} < \infty$ and

$$G_{t_0} = \inf_{x\in\Omega} \frac{\rho_{t_0}(x)}{\rho_*(x)} = \inf_{x\in\Omega} g_{t_0}(x) > 0, \qquad [14]$$

we show that

$$\forall t \geq t_0 : D_t \leq \frac{D_{t_0}}{(G_{t_0}(e^{\alpha(t-t_0)} - 1) + 1)^2}. \qquad [15]$$

This equation indicates that $D_t \leq D_{t_0}$ remains approximately constant for $\alpha(t - t_0) \leq \log G_{t_0}^{-1}$ and then, decays exponentially with constant rate $2\alpha > 0$ subsequently. The derivation of Eq. **15** also shows that the exponential rate is controlled by the resampling step of the MCMC algorithm that relies on the NF, and this rate can only improve when we concurrently use Langevin dynamics steps. In *SI Appendix*, section S4, we connect the sampling scheme we use to a birth–death Fokker–Planck equation (33), which could also be implemented in practice as a Markov jump process; again, this analysis emphasizes the favorable convergence properties of the scheme.

Scalability: Model-Informed Base Distributions and Maps, Mixtures, Etc. As the dimension of the problem grows, it becomes increasingly difficult to train a map to produce a push-forward distribution matching the target to a given level of accuracy. Before presenting numerical experiments, we emphasize a few additional ingredients, easing the learning of generative models for the sampling of complex high-dimensional systems.

When training an NF to represent a target density for which a preexisting empirical dataset is available, a standardizing transformation or a "whitening layer" is typically added at the output (21). This layer centers and rescales the different input dimensions such that their covariance matches the identity covariance of the standard normal distribution usually used as base distribution. This operation, although it requires preexisting data, crucially improves the outcome of learning when the original covariance of the data is highly anisotropic. In the experiments below, we show that it is sometimes possible to rely on the knowledge of the target distribution to perform an operation akin to this whitening layer with no preexisting data samples. For instance, below we choose a base Gaussian distribution with covariance matching the short-scale correlations of equilibrium configurations of the system's known Hamiltonian. We can also design physics-informed base distributions that are more adapted to the problem at hand than a Gaussian distribution; for example, in the interacting particle system, we used the uniform distribution of the particle in the domain, which is an ideal distribution in the gaseous phase.

Using prior knowledge about the physics can also help in designing the class of maps T to optimize upon. For example, in the interacting particle system, we used maps that factorize in ways tailored to the system's features. Yet another way of easing learning, especially when modes have very different fine structures or statistical weights, is to rely on a mixture of maps instead on a single map, training each component to represent a different mode. A similar idea was exploited in ref. 21 to compute free energy differences between basins after training. In practice, a map T_m is pretrained for each mode indexed by m using data generated with the local MCMC sampler initialized in the corresponding mode. Then, the adaptive MCMC procedure described in *Algorithm 1* can be started. The nonlocal proposal is the mixture of the push-forward $\hat{\rho}_m$ with initial weights p_m. The adaptive part of the proposal then amounts to optimizing the mixture weights p_m via Eq. **11**, in a similar fashion as the parameters of the flow when using a single map.* This mixture method requires that we train several maps but allows for treatment of more complex systems, as demonstrated below.

Numerical Experiments

Fast-Mixing Augmented MCMC for Random Fields. As a first example to illustrate the efficacy of adaptive sampling, we consider a stochastic Allen–Cahn model, a canonical and ubiquitous model for the microscopic physics of phase transitions in condensed matter systems (34).

Field system. The stochastic Allen–Cahn equation is defined in terms of a random field $\phi : [0, 1] \to \mathbb{R}$ that satisfies

$$\partial_t \phi = a\partial_s^2 \phi + a^{-1}(\phi - \phi^3) + \sqrt{2\beta^{-1}}\,\eta(t, s), \qquad [16]$$

where $a > 0$ is a parameter, β is the inverse temperature, $s \in [0, 1]$ denotes the spatial variable, and η is a spatiotemporal white noise, and we impose Dirichlet boundary conditions in which $\phi(s = 0) = \phi(s = 1) = 0$ throughout. This stochastic partial differential equation (SPDE) is well posed in one spatial dimension

*In principle, the parameters of each T_m could also be further refined in this stage, but we have not tested this scenario in experiments yet.

(35, 36), and its invariant measure is the Gibbs measure associated with the Hamiltonian

$$U_*[\phi] = \beta \int_0^1 \left[\frac{a}{2} (\partial_s \phi)^2 + \frac{1}{4a} \left(1 - \phi^2(s)\right)^2 \right] ds. \qquad [17]$$

The first term in the Hamiltonian [**17**] is a spatial coupling that penalizes changes in ϕ and hence, at low temperature, has the effect of aligning the field in positive or negative direction. As a result, the Hamiltonian [**17**] has two global minima, denoted by ϕ^+ and ϕ^-, in which the typical values of ϕ are ± 1 (Fig. 1*A*). Because there is a free energy barrier between ϕ^+ and ϕ^-, local updates via traditional MCMC based on, for example, using the stochastic Allen–Cahn Eq. **16** will not mix, even on very long timescales. Indeed, if we wanted to compute the free energy difference between these basins, we would need to construct a pathway through configuration space and use importance sampling techniques along the path (37). Our adaptive MCMC algorithm, augmented with an NF, offers an alternative approach. Fig. 2 demonstrates that a map T can be trained to efficiently generate samples with high statistical weight in the target distribution enabling rapid mixing across the free energy barrier.

Informed base measure. In order to learn the map robustly, a standard implementation of an NF model, with a standard Gaussian field with uncoupled spins as base measure, does not suffice in this instance. Using a base measure that is "informed" alleviates this issue. Explicitly, we sample the base measure corresponding to a Gaussian random field with a local coupling (a "Ornstein–Uhlenbeck bridge"), which corresponds to a system with Hamiltonian

$$U_{\mathrm{B}}[\phi] = \beta \int_0^1 \left[\frac{a}{2} (\partial_s \phi)^2 + \frac{1}{2a} \phi^2 \right] ds. \qquad [18]$$

Importantly, this measure does not have any metastability and remains easy to sample. As discussed in *SI Appendix*, section S2, at this continuous-field level, we must choose this measure to ensure that the push-forward distribution has a nonvanishing statistical weight in the target distribution.

Numerical implementation and results. In practice, we must discretize the field on a grid, and throughout, we take $N = 100$ with a lattice spacing $\Delta s = 1/N$, meaning that the map we must learn is high-dimensional $T : \mathbb{R}^N \to \mathbb{R}^N$. We also use the associated Langevin equation as the discretized version of the SPDE [**16**] to generate samples as the local component of our compounded MCMC scheme.

We trained maps T and T_{U} along our adaptive MCMC with the informed base measure [**18**] and an uninformed Gaussian measure that lacked a coupling term (*SI Appendix*, Eq. **S66**), respectively, using the same architecture and compared their suitability for resampling after an equal number of iterations. Typical configurations $\phi(x)$, in this case generated by the NF T, are shown in Fig. 1*A*. For comparison, we show in *SI Appendix*, Fig. S3 samples generated with T_U.

While T generates samples that are accepted in the MCMC procedure with average acceptance rate approaching 60% (Fig. 2), T_{U} fails to produce samples that have appreciable statistical weight in the target distribution. The evident difference is in the local structure of the random fields that are produced. Fig. 1*C* shows the Fourier spectrum of field ϕ computed with samples from the target measure (obtained using the proposed MCMC method after convergence) as well as from the push forward in the informed $\hat{\rho}$ and uninformed $\hat{\rho}_{\mathrm{U}}$ case. While $\hat{\rho}$ accurately captures the decay of the Fourier components at all scales, T_{U} fails to compensate for the uncoupled base measure, and $\hat{\rho}_{\mathrm{U}}$ does not accurately capture high-frequency oscillations of the field ϕ, which subsequently leads to high rejection rates in the MCMC procedure.

While at the discrete level, the adequacy of the base measure is a priori less stringent than at the continuous level examined in *SI Appendix*, section S2, this experiment shows that at $N = 100$, it is already highly beneficial to preadapt the covariance of the push forward. In the absence of preexisting samples to compute an empirical whitening transform, it is the role of the proposed informed base measure.

Interpreting the map. Examining the learned map T reveals its simple underlying structure. As shown in Fig. 1*B*, the map is spatially local, transporting spins near the center of the domain to ± 1, while spins near the boundary are mapped closer to 0. It is again useful to examine the properties of a mapped configuration in Fourier space. The $k = 0$ mode reveals that the mean value is transported substantially; $T(\hat{\phi}_0)$ is approximately ± 1, as shown in *SI Appendix*, Fig. S2. However, higher-frequency modes are left invariant by the map (*SI Appendix*, Fig. S2).

Calculating free energy differences. Perhaps most remarkably, the learned map T can be used to evaluate free energy differences between the metastable basins ϕ^- and ϕ^+, even in thermodynamic conditions distinct from those in which the map was trained. Fig. 1*D* shows an estimate of the free energy difference between the positive and negative metastable basins as a function of an external field h, which enters the Hamiltonian as

$$U_{*,h}[\phi] = \beta \int_0^1 \left[\frac{a}{2} (\partial_s \phi)^2 + \frac{1}{4a} (1 - \phi^2(s))^2 + h\phi(s) \right] ds. \qquad [19]$$

These estimates are produced with importance sampling using $\hat{\rho}$ as described in *SI Appendix*, section S11. Analytical estimates at low temperature via a Laplace approximation reveal that the NF accurately recapitulates the free energy difference despite the fact that the map was optimized only with samples where $h = 0$. Similar generalization properties were observed in refs. 21, 38, and 39, where a map was used at temperatures distinct from the temperature at which training data were collected. This approach is valid in cases where the modified parameter, here the field h, distorts the relative populations of the metastable basins but has a mild effect on the local structure of the field, which can be controlled by monitoring the variance of the estimator.

Additional tests for related applications are presented in *SI Appendix*. For this stochastic Allen–Cahn system, we show that the method can be useful to sample configurations with domain walls by tilting the Hamiltonian (*SI Appendix*, Fig. S5). In *SI Appendix*, section S9, we discuss a similar sampling problem that involves the nonequilibrium transition path, which we employ to illustrate the use of Brownian bridge base measures. This example is challenging as metastable basins have very different statistical weights, which is also the case for the particle systems discussed in the next section, where we demonstrate the usage of mixtures to tackle this circumstance.

Detecting Phase Transitions in Interacting Particle Systems. Thermal systems undergoing a first-order phase transition are archetypal examples of models displaying metastability. Near the transition point, ergodic mixing from the unstable to the stable phase is broken in the thermodynamic limit, leading to the well-known challenge of detecting these transitions with molecular dynamic simulations. In this section, we show our method to be useful in this context.

Particle system and phase diagram. As an example, we consider a system of N interacting particles evolving in a two-dimensional periodic box of lateral size L according to the Langevin equation (here written in the overdamped limit):

$$dx_i = -\frac{1}{N} \sum_{j=1}^{N} \nabla W(x_i - x_j)\, dt + \sqrt{2\beta^{-1}}\, dW_i. \qquad [20]$$

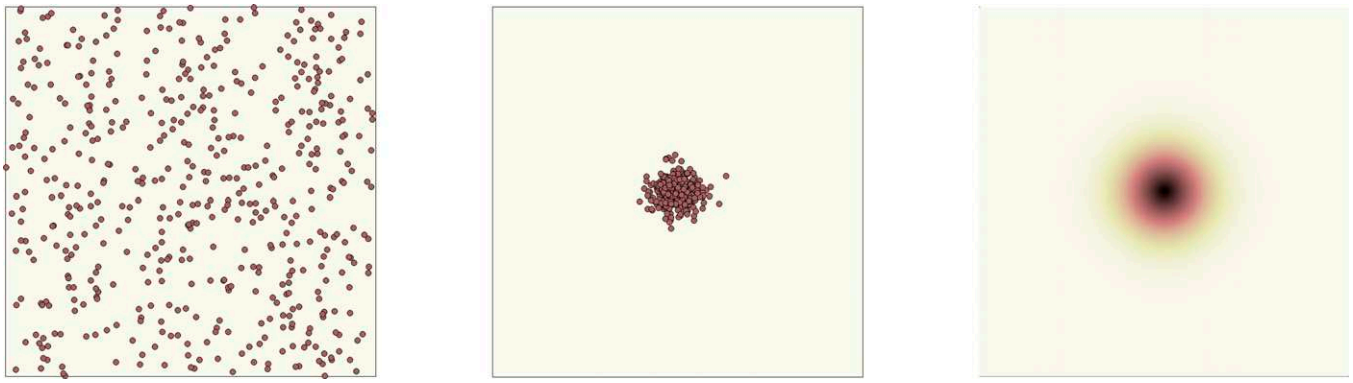

Fig. 3. Detecting phase transitions in interacting particle systems. (*Left* and *Center*) Two hundred particles seen in the gas and liquid phases, respectively, in dimension $d = 2$ at a temperature below the critical β_c^{-1}, at which both phases are metastable but the clustered one is thermodynamically preferred. (*Right*) A contour plot of the local density $u_d(x)$ of the particles in the liquid phase plotted in log scale.

The interaction $W(x)$ is a pairwise attracting potential with range $a > 0$:

$$W(x) = -\exp(L^2[1-\cos(2\pi|x|/L)^2]/(4\pi^2 a^2)), \qquad [21]$$

which when $a \ll L$, is well approximated by $W(x) = -\exp(-|x|^2/[2a^2])$. These equations sample the Boltzmann–Gibbs distribution of the system

$$\rho_*(X) = Z_*^{-1} \exp\left(-\frac{\beta}{2N}\sum_{i,j=1}^{N} W(x_i - x_j)\right), \qquad [22]$$

where we denote by $X = (x_1, \dots, x_N) \in [0, L]^{2N}$ the state of the N particle system.

When a is much smaller than L, in the thermodynamic limit ($N \gg 1$), this system displays a first-order phase transition between a gas-like phase, where the particles are uniformly distributed in the domain that is preferred at high temperatures, and a liquid-like phase, where they cluster in a droplet that is preferred at low temperatures. Typical particle configurations in these phases are shown in Fig. 3. The phase diagram of the model can be estimated using a mean-field approximation (*SI Appendix*, section S10 has details) and is shown in Fig. 4.

Detecting this phase transition via brute-force simulation of Eq. **20** is, however, challenging because the particles stay trapped in whichever configuration they occupy (homogeneous or droplet) for very long periods of time; in fact, the transition times from one phase to the other in a parameter regime where they are both metastable can be estimated as $t_{l\to g} \asymp \exp(N\beta F_{l\to g})$ and $t_{g\to l} \asymp \exp(N\beta F_{g\to l})$, where $F_{g\to l}$ and $F_{l\to g}$ denote, respectively, free energy barriers between the liquid and the gas phase and vice versa. Since these barriers are both independent of N, these transition times diverge exponentially with the number of particles N.

Adaptive simulations augmented by nonlocal resampling. Simulation of Eq. **20** augmented by a nonlocal resampling map can detect the phase transition. As the two modes of interest have here very different structures and very different statistical weights across the phase transition, we resort to a parameterization of the nonlocal proposal density $\hat{\rho}$ in terms of a mixture. For the homogeneous phase mixture component, it is straightforward to draw particles configurations; we can simply pick each of their individual positions uniformly in the box. For the droplet phase mixture component, it is natural to use as base distribution the uniform distribution $\rho_B(X) = 1/L^2$ that corresponds to the homogeneous phase and train a map T that then takes one such configuration and maps it onto a droplet configuration whose local density is close to that of the particles in the liquid phase. Denoting this local density by $u_d(x)$, we can also exploit the fact that the liquid droplet has no internal structure and factorize the map as $T(X) = (t(x_1), t(x_2), \dots, t(x_N))$ with $t : [0, L]^2 \to [0, L]^2$ such that $t(x)$ has density $u_d(x)$ if x is drawn uniformly in $[0, L]^2$: that is, $u_d(x) = L^{-2} \det \nabla \bar{t}(x)$, where $\bar{t}$ is the inverse of the map t. All in all, this leads to a resampling mixture density $\hat{\rho}$ that can be expressed as

$$\hat{\rho}(X) = p\prod_{i=1}^{N} u_d(x_i) + qL^{-2N}, \qquad [23]$$

where $p \in [0, 1]$ is a factor to be learned, $q = 1 - p$, and the local density $u_d(x)$ needs to be estimated—in the simulations, we simply used the mean-field approximation recalled in *SI Appendix*, section S10 to calculate $u_d(x)$ numerically, but this density could also be estimated directly from the molecular dynamics (MD) simulations.

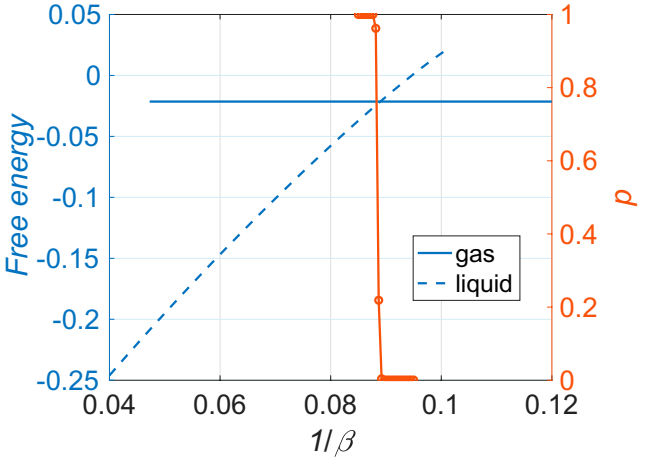

Fig. 4. Detecting phase transitions in interacting particle systems. Blue curves and labels show the free energies of the gas and liquid phases, showing that a first-order phase transition occurs at the critical $1/\beta_c \approx 0.089$. For temperatures around this value, particles configurations in either the homogeneous or the clustered phase are highly metastable, and no transition between these states is observed in brute-force simulation of Eq. **20**. Red curve and labels show the value of p in the mixture [**23**] learned by our adaptive procedure augmented with an NF starting from $p(0) = 1/2$; the algorithm correctly learns the right value of p in the mixture and thereby, is able to detect the phase transition.

Implementation and results. Consistent with *Algorithm 1*, we run Eq. **20** used as the local sampler for a fixed duration t_L and then attempt a resampling move by proposing a configuration from Eq. **23**. This resampling step requires one to evaluate the Metropolis–Hastings probability in Eq. **6** accurately, which is nontrivial when N is large. Here, we used as approximation $\log \hat{\rho}(X) \approx S(X)$ with

$$S(X) = \min\left(\sum_{i=1}^{N} \log u_d(x_i) + \log p, -2N \log L + \log q\right), \quad [24]$$

leading to the following explicit approximation for the ratio $\rho_*(X)/\hat{\rho}(X)$:

$$\frac{\rho_*(X)}{\hat{\rho}(X)} \approx \exp\left(-\frac{\beta}{2N}\sum_{i,j=1}^{N} W(x_i - x_j) - S(X)\right). \quad [25]$$

This expression shows that the presence $\hat{\rho}(X)$ in the Metropolis–Hastings probability effectively accounts for the entropy of the particle configurations, as opposed to their energy accounted for by $\rho_*(X)$. Eq. **25** also emphasizes the need for $\hat{\rho}$ (i.e., the NF map in general) to be accurate enough; indeed, to get any significant probability of acceptance, even for a move aimed toward the thermodynamically preferred phase, the factor in the exponentials in Eq. **25** must be of order 1 in N. If the map fails to achieve this accuracy and the factors remain of order N (which is their typical scale for a map that is unadapted), the move would be systematically rejected (or accepted between configurations with little resemblance to permitted ones). This issue will be generic for problems where the system's energy is extensive.

In the context of the present example, once $u_d(x)$ has been estimated, the learning component of *Algorithm 1* reduces to optimizing the parameter p. This is done by approximating the KL divergence $D_{\mathrm{KL}}(\hat{\rho}(X)||\rho_*(X))$ using an estimator based on using the current state $X(t)$ of the system. Specifically, we used $\log \hat{\rho}(X(t)) \approx S(X(t))$ as an objective on which we performed gradient descent in p concurrently with running the augmented MD strategy above.

We applied the procedure above to a system of $N = 200$ particles drawn initially from the mixture density [**23**] using $p(0) = 1/2$ as the initial value. As can be seen in Fig. 4, this allowed us to train p to values converging either to $p = 0$ or $p = 1$ in a way that detects the phase transition. That is, the augmented procedure correctly reweights the homogeneous and droplet configurations and determines which of the two is the most likely even in situations where brute-force MD simulations would observe no transitions between these configurations.

We stress that a simplifying feature of this example is that the particles experience no short-range repulsion (i.e., there is no order in the droplet phase). This is what allowed us to use the product of $u_d(x)$ in the mixture density in Eq. **23**. In systems with hard-core repulsions, this approximation is invalid, in which case more complicated mixtures (or equivalently, more complicated maps in the NF) will have to be used. We leave investigation of such situations for future work.

Conclusions

Connections and Differences with Previous Works. Most of the methods that seek to train an NF to (approximately) sample from the Boltzmann distribution of a known target energy rely on the reverse KL training using Eq. **9** (15, 18, 38, 39). However, this objective is known to be prone to "mode collapse," where the estimation concentrates on the bulk of one mode. This failure comes typically from the fact that the map may never explore modes far away from the core of the base distribution—such that these modes are missed altogether[†] (40). Additionally, the reverse KL objective is known to lead to underestimation of the tails (41).

To alleviate the shortcomings of reverse KL training, ref. 21 relied on initial short trajectories to estimate and factor in the optimization objective of the forward KL divergence. Closer to our proposition, the authors of ref. 42 propose Markov score climbing, an adaptive MCMC strategy using the same estimate of the forward KL as Eq. **11** to discover good variational approximators. Our method can be seen as an extension of Markov score climbing, introducing the additional alternation with a local sampler and the replacement of simple variational families by the more flexible NFs. Along the same lines, ref. 43 investigated proposals parametrized by lower triangular maps. Interestingly, ref. 21 also proposed an iteratively retrained variant of its Boltzmann generator that shares similarity with our proposition. Note that an advantage of the adaptive MCMC over the consecutive training, then sampling schemes with either reverse KL (15, 18, 38) or forward KL (11, 21) training objectives is to allow for real-time monitoring of the quality of training toward the final purpose of obtaining well-mixed samples. In the experiments, that was done by monitoring, for instance, acceptance rates and autocorrelation of the chains across iterations.

The adaptive MCMC proposed here retains a local component in the sampling in the form of interleaved steps of a local sampler that brings robustness to the scheme. Albeit different, the adaptive MCMC proposed in ref. 21 also defines an intermediary between a purely global and a purely local procedure. The proposals consist of local steps in the latent space of the NF, while our query of the generative model yields completely independent resampling. To encourage transition across modes, the algorithm of ref. 21 was augmented with parallel tempering in ref. 39, a step that does not appear to be necessary in our scheme.

Outlook and Future Work. As the use of data-driven methods from machine learning becomes increasingly routine in the physical sciences, we must carefully assess the cost of data acquisition and training to ensure that we can leverage machine learning methods in a productive fashion. Sampling systems with complex local structure and multiple metastable basins is a generically challenging task in high-dimensional systems, and we have already seen that neural networks can contend with this challenge in nontrivial settings (5, 6, 15, 18, 21). Nonlocal transport in MCMC algorithms can significantly enhance mixing, and NFs provide a compelling framework for designing adaptive schemes, even in cases where no statistically representative dataset is available at first. Nevertheless, we do not find that these methods enable discovery of unknown modes of a target distribution, emphasizing the importance of having some a priori information about the metastable states of the system.

Many questions remain about how to ensure efficient learning in complex high-dimensional systems and encourage desirable properties of the map, such as locality and transferability. Incorporating known invariances and symmetries of target distributions into architectures is currently a key area of research (e.g., refs. 44 and45) that will help scaling further applications of sampling methods enhanced by learning.

Data Availability. Python code and trained models have been deposited in Zenodo (https://zenodo.org/record/4783701#.Yfv53urMJD8).

ACKNOWLEDGMENTS. G.M.R. acknowledges support from the Terman Faculty Fellowship. E.V.-E. acknowledges partial support from NSF Materials Research Science and Engineering Center Program Grant DMR-1420073, NSF Grant DMS-1522767, and a Vannevar Bush Faculty Fellowship.

[†] Note that annealing of the target can help to catch multiple modes in some simple cases but offers no guarantees (18).

1. P. G. Bolhuis, D. Chandler, C. Dellago, P. L. Geissler, Transition path sampling: Throwing ropes over rough mountain passes, in the dark. *Annu. Rev. Phys. Chem.* **53**, 291–318 (2002).
2. I. Goodfellow *et al.*, "Generative adversarial nets" in *Proceedings of the 27th International Conference on Neural Information Processing Systems* (MIT Press, Cambridge, MA, 2014), pp. 2672–2680.
3. D. P. Kingma, M. Welling, "Auto-encoding variational Bayes" in *2nd International Conference on Learning Representations* (ICLR, 2014).
4. J. Song, S. Zhao, S. Ermon, "A-NICE-MC: Adversarial training for MCMC" in *Advances in Neural Information Processing Systems*, I. Guyon *et al.*, Eds. (Curran Associates, Inc., Red Hook, NY, 2017), vol. 30.
5. Y. Song *et al.*, "Score-based generative modeling through stochastic differential equations" in *International Conference on Learning Representations* (ICLR, 2021).
6. L. Huang, L. Wang, Accelerated Monte Carlo simulations with restricted Boltzmann machines. *Phys. Rev. B* **95**, 035105 (2017).
7. D. Sejdinovic, H. Strathmann, M. L. Garcia, C. Andrieu, A. Gretton, "Kernel adaptive Metropolis-Hastings" in *Proceedings of the 31st International Conference on Machine Learning* (PMLR, 2014), vol. 32, pp. 1665–1673.
8. D. Levy, M. D. Hoffman, J. Sohl-Dickstein, "Generalizing Hamiltonian Monte Carlo with neural networks" in *International Conference on Learning Representations* (ICLR, 2018).
9. M. K. Titsias, Learning model reparametrizations: Implicit variational inference by fitting MCMC distributions. arXiv [Preprint] (2017). https://arxiv.org/abs/1708.01529 (Accessed 1 January 2021).
10. T. A. Le, M. Igl, T. Rainforth, T. Jin, F. Wood, "Auto-encoding sequential Monte Carlo" in *International Conference on Learning Representations* (ICLR, 2018).
11. B. McNaughton, M. V. Milošević, A. Perali, S. Pilati, Boosting Monte Carlo simulations of spin glasses using autoregressive neural networks. *Phys. Rev. E* **101**, 053312 (2020).
12. M. Germain, K. Gregor, I. Murray, H. Larochelle, "MADE: Masked autoencoder for distribution estimation" in *Proceedings of the 32nd International Conference on Machine Learning* (PMLR, 2015), vol. 37, pp. 881–889.
13. D. Rezende, S. Mohamed, "Variational inference with normalizing flows" in *Proceedings of the 32nd International Conference on Machine Learning* (PMLR, 2015), vol. 37, pp. 1530–1538.
14. G. Papamakarios, E. Nalisnick, D. J. Rezende, S. Mohamed, B. Lakshminarayanan, Normalizing flows for probabilistic modeling and inference. *J. Mach. Learn. Res.* **22**, 1–64 (2021).
15. M. S. Albergo, G. Kanwar, P. E. Shanahan, Flow-based generative models for Markov chain Monte Carlo in lattice field theory. *Phys. Rev. D* **100**, 034515 (2019).
16. H. Sidky, W. Chen, A. L. Ferguson, Molecular latent space simulators. *Chem. Sci. (Camb.)* **11**, 9459–9467 (2020).
17. L. Sbailò, M. Dibak, F. Noé, Neural mode jump Monte Carlo. *J. Chem. Phys.* **154**, 074101 (2021).
18. D. Wu, L. Wang, P. Zhang, Solving statistical mechanics using variational autoregressive networks. *Phys. Rev. Lett.* **122**, 080602 (2019).
19. K. A. Nicoli *et al.*, Estimation of thermodynamic observables in lattice field theories with deep generative models. *Phys. Rev. Lett.* **126**, 032001 (2021).
20. L. Del Debbio, J. M. Rossney, M. Wilson, Efficient modelling of trivializing maps for lattice ϕ^4 theory using normalizing flows: A first look at scalability. *Phys. Rev. D* **104**, 094507 (2021).
21. F. Noé, S. Olsson, J. Köhler, H. Wu, Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning. *Science* **365**, eaaw1147 (2019).
22. C. Andrieu, A. Jasra, A. Doucet, P. D. Moral, On nonlinear Markov chain Monte Carlo. *Bernoulli* **17**, 987–1014 (2011).
23. S. P. Meyn, R. L. Tweedie, *Markov Chains and Stochastic Stability* (Springer Science & Business Media, 2012).
24. G. O. Roberts, R. L. Tweedie, Exponential convergence of Langevin distributions and their discrete approximations. *Bernoulli* **2**, 341–363 (1996).
25. C. Andrieu, É. Moulines, On the ergodicity properties of some adaptive MCMC algorithms. *Ann. Appl. Probab.* **16**, 1462–1505 (2006).
26. H. Haario, E. Saksman, J. Tamminen, An adaptive Metropolis algorithm. *Bernoulli* **7**, 223–242 (2001).
27. A. Jasra, D. A. Stephens, C. C. Holmes, On population-based simulation for static inference. *Stat. Comput.* **17**, 263–279 (2007).
28. C. Villani, *Topics in Optimal Transportation* (Graduate Studies in Mathematics, American Mathematical Society, Providence, RI, 2003), vol. 58.
29. F. Santambrogio, *Optimal Transport for Applied Mathematicians* (Progress in Nonlinear Differential Equations and Their Applications, Springer International Publishing, Cham, Switzerland, 2015), vol. 87.
30. L. Dinh, D. Krueger, Y. Bengio, "NICE: Non-linear independent components estimation" in *3rd International Conference on Learning Representations, ICLR 2015*. http://arxiv.org/abs/1410.8516. Accessed 23 February 2022.
31. L. Dinh, J. Sohl-Dickstein, S. Bengio, "Density estimation using real NVP" in *International Conference on Learning Representations* (ICLR, 2017), p. 32.
32. D. W. Stroock, "Logarithmic Sobolev inequalities for Gibbs states" in *Dirichlet Forms: Lectures Given at the 1st Session of the Centro Internazionale Matematico Estivo (C.I.M.E.)*, E. Fabes *et al.*, Eds. (Lecture Notes in Mathematics, Springer, Berlin, Germany, 1993), pp. 194–228.
33. Y. Lu, J. Lu, J. Nolen, Accelerating Langevin sampling with birth-death. arXiv [Preprint] (2019). https://arxiv.org/abs/1905.09863 (Accessed 8 February 2021).
34. N. Berglund, G. D. Gesù, H. Weber, An Eyring–Kramers law for the stochastic Allen–Cahn equation in dimension two. *Electron. J. Probab.* **22**, 1–27 (2017).
35. W. G. Faris, G. Jona-Lasinio, Large fluctuations for a nonlinear heat equation with noise. *J. Phys. Math. Gen.* **15**, 3025–3055 (1982).
36. R. Marcus, Parabolic Ito equations. *Trans. Am. Math. Soc.* **198**, 177–190 (1974).
37. D. Frenkel, B. Smit, *Understanding Molecular Simulation: From Algorithms to Applications* (Elsevier, 2001).
38. K. A. Nicoli *et al.*, Asymptotically unbiased estimation of physical observables with neural samplers. *Phys. Rev. E* **101**, 023304 (2020).
39. M. Dibak, L. Klein, F. Noé, "Temperature-steerable flows" in *Third Workshop on Machine Learning and the Physical Sciences (NeurIPS 2020)* (2020). https://ml4physicalsciences.github.io/2020/files/NeurIPS_ML4PS_2020_67.pdf. Accessed 23 February 2022.
40. G. S. Hartnett, M. Mohseni, Self-supervised learning of generative spin-glasses with normalizing flows. arXiv [Preprint] (2020). https://arxiv.org/abs/2001.00585 (Accessed 22 April 2020).
41. Y. Yao, A. Vehtari, D. Simpson, A. Gelman, "Yes, but did it work? Evaluating variational inference" in *Proceedings of the 35th International Conference on Machine Learning* (PMLR, 2018), vol. 80, pp. 5581–5590.
42. C. A. Naesseth, F. Lindsten, D. Blei, "Markovian score climbing: Variational inference with KL(p||q)" in *Advances in Neural Information Processing Systems 33*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, H. Lin, Eds. (Curran Associates, Red Hook, NY, 2020).
43. M. D. Parno, Y. M. Marzouk, Transport map accelerated Markov chain Monte Carlo. *SIAM-ASA J. Uncertain* **6**, 645–682 (2018).
44. J. Köhler, L. Klein, F. Noé, "Equivariant flows: Exact likelihood generative learning for symmetric densities" in *Proceedings of the 37th International Conference on Machine Learning*, H. Daumé III, A. Singh, Eds. (PMLR, 2020), vol. 119, pp. 5361–5370.
45. D. J. Rezende *et al.*, "Normalizing flows on tori and spheres" in *Proceedings of the 37th International Conference on Machine Learning* , H. Daumé III, A. Singh, Eds. (PMLR, 2020), vol. 119, pp. 8039–8048.