Micro-Macro Consistency in Multiscale Modeling: Score-Based Model Assisted Sampling of Fast/Slow Dynamical Systems

E. R. Crabtree, ¹ J. M. Bello-Rivas, ¹ and I. G. Kevrekidis ¹

Department of Chemical and Biomolecular Engineering, Johns Hopkins University,

Baltimore, MD 21218 USA

(Dated: 29 December 2023)

A valuable step in the modeling of multiscale dynamical systems in fields such as computational chemistry, biology, materials science and more, is the representative sampling of the phase space over long timescales of interest; this task is not, however, without challenges. For example, the long term behavior of a system with many degrees of freedom often cannot be efficiently computationally explored by direct dynamical simulation; such systems can often become trapped in local free energy minima. In the study of physics-based multitime-scale dynamical systems, techniques have been developed for enhancing sampling in order to accelerate exploration beyond free energy barriers. On the other hand, in the field of Machine Learning, a generic goal of generative models is to sample from a target density, after training on empirical samples from this density. Score based generative models (SGMs) have demonstrated state-of-the-art capabilities in generating plausible data from target training distributions. Conditional implementations of such generative models have been shown to exhibit significant parallels with long-established –and physics based– solutions to enhanced sampling. These physics-based methods can then be enhanced through coupling with the ML generative models, complementing the strengths and mitigating the weaknesses of each technique. In this work, we show that that SGMs can be used in such a coupling framework to improve sampling in multiscale dynamical systems.

We discuss the parallels between generative models in machine learning, particularly score-based generative models (SGMs), and traditional physics-based enhanced sampling methods like Umbrella Sampling (US). We show that SGMs can be coupled with traditional enhanced sampling to mitigate drawbacks that each tool has when used alone. We apply this coupling method to multi-time-scale stochastic dynamical systems and demonstrate that the coupling can offer faster convergence to the desired system conditional distribution.

I. INTRODUCTION

In the realm of computational chemistry and physics, numerous enhanced sampling tools have been developed for the purpose of exploring the phase space of multiscale systems of interest¹. Often, the goal in the implementation of these tools is to understand system behavior over long time scales; this is challenging due to the many degrees of freedom in typical models and simulations that are often performed at the fine scale. Many of these enhanced sampling tools are aimed at coarse-grained, low-dimensional models, in terms of coarse collective variables, that describe the effective, macroscopic behavior of the system. When these reduced descriptors are not a priori known, they must be identified, either analytically or in a data-driven way, before enhanced sampling can proceed. The construction of reduced, coarse-scale models from fine scale ones, in terms of such "Collective Variables" (CVs), has been a prevalent question in multiscale modeling for decades, and many data-driven techniques for identifying low dimensional descriptors have been developed over the years^{2–6}. It follows that, once a reduced order representation of a system is constructed, there also arises a need for "returning" selected reduced representation instances back to corresponding instances in the full, ambient space of the system. The task is not trivial, but methods to practically accomplish this inverse mapping, sometimes called "lifting"^{7,8} back to the ambient space, have been developed and commonly used in many fields involving first principles modeling. For example, umbrella sampling (US) in molecular dynamics⁹ involves adding a harmonic biasing potential to a CV in order to bias a fine-scale molecular simulation toward prescribed coarse-scale variable values of interest. This produces fine-scale data conditioned on a coarse-scale representation, and thus accomplishes the aforementioned task of "lifting."

Meanwhile, in the field of Machine Learning, generative models have become popular for their ability to generate high dimensional data from latent distributions^{10–12}. In recent years,

generative adversarial networks (GANs) have been used to generate realistic data (such as images of a human face) from partial descriptors (e.g. eyes, noses); furthermore, both GANs and other neural network based generative models have been used for the purpose of "lifting" in the context of returning lower dimensional, coarse-grained descriptions back to the full ambient space of a multiscale dynamical system –whether through the interpolation of molecular conformations, or through generation of high dimensional data from identified CV representations^{13,14}. Presently, score based generative models (SGMs) constitute state-of-the-art models that have demonstrable capabilities in generating plausible data samples from a target training distribution in complex problems and systems^{15,16}.

Established enhanced sampling methods, whose outputs are fine-scale representations consistent with (conditioned on) coarse-scale representations, present parallels with new and emerging ML generative models. Both categories of techniques accomplish similar tasks, but exhibit contrasting drawbacks: physics based sampling methods (that do not require training data) often suffer from slow convergence due to metastability; generative models must be trained offline on huge datasets and, by virtue of being data-driven, they will not explicitly account for all physical constraints of a system. We will demonstrate that the two types of methods, arising from differing origins, are capable of complementing each other, potentially alleviating their respective drawbacks, for more efficient sampling when used in tandem. Previous work demonstrated that *conditional GANs* can be used efficiently to create new microscopic initial conditions for physics-based enhanced sampling tools; conversely, the enhanced ML sampling can create new training data for the GANs on-the-fly¹⁷. The present work shows that this coupling framework can be extended to new state-of-the-art generative models, SGMs.

The paper is organized as follows: Section II presents an overview of SGMs as a generative model. Section III proposes a framework in algorithmic form for coupling SGMs with physics-based enhanced sampling methods. Section IV demonstrates the capabilites of SGMs in generating conditional distributions from training data produced by multiscale stochastic dynamical systems models. Section V contains illustrative results from applying the coupling framework to different examples. Section VI critically summarizes the results and outlines possible future work.

II. OVERVIEW OF SGMS

A. Denoising SDEs

SGMs are trained to generate new data points of a target density by employing a discretized stochastic differential equation (SDE): this SDE takes the form of a reverse, or backward, SDE of a forward SDE of choice. The forward SDE is any SDE that would gradually transform training data into normally distributed noise, while the reverse SDE gradually "denoises" a normally distributed variable back to plausible data. These SDEs can take many possible forms but, in generic form, can be defined as

$$dx = f(x,t)dt + g(t)dB_t, (1)$$

and the corresponding reverse SDE is defined as

$$dx = [f(x,t) - g(t)^2 \nabla_x \log p_t(x)] dt + g(t) dB_t.$$
(2)

A feature of these two SDEs is that, if the initial condition in the reverse SDE is chosen to be x(T) of the forward SDE (where T is some specified time) then the reverse SDE can be integrated to eventually obtain x(0). Therefore, if we are able to run the reverse SDE initialized from the distribution x(T), then we can generate samples from x(0). We do not know x(0) a priori; however, we do have access to x(T) through the feature of choosing or parameterizing the SDE, so that its marginals with respect to time (here these marginals of x with respect to time, t, are written as $p_t(x)$, and when this marginal refers to a specific time, for example t = 0, this marginal can be written as p(x(0))) converge to a normal distribution at an exponential speed, independently of the original distribution of x(0). Therefore, we can start the reverse SDE in $x(T) = \mathcal{N}(0,I)$. The last unknown of this problem is then x(t) and therefore the drift $\nabla \log p_t(x)$ (where again $p_t(x)$ is the marginal distribution of x at time t) of the reverse SDE, which we can then approach by training a neural network (whose inputs are the samples x and respective times t, with learnable parameters denoted by θ) to approximate the unknown $s_{\theta}(x,t) \approx \nabla \log p_t(x)$.

B. Neural Network Drift Approximation

The reverse SDE is implemented using the Euler-Maruyama method¹⁸. To advance the SDE by $\delta t = t_{i+1} - t_i$, we use the following discretized equation:

$$x(t_{i+1}) = x(t_i) + (t_{i+1} - t_i) \left(f(x, t_i) - g(t)^2 \nabla_x \log p_{t_i}(x) \right) + \sqrt{(t_{i+1} - t_i)} g(t) B_{t_{i+1} - t_i}, \tag{3}$$

where $B_{t_{i+1}-t_i}$ is a normal random variable with standard variance. Now that we have a discretized reverse SDE, we can use a neural network to approximate $\nabla_x \log \hat{p}_t(x)$ (where $\hat{p}_t(x)$ refers to an approximation of the distribution $p_t(x)$). We do this through Score-Matching^{15,19,20}, resulting in the following loss for the neural network approximation of the reverse SDE drift:

$$L(\theta) = \mathbb{E}_{t \sim U(0,T)} \mathbb{E}_{x(0) \sim p(x(0))} \mathbb{E}_{x(t) \sim p(x(t)|x(0))} [\lambda(t) \| \nabla \log \hat{p}_t(x) - s_{\theta}(x(t),t) \|_2^2],$$

where U(0,T) is a uniform distribution over the time interval of [0,T], $\lambda(t)$ is a positive weighting function, and x(0) denotes x(t) at time t=0. In practice this Score-Matching loss is comparable to a more computationally efficient *denoising score matching objective*²⁰, which can be used as a substitute loss function.

C. Conditional Diffusion Models

We can also create a conditional diffusion model by having the neural network approximate $\nabla \log \hat{p}_t(x|y)$ where y is some given label, using the following score-matching loss function, also called the conditional denoising estimator (CDE), used and discussed in previous works^{21–23}:

$$L(\theta) = \mathbb{E}_{t \sim U(0,T)} \mathbb{E}_{x(0),y \sim p(x(0)|y)} \mathbb{E}_{x(t) \sim p(x(t)|x(0))} [\lambda(t) \| \nabla \log \hat{p}_t(x|y) - s_{\theta}(x(t),t,y) \|_2^2],$$

meaning that the new reverse SDE that we will evaluate becomes:

$$x(t_{i+1}) = x(t_i) + (t_{i+1} - t_i) \left(f(x,t) - g(t)^2 \nabla_x \log p_t(x|y) \right) + \sqrt{t_{i+1} - t_i} g(t) B_{t_{i+1} - t_i}.$$
 (4)

This is essentially the same SDE as equation 3 (the unconditional reverse SDE), but with a change to the Neural Net learned drift term. For conditional SGMs, Song et al.¹⁵ originally proposed a SDE with a score matching term of

$$\nabla_x \log p_t(x|y) = \nabla_x \log p_t(x) + \nabla_x \log p_t(y|x). \tag{5}$$

Score-Based Model Assisted Sampling

While this term may be convenient for situations in which an unconditional SGM has already been trained on the system of interest (or if the heuristics and domain knowledge needed to construct the closed form of $\log p_t(y|x)$ are readily available), we are only interested in training conditional SGMs (cSGMs) in this work. Therefore, we choose to use a reverse SDE with a score matching term that only requires training to approximate $\nabla_x \log p_t(x|y)$, and does not require prior training of an unconditional model in addition to an approximation of $\nabla_x \log p_t(y|x)$. Note that in this work, we are conditioning on continuous variables, rather than discrete labels. Diffusion models in literature have typically used discrete labels for class conditional generation¹⁵, while continuous labels have been used for regression based generation purposes such as inpainting $^{15,21-23}$.

III. PROPOSED FRAMEWORK

Algorithms 1 and 2 lay out the general framework of our proposed method for coupling SGMs with physics-based sampling methods.

Algorithm 1 Training a Conditional SGM on a Dynamical System

Input: Samples of data from a dynamical system, x(t), with a corresponding distribution P(x) and corresponding labels, y, if known $a \ priori$

- 1: Sample from P(x);
- 2: Identify a slow direction of the system, *y*, through *a priori* knowledge or using nonlinear dimensionality reduction;
- 3: Train a cSGM on data sampled from P(x) with labels, y, that correspond to the slow direction of the system;
- 4: Use the trained cSGM to sample from the approximated conditional measures of the system, $\hat{P}(x|y)$
- 5: **return** $\hat{P}(x|y)$ at prescribed y values.

Algorithm 2 Sampling Initialized by Conditional SGMs

Input: N samples produced by a SGM from an estimated distribution, $\hat{P}(x|y)$ at prescribed y values

- 1: **for** n = 1 **to** N **do**
- 2: Use physics-based biasing methods such as Umbrella Sampling to sample from P(x|y) initialized by an initial condition generated from $\hat{P}(x|y)$ –for new values of y–;
- 3: end for
- 4: Reweigh the biased samples if necessary
- 5: **return** P(x|y) at interpolated and extrapolated y values.

While training the cSGM has an up-front computational cost, once trained it can provide samples from a reasonable estimation of the true conditional distribution at the prescribed values set by the user. These estimated samples can then be used to initialize methods like US and help them converge to the true conditional distribution of the system faster than the original method, without the "informed" initialization. Section V further elaborates with numerical examples and presents evidence that this proposed approach can be faster and more computationally efficient than biased sampling of the dynamical system alone (provided that the conditional approximations generated by the SGM are sufficient approximations of the true conditionals). Furthermore, all SDE parameters for our SGM models were chosen using guidance from Karras et. al.²⁴. Network hyperparameters were also considered. The examples investigated in this work are fundamentally different from the image examples used by Karras, so our network is smaller and simpler than their networks. For all 2D dynamical system examples, we used a dense network of 8 layers of 64 to 512 nodes. Larger networks were also investigated but only provided marginal improvements.

IV. USING SGMS TO APPROXIMATE CONDITIONAL EQUILIBRIUM MEASURES

In the field of molecular simulation, many techniques have been created and used to surmount free energy barriers that impede the exploration of a molecule's phase space; full, efficient exploration of these phase spaces is important to fields such as drug design, materials discovery, and more. A popular example of this in computational chemistry is the aforementioned US method⁹. In US, an artificial bias is added to the system to force the dynamics toward a prescribed value of a variable that governs the bulk behavior of the system, often called a collective variable (CV).

Methods using machine learning-based generative models to produce high-dimensional molecular configurations consistent with given values for a system's CVs have been developed, and we have shown in previous work that coupling of the physics-based simulations with machine learning models is possible, using Generative Adversarial Networks as the ML model¹⁷. In this work, we propose a framework for coupling SGMs with physics-based simulations, initializing the physics with SGM generated data consistent with a desired CV.

However, before discussing the proposed method of coupling SGMs with physics based sampling, we first must discuss how well SGMs by themselves approximate conditional measures of multi-time-scale dynamical systems. Consider a dynamical system described by a two-dimensional set of multiscale SDEs:

$$dz_1(t) = a_1 dt + a_2 dB_1$$

$$dz_2(t) = -(-1 + 0.2z_1(t) + 4z_2(t)(-1 + z_2(t)^2))dt + a_3 dB_2(t)$$
(6)

where B_1 and B_2 are standard normal random variables, $a_1 = a_2 = 10^{-4}$, and $a_3 = \varepsilon a_1 = \varepsilon a_2 = 10^{-1}$ where the ratio of time scales is $\varepsilon = 10^3$; this implies that z_1 is the slow variable and z_2 is the fast variable (z_2 is faster than z_1 by a factor of z_2 is defined by a double-well potential where two basins of attraction at $z_2 = 1$ and $z_2 = -1$ are separated by a tall barrier. Furthermore, the wells are slowly evolving in relative depth as the values of the slow direction, z_1 , evolve in time. At $z_1 = 0$, the well at $z_2 = 1$ is relatively much deeper than the other, and as z_1 increases with time, the well at $z_2 = 1$ becomes progressively shallower, while the well at $z_2 = -1$ becomes deeper. The wells have the same depth when $z_1 = 5$, and the evolving well behavior is symmetric around $z_1 = 5$. Analogous to US in molecular dynamics, performed by adding a harmonic bias to a collective variable, we can sample conditional distributions of the z_2 variable by adding a harmonic biasing potential to the z_1 equation of our system, resulting in the following new equations:

$$dz_1(t) = (a_1 - \kappa(z_1(t) - z_1^0))dt + a_2dB_1$$

$$dz_2(t) = -(-1 + 0.2z_1(t) + 4z_2(t)(-1 + z_2(t)^2))dt + a_3dB_2(t)$$
(7)

where κ is a force constant dictating the strength of the bias, and z_1^0 is the value of z_1 toward which we wants to bias the simulation. We trained a cSGM on a long unbiased simulation of this system where the samples are labeled by their z_1 values (in this case, the dataset consisted of 100,000 samples from a simulation of these SDEs with $dt = 10^{-2}$). We then compared an output of the

trained SGM with a US output conditioned at the same slow variable (z_1) value, as well as the true probability density function (PDF) given by the equations of the system.

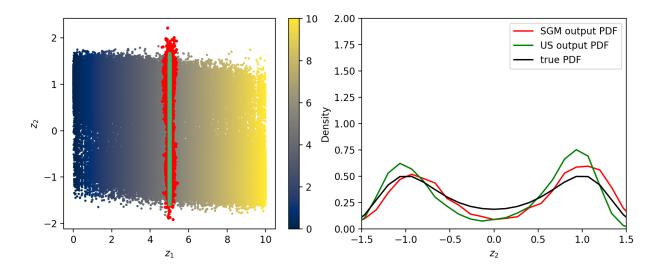


FIG. 1. On the left, output from a trained SGM conditioned at $z_1 = 5$ (red) is compared with the output from a US simulation (green) conditioned on the same z_1 value on top of a background of the training data for the SGM. On the right, the PDF estimated from the SGM output is compared to the PDF estimated from the US simulation, as well as with the true PDF, given by the known z_2 equation of the system (black line).

Figure 1 shows that, once trained, SGMs can provide data that is a valid approximation of a prescribed conditional distribution in a dynamical system. Additionally, cSGMs are capable of extrapolating beyond their training data. Figure 2 demonstrates that SGMs can step out from their training data and *provide a valid extrapolated conditional distribution* of data. Figure 3 shows that the extrapolation produces values consistent with the physics and the free energy surface of the system. This is useful for dynamical systems with trajectories that are computationally expensive to generate, or systems with large free energy barriers between relative minima. However, the outputs of the cSGM are ultimately approximations, and do not truly sample the physics of the system in question. Thus, the need for the framework proposed in Section III arises.

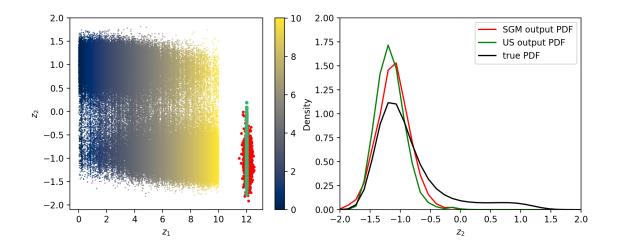


FIG. 2. On the left: output from a trained SGM, conditioned at $z_1 = 12$ (red) is compared with the output from a US simulation (green), conditioned at the same z_1 value plotted with the training data for the SGM. On the right, the PDF estimated from the SGM output is compared to the PDF estimated from the US simulation, as well as with the true PDF given by the known z_2 equation of the system (black line).

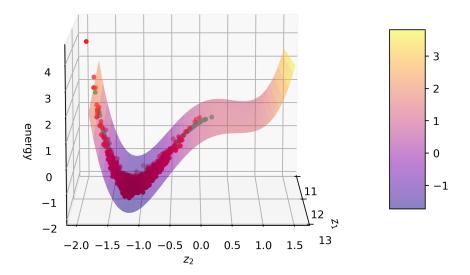


FIG. 3. The output from a trained SGM conditioned at $z_1 = 12$ (red) and a US simulation output (green) shown in Figure 2 are plotted against the potential energy surface of the system for a range around the conditioned z_1 value. Both the vertical axis and color of the surface describe the potential energy of the system. Note that there is now one local minimum at $z_2 = -1$ and the former local minimum at $z_2 = 1$ has flattened. Thus, both the SGM and US simulation mostly produce values around the single local minimum.

V. SGMS FOR INITIAL CONDITION GENERATION

A. SGM Assisted Physics-based Sampling

Consider a dynamical system described by the following two-dimensional set of multiscale SDEs:

$$dx_{1}(t) = a_{1}dt + a_{2}dB_{1}$$

$$dx_{2}(t) = -((1+x_{2}(t))^{2}(2(1+h-k)x_{2}(t) - 2h + 3(0.75k - 2)x_{2}(t)^{2} + 4x_{2}(t)^{3}) + 2(1+x_{2}(t))(h-2hx_{2}(t) + (1+h-k)x_{2}(t)^{2} + (0.75k - 2)x_{2}(t)^{3} + x_{2}(t)^{4})dt + a_{3}dB_{2}(t)$$
(8)

where B_1 and B_2 are standard normal random variables, $a_1 = a_2 = 10^{-4}$, and $a_3 = \varepsilon a_1 = \varepsilon a_2 = 10^{-1}$ where the ratio of time scales is $\varepsilon = 10^3$. Similar to the system detailed in Section IV, x_1 is the slow variable and x_2 is the fast variable (x_2 is faster than x_1 by a factor of 10^3). The behavior of x_2 is again defined as a double-well potential where two basins of attraction at $x_2 = 1$ and $x_2 = -1$ are separated by a free energy barrier. However, this system differs from the system in Section IV in that the wells in x_2 are not evolving with respect to x_1 and, in addition, in the inclusion of the variable h (that defines the height of the barrier) and the variable k (which describes the depth of the well at $x_2 = 1$). While the two directions, x_1 and x_2 are now independent, this new system allows us to demonstrate cases which the barrier between the two free energy basins is very large and practically insurmountable without enhanced sampling or high temperature. For the figures and examples shown, h = 8 and k = 0. We can also add a harmonic biasing potential to the x_1 equation of our system for sampling x_2 distributions with US, resulting in the following new equations:

$$dx_{1}(t) = (a_{1} - \kappa(x_{1}(t) - x_{1}^{0}))dt + a_{2}dB_{1}$$

$$dx_{2}(t) = -((1 + x_{2}(t))^{2}(2(1 + h - k)x_{2}(t) - 2h + 3(0.75k - 2)x_{2}(t)^{2} + 4x_{2}(t)^{3}) + 2(1 + x_{2}(t))(h - 2hx_{2}(t) + (1 + h - k)x_{2}(t)^{2} + (0.75k - 2)x_{2}(t)^{3} + x_{2}(t)^{4}))dt + a_{3}dB_{2}(t)$$
(9)

where κ is a force constant dictating the strength of the bias, and x_1^0 is the value of x_1 to bias toward. In order to produce initial conditions for the US simulations, we train an SGM on a long unbiased simulation of this system where the samples are labeled by their x_1 values (in this case, the dataset consisted of 100,000 samples from a simulation of these SDEs with $dt = 10^{-2}$). Once

trained, our SGM can generate point distributions in space consistent with a prescribed x_1 value (as demonstrated in Section IV). We then use these generated points to initialize ten US simulations parameterized by (9). The ten US simulations were run with $dt = 10^{-2}$ for 1000 time steps each.

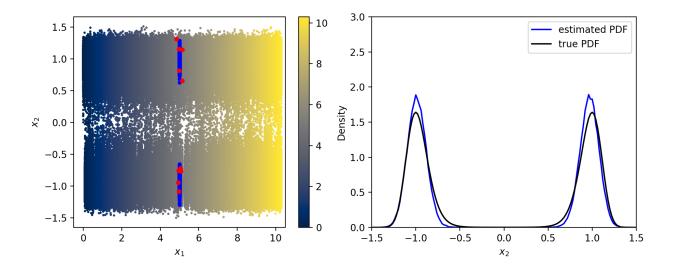


FIG. 4. On the left, outputs of 10 US simulations of 1,000 time steps (blue) conditioned at $x_1 = 5$ initialized by SGM generated points (red) on top of a background of the training data for the SGM. On the right, the probability density function estimated from pooling the histograms of the coupled US+SGM simulations (blue line) compared to the true probability density function given by the known x_2 equation of the system (black line).

These US simulations produce conditional measures of x_2 values constrained on x_1 values that are close to the respective biased value, x_1^0 . If the true distribution of the unbiased variable (x_2 in this case) is of interest, each US run may not fully span the true conditional distribution, so it is necessary to combine the histograms of each US run in order to produce the true conditional distributions of the system, which is accomplished through pooling the histograms. If biases are introduced to the x_2 direction, reweighting techniques such as the weighted histogram analysis method (WHAM)²⁵ (or multistate Bennett acceptance ratio (MBAR)²⁶) can be used to combine the biased histograms. These biased histograms are then utilized through WHAM to approximate the free energy surface (and thus, the unbiased joint distribution of x_2).

In addition to the multiple runs being able to be run *in parallel*, we can also perform convergence analysis with respect to the number of integration steps of the discretized system of SDEs required for convergence to the true probability density of x_2 . Figure 2 shows that the coupling of

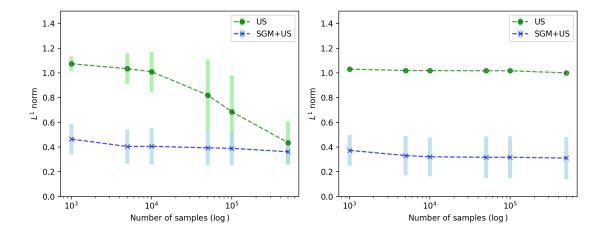


FIG. 5. The L^1 norm of the absolute value of the difference between the true PDF and the estimated PDF determined by US (green) and coupled SGM+US (blue) runs at two different barrier heights. On the left, the barrier height is h = 4. On the right, h = 8, L^1 norm is a function of the sample size (*i.e.*, number of time steps). Each data point and its corresponding error bar reflects the results of 1,000 independent experiments.

a SGM with US results in a further increased performance, as compared to US alone, especially at smaller sample sizes, because US alone can often become trapped in the metastable states of the system.

B. Manifold Learning For Identifying Collective Variables

Manifold learning or dimensionality reduction techniques can be used if the slow direction is not known *a priori*, as mentioned in Section III. In previous work with conditional generative adversarial networks¹⁷ we investigated in some detail the training of generative models on data labeled by values of slow directions identified by nonlinear dimensionality reduction; we also demonstrated performing biased sampling directly on these data-mined variables.

This same workflow can now be implemented with SGMs. In order to produce initial conditions for the US simulations, we first run a long unbiased simulation of this system. In this case, the dataset consisted of 10,000 samples from a simulation of these SDEs with $dt = 10^{-2}$. This is downsampled from the previous example for memory considerations for the next dimensionality reduction step. We then use nonlinear dimensionality reduction, in this case Diffusion Maps⁶, to identify a coordinate that one-to-one with the slow direction. We finally train the SGM on the data set labeled by the leading diffusion map coordinate Φ .

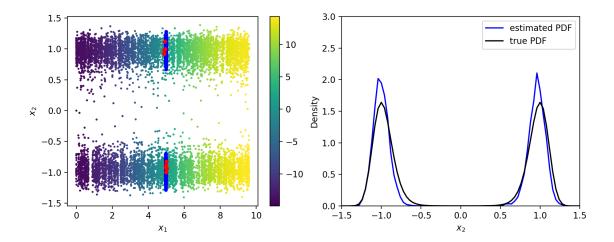


FIG. 6. On the left, outputs of 10 US simulations of 1,000 time steps (blue) conditioned on the Diffusion Map coordinate value $\Phi = 0$ (which corresponds to the approximate halfway point of label produced by diffusion maps that is one-to-one with the slow direction x_1) initialized by SGM generated points (red) on top of a background of the training data for the SGM colored by the diffusion map coordinate. The training data is much more sparse than previous examples due to the downsampling performed for the diffusion maps. On the right, the probability density function estimated from pooling the histograms of the coupled US+SGM simulations (blue line) compared to the true probability density function given by the known x_2 equation of the system (black line).

The approximated PDF of this example is less accurate than the example labeled by the known slow direction; this should be attributed to the downsampling of the data set and the resulting smaller number of points in the low probability/high energy areas of x_2 space.

C. SGM-Generated Initial Conditions for Molecular Dynamics

SGMs can be trained to generate initial conditions of high dimensional problems. For example in the field of computational chemistry, exploring the conformation space of new molecules of interest is an important step in workflows for drug discovery and materials science. Consider a 20 nanosecond molecular dynamics (MD) simulation of alanine dipeptide performed using the OpenMM molecular dynamics package with the Amber14 force field in implicit solvent, stochastic (Langevin) dynamics performed at 300 Kelvin, a time step of 1 fs, and a collision rate of 1 ps⁻¹.^{27–29} This simulation sampled two local free energy basins at distinct values of Φ and Ψ

Score-Based Model Assisted Sampling

(the commonly accepted CVs, which are dihedral angles that characterize the bulk behavior of the molecule). We then train an SGM on the molecular conformations from this simulation labeled by their values of Ψ . The result is an SGM that can generate alanine dipeptide molecules that are consistent with a desired CV value. These in turn can be used as proposed in section 4.1 to initialize US simulations and approximate free energy surfaces along the CV coordinate through the use of WHAM. Note that SGMs have been used to generate molecular conformations 30,31 , but coupling SGMs with physics-based models has not yet been investigated, to our knowledge. All parameters of the SGM SDE were kept consistent with previous examples, as were the learning rates; however, the score network that was used for these molecules was larger than the previous networks for the SDE dynamical system examples.

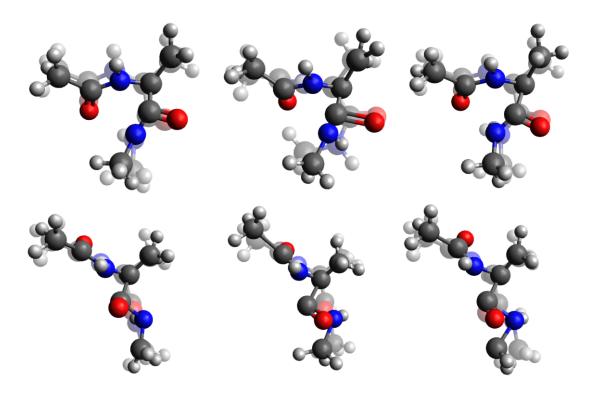


FIG. 7. Alanine dipeptide molecular conformations produced by an SGM. In the top row are alanine molecules which have a Ψ value equal to approximately 4 radians. In the bottom row, the alanine molecules Ψ value is equal to approximately 6 radians. The transparent conformation visible "behind" each molecule is the same molecule after its energy is minimized through 50 steps of a steepest descent algorithm.

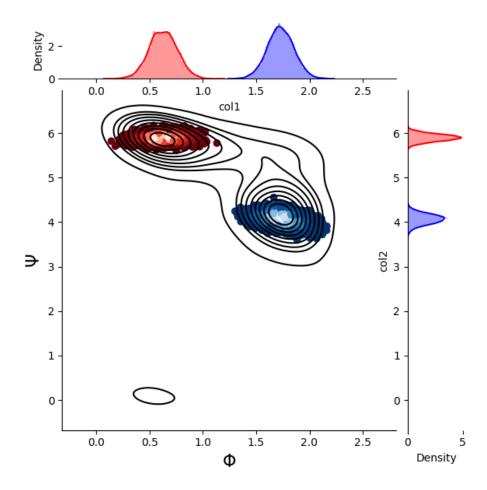


FIG. 8. Dihedral angles of many SGM output samples; red(blue) correspond to conformations produced by a SGM conditioned on the Ψ value of the left(right) free energy minimum. Color gradients (dark red to light red, dark blue to light blue) represent empirical sample densities. The black contour lines represent the empirical sample density of a long unbiased simulation run that was used to train the SGM. The empirical marginals for the SGM samples are included on the margins.

VI. CONCLUSIONS

Conditional SGMs, like other conditional generative models, exhibit considerable parallels with physics-based sampling methods. Coupling the ML-based SGMs with established approaches in physics and computational chemistry can mitigate the need for guessing appropriate initial conditions for simulation and can also alleviate the need for long equilibration simulations starting at such initial conditions. Moreover, coupling SGMs with US (and WHAM where relevant) provides a faster convergence to the true joint distribution when compared to US alone (and may even pro-

Score-Based Model Assisted Sampling

vide convergence to the joint distribution when US alone will not realistically converge). In future work, we hope to use this framework to investigate systems with more complex equilibrium distributions. In particular, we hope to investigate more complex molecular model-based multiscale systems as well as dynamical systems in which bifurcations are present.

ACKNOWLEDGMENTS

The authors wish to acknowledge the support of the US Department of Energy and the Air Force Office of Scientific Research (through an AFOSR MURI). We also thank Holden Lee for insightful discussions.

DATA AVAILABILITY STATEMENT

The data and code that supports the findings of this study are openly available in GitHub at https://github.com/ecrab/SGM_assisted_sampling

Appendix A: SGMs with Manifold Learning Continued

In subsection V B, we showed that, if the CVs of a dynamical system of interest are not known a priori, one could use nonlinear dimensionality reduction to identify low dimension variables that are one-to-one with the CVs of the system. Consider again the example used in IV. Instead of using the slow direction, z_1 , as the label for the cSGM, we use diffusion maps to identify a leading eigenvector coordinate Φ_1 that is one-to-one with z_1 . The training data used for this model is more sparse (by a factor of 20) than the original example in IV due to the cost of calculating the transition matrix for diffusion maps.

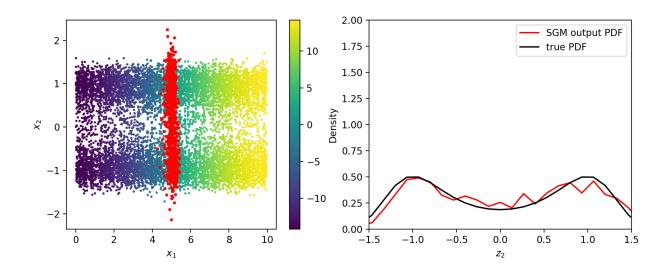


FIG. 9. On the left, output from a trained SGM conditioned on the leading Diffusion Maps coordinate value $\Phi_1 = 0$ (red) on top of a background of the training data for the SGM colored by Φ_1 values. On the right, the PDF estimated from the SGM output is compared to the true PDF given by the known z_2 equation of the system (black line).

REFERENCES

¹Jérôme Hénin, Tony Lelièvre, Michael R. Shirts, Omar Valsson, and Lucie Delemotte. Enhanced sampling methods for molecular dynamics simulations [article v1.0]. *Living Journal of Computational Molecular Science*, 4(1):1583, Dec. 2022.

²Karl Pearson. LIII. On lines and planes of closest fit to systems of points in space, November 1901.

³Mark A. Kramer. Nonlinear principal component analysis using autoassociative neural networks. *AIChE Journal*, 37(2):233–243, 1991.

⁴Joshua B. Tenenbaum, Vin de Silva, and John C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

⁵Sam T. Roweis and Lawrence K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

⁶Ronald R. Coifman and Stéphane Lafon. Diffusion maps. *Applied and Computational Harmonic Analysis*, 21(1):5–30, 2006. Special Issue: Diffusion Maps and Wavelets.

- ⁷C. William Gear, James M. Hyman, Panagiotis G Kevrekidid, Ioannis G. Kevrekidis, Olof Runborg, and Constantinos Theodoropoulos. Equation-Free, Coarse-Grained Multiscale Computation: Enabling Mocroscopic Simulators to Perform System-Level Analysis. *Communications in Mathematical Sciences*, 1(4):715 762, 2003.
- ⁸Ioannis G. Kevrekidis and Giovanni Samaey. Equation-free multiscale computation: Algorithms and applications. *Annual Review of Physical Chemistry*, 60(1):321–344, 2009. PMID: 19335220.
- ⁹G.M. Torrie and J.P. Valleau. Nonphysical sampling distributions in monte carlo free-energy estimation: Umbrella sampling. *Journal of Computational Physics*, 23(2):187–199, 1977.
- ¹⁰I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative Adversarial Networks. *arXiv:1406.2661 [cs, stat]*, June 2014. arXiv: 1406.2661.
- ¹¹Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2013.
- ¹²M. Mirza and S. Osindero. Conditional Generative Adversarial Nets. *arXiv:1411.1784* [cs, stat], November 2014. arXiv: 1411.1784.
- ¹³Hythem Sidky, Wei Chen, and Andrew L. Ferguson. Molecular latent space simulators. *Chem. Sci.*, 11:9459–9467, 2020.
- ¹⁴Zineb Belkacemi, Paraskevi Gkeka, Tony Lelièvre, and Gabriel Stoltz. Chasing collective variables using autoencoders and biased trajectories. *Journal of Chemical Theory and Computation*, 18(1):59–78, 2022. PMID: 34965117.
- ¹⁵Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations, 2020.
- ¹⁶Yang Song and Stefano Ermon. Improved techniques for training score-based generative models, 2020.
- ¹⁷Ellis R. Crabtree, Juan M. Bello-Rivas, Andrew L. Ferguson, and Ioannis G. Kevrekidis. Gans and closures: Micro-macro consistency in multiscale modeling. *Multiscale Modeling & Simulation*, 21(3):1122–1146, 2023.
- ¹⁸Gisiro Maruyama. Continuous markov processes and stochastic equations. *Rendiconti del Circolo Matematico di Palermo*, 4(1):48–90, 1955.
- ¹⁹Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution, 2019.

- ²⁰Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural computation*, 23(7):1661–1674, 2011.
- ²¹Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. Csdi: Conditional score-based diffusion models for probabilistic time series imputation, 2021.
- ²²Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J. Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement, 2021.
- ²³Georgios Batzolis, Jan Stanczuk, Carola-Bibiane Schönlieb, and Christian Etmann. Conditional image generation with score-based diffusion models, 2021.
- ²⁴Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models, 2022.
- ²⁵Shankar Kumar, John M. Rosenberg, Djamal Bouzida, Robert H. Swendsen, and Peter A. Kollman. The weighted histogram analysis method for free-energy calculations on biomolecules. i. the method. *Journal of Computational Chemistry*, 13(8):1011–1021, 1992.
- ²⁶Michael R. Shirts and John D. Chodera. Statistically optimal analysis of samples from multiple equilibrium states. *The Journal of Chemical Physics*, 129(12):124105, 2008.
- ²⁷Peter Eastman, Jason Swails, John D. Chodera, Robert T. McGibbon, Yutong Zhao, Kyle A. Beauchamp, Lee-Ping Wang, Andrew C. Simmonett, Matthew P. Harrigan, Chaya D. Stern, Rafal P. Wiewiora, Bernard R. Brooks, and Vijay S. Pande. Openmm 7: Rapid development of high performance algorithms for molecular dynamics. *PLOS Computational Biology*, 13(7):1–17, 07 2017.
- ²⁸David Case, V. Babin, Joshua Berryman, R. Betz, Qin Cai, D. Cerutti, Thomas Cheatham, Thomas Darden, Robert Duke, H. Gohlke, Andreas Götz, Sergey Gusarov, N. Homeyer, Pawel Janowski, J. Kaus, István Kolossváry, Andriy Kovalenko, Tai-Sung Lee, S. Legrand, and P. Kollman. Amber 14, 2014.
- ²⁹John Mongan, Carlos Simmerling, J. Andrew McCammon, David A. Case, and Alexey Onufriev. Generalized born model with a simple, robust molecular volume correction. *Journal of Chemical Theory and Computation*, 3(1):156–169, 2007. PMID: 21072141.
- ³⁰Gabriele Corso, Hannes Stärk, Bowen Jing, Regina Barzilay, and Tommi Jaakkola. Diffdock: Diffusion steps, twists, and turns for molecular docking, 2023.
- ³¹Dwaraknath Gnaneshwar, Bharath Ramsundar, Dhairya Gandhi, Rachel Kurchin, and Venkatasubramanian Viswanathan. Score-based generative models for molecule generation, 2022.