STANLEY: Stochastic Gradient Anisotropic Langevin Dynamics for Learning Energy-Based Models

Abstract

We propose in this paper, **STANLEY**, a **ST**ochastic gradient **AN**isotropic **L**ang**E**vin d**Y**namics, for sampling high dimensional data. With the growing efficacy and potential of Energy-Based modeling, also known as non-normalized probabilistic modeling, for modeling a generative process of different natures of high dimensional data observations, we present an end-to-end learning algorithm for Energy-Based models (EBM) with the purpose of improving the quality of the resulting sampled data points. While the unknown normalizing constant of EBMs makes the training procedure intractable, resorting to Markov Chain Monte Carlo (MCMC) is in general a viable option. Realizing what MCMC entails for the EBM training, we propose in this paper, a novel high dimensional sampling method, based on an anisotropic stepsize and a gradient-informed covariance matrix, embedded into a discretized Langevin diffusion. We motivate the necessity for an anisotropic update of the negative samples in the Markov Chain by the nonlinearity of the backbone of the EBM, here a Convolutional Neural Network. Our resulting method, namely STANLEY, is an optimization algorithm for training Energy-Based models via our newly introduced MCMC method. We provide a theoretical understanding of our sampling scheme by proving that the sampler leads to a geometrically uniformly ergodic Markov Chain. Several image generation experiments are provided in our paper to show the effectiveness of our method.

1 Introduction

The modeling of a data generating process is critical for many modern learning tasks. A growing interest in generative models within the realm of computer vision has led to multiple interesting solutions. In particular, Energy-Based models (EBM) (Zhu, Wu, and Mumford 1998; LeCun et al. 2006), are a class of generative models that learns high dimensional and complex (in terms of landscape) representation/distribution of the input data. Since inception, EBMs have been used in several applications including computer vision (Ngiam et al. 2011; Xie et al. 2016; Du and Mordatch 2019), natural language processing (Mikolov et al. 2013; Deng et al. 2020), density estimation (Wenliang et al. 2019; Song et al. 2020) and reinforcement learning (Haarnoja et al. 2017).

Formally, EBMs are built upon an unnormalized log probability, called the energy function, that is not required to sum to one as standard log probability functions. This noticeable feature allows for more freedom in the way one parameterizes the EBM. For instance, Convolutional Neural Network (CNN) can be employed to parameterize the energy function, see Xie et al. (2016). Note that the choice of the EBM backbone is highly related to the type of the input data.

The training procedure of such models consists of finding an energy function that assigns to lower energies to observations than unobserved points. This phase can be cast to an optimization task and several ways are possible to achieve it. In this paper, we will focus on training the EBM via Maximum Likelihood Estimation (MLE). Alternative procedures include learning EBMs using Noise Contrastive Estimation as in Gao et al. (2020). Particularly, while using MLE to fit the EBM on a stream of observed data, the high non-convexity of the loss function leads to a non closed form maximization step. In general, gradient based optimization methods are thus used during that phase. Besides, given the intractability of the normalizing constant of our model, the aforementioned gradient, which is an intractable integral, needs to be approximated. A popular and efficient way to conduct such approximation is to use Monte Carlo approximation where the samples are obtained via Markov Chain Monte Carlo (MCMC) (Meyn and Tweedie 2012). The goal of this embedded MCMC procedure while training the Energy-Based model is to synthesize new examples of the input data and use those new *synthetic* observations to approximate quantities of interest.

The sampling phase is thus crucial for both the EBM training speed and its final accuracy in generating new synthetic samples. The computational burden of those MCMC transitions, at each iteration of the EBM training procedure, is alleviated via different techniques in the literature. For instance, in Nijkamp et al. (2019), the authors develop a short-run MCMC as a flow-based generator mechanism despite its non convergence property. Other principled approach, as in Hinton (2002), keeps in memory the final chain state under the previous global model parameter and uses it as the initialization of the current chain. The heuristic of such approach is that along the EBM iterations, the conditional distributions, depending on the model parameter, are more and more similar and thus using a good sample from the previ-

ous chain is in general a good sample of the current one. Though, this method can be limited during the first iterations of the EBM training since when the model parameter changes drastically, the conditional distributions do change too, and samples from two different chains can be quite inconsistent. Several extensions modifying the way the chain is initialized can be found in Welling and Hinton (2002); Gao et al. (2018); Du and Mordatch (2019).

An interesting line of work in the realm of MCMC-based EBM tackles the biases induced by stopping the MCMC runs too early. Indeed, it is known, see Meyn and Tweedie (2012), that before convergence, MCMC samples are biased and thus correcting this bias while keeping a short and less expensive run is an appealing option. Several contributions aiming at removing this bias for improved MCMC training include coupling MCMC chains, see Qiu, Zhang, and Wang (2019); Jacob, O Leary, and Atchadé (2020) or simply estimating this bias and correct the chain afterwards, see Du et al. (2020).

Here, our work is in line with the context of – high-dimensional data, – EBM parameterized by deep neural networks and – MLE-based optimization via MCMC, which make our method particularly attractive to all of the above combined. We also consider the case of a short-run MCMC for the training of an EBM. Rather than focusing on debiasing the chain, we develop a new sampling scheme where the goal is to obtain better samples from the target distribution in fewer MCMC transitions. We consider that the shape of the target distribution, which inspires our proposed method, is of utmost importance to obtain such negative samples. Our contributions are summarized as follows:

- We develop STANLEY, an Energy-Based model training method that embeds a newly proposed *convergent* and *efficient* MCMC sampling scheme, focusing on curvature informed metrics of the target distribution one wants to sample from.
- Based on an anisotropic stepsize, our method, which is an improvement of the Langevin Dynamics, achieves to obtain negative samples from the Energy-Based model data distribution and improves the overall optimization algorithm.
- We prove the geometric ergodicity uniformly on any compact set of our MCMC method assuming some regularity conditions on the target distribution and on the backbone model of the EBM.
- We empirically assess the effectiveness of our method on several image generation tasks, both on synthetic and real datasets including the Oxford Flowers 102 dataset, CIFAR-10 and CelebA. We conclude the work with an Image inpainting experiment on a benchmark dataset.

Section 2 introduces important notations and related work. Section 3 develops the main algorithmic contribution of this paper, namely STANLEY. Section 4 presents our main theoretical results focusing on the ergodicity of the proposed MCMC sampling method. Section 5 presents several image generation experiments on both synthetic and real datasets. The complete proofs of our theoretical results can be found in the supplementary material.

2 On MCMC based Energy Based Models

Given a stream of input data noted $x \in \mathcal{X} \subset \mathbb{R}^p$, the Energy-Based model (EBM) is a Gibbs distribution defined as follows:

$$p(x, \theta) = \frac{1}{Z(\theta)} \exp(f_\theta(x)) \ , \tag{1}$$

where $\theta \in \Theta \subset \mathbb{R}^d$ denotes the global parameters vector of our model and $Z(\theta) := \int_x \exp(f_\theta(x)) \mathrm{d}x$ is the normalizing constant (with respect to x). The natural way of fitting model (1) is to employ Maximum Likelihood Estimation (MLE) maximizing the marginal likelihood $p(\theta)$, *i.e.,* finding the vector θ^* such that for any $x \in \mathcal{X}$,

$$\theta^* = \arg\max_{\theta \in \Theta} \log p(\theta) \ , \tag{2}$$

where the quantity of interest $p(\theta)$ is obtained by marginalizing over the input data $x \in \mathcal{X}$ and formally reads $p(\theta) := \int_{x \in \mathcal{X}} p(x, \theta) q(x) \mathrm{d}x$. We denote by $q(x)$ the true distribution of the input data x. The optimization task (2) is not tractable in closed form and requires an iterative procedure in order to be solved. The standard algorithm used to train EBMs is Stochastic Gradient Descent (SGD), see Robbins and Monro (1951); Bottou and Bousquet (2008). SGD requires having access to the gradient of the objective function $\log p(\theta)$. This latter requires computing an intractable integral, due to the high nonlinearity of the generally utilized parameterized model $f_\theta(x)$. Given the general form defined in (1), we have that:

$$\begin{aligned} \nabla \log p(\theta) &= \int_{x \in \mathcal{X}} \nabla_\theta \log p(x, \theta) q(x) \mathrm{d}x \\ &= \mathbb{E}_{q(x)}[\nabla_\theta f_\theta(x)] - \mathbb{E}_{p(x,\theta)}[\nabla_\theta f_\theta(x)] \ , \end{aligned}$$

and a simple Monte Carlo approximation of $\nabla \log p(\theta)$ yields the following important expression of the gradient

$$\nabla \log p(\theta) \approx \frac{1}{n} \sum_{i=1}^{n} \nabla_\theta f_\theta(x_i^q) - \frac{1}{M} \sum_{m=1}^{M} \nabla_\theta f_\theta(z_m) \ , \tag{3}$$

where $\{z_m\}_{m=1}^M$ are samples obtained from the EBM $p(x, \theta)$ and $\{x_i^q\}_{i=1}^n$ are drawn uniformly from the true data distribution $q(x)$. While drawing samples from the data distribution is trivial, the challenge during the EBM training phase is to obtain good samples from the EBM distribution $p(x, \theta)$ for any model parameter $\theta \in \Theta$. This task is generally done using MCMC methods. State-of-the-art MCMC used in the EBM literature include Langevin Dynamics, see Grenander and Miller (1994); Roberts and Tweedie (1996) and Hamiltonian Monte Carlo (HMC), see Neal et al. (2011). Those methods are detailed in the sequel and are important concepts used throughout our paper.

Energy Based Models: Energy based models (LeCun et al. 2006; Ngiam et al. 2011) are a class of generative models that leverages the power of Gibbs potential and high dimensional sampling techniques to produce high quality synthetic image samples. Just as Variational Autotencoders (VAE) (Kingma and Welling 2013) or Generative Adversarial Networks (GAN) (Goodfellow et al. 2014), EBMs are

powerful tools for generative modeling tasks, as a building block for a wide variety of tasks. The main purpose of EBMs is to learn an energy function (1) that assigns low energy to a stream of observation and high energy values to other inputs. In several general applications, authors leverage the power of EBMs for develop an energy-based optimal policy where the parameters of that energy function are provided by the reward of the overall system. Learning EBMs with alternative strategies include contrastive divergence (CD) (Hinton 2002; Tieleman 2008), noise contrastive estimation (NCE) (Gutmann and Hyvärinen 2010; Gao et al. 2020) , introspective neural networks (INN) (Lazarow, Jin, and Tu 2017; Jin, Lazarow, and Tu 2017; Lee et al. 2018), cooperative networks (CoopNets) (Xie, Zheng, and Li 2021; Xie et al. 2021a), f-divergence (Yu et al. 2020), and triangle divergence (Han et al. 2019, 2020). Recently, EBMs parameterized by modern neural networks have drawn much attention from the computer vision and machine learning communities. Successful applications with EBMs include generations of images (Xie et al. 2016; Gao et al. 2018; Du and Mordatch 2019), videos (Xie, Zhu, and Wu 2017, 2019), 3D volumetric shapes (Xie et al. 2018, 2020), texts (Deng et al. 2020), molecules (Ingraham et al. 2018; Du et al. 2019), as well as image-to-image translation (Xie et al. 2021a,b), out-of-distribution detection (Liu et al. 2020), inverse optimal control (Xu et al. 2019) and deep regression (Gustafsson et al. 2020). Yet, unlike VAE or GAN Energy-Based models enjoy from a single structure requiring training (versus several networks) resulting in more stability. The use of implicit sampling techniques, such as MCMC, as detailed in the sequel, allows more flexibility by trading off sample quality for computation time. Overall, the *implicit* property of the EBM, seen as an energy function, makes it a tool of choice as opposed to *explicit* generators that are limited to some design choices, such as the choice of the prior distribution for VAEs or both neural networks design in GANs.

MCMC procedures: Whether for sampling from a posterior distribution (Robert and Casella 2010), or in general intractable likelihoods scenario (Doucet, Godsill, and Andrieu 2000), various inference methods are available. Approximate inference is a partial solution to the inference problem and include techniques such as Variational Inference (VI) (Wainwright and Jordan 2008; De Freitas et al. 2001) or Laplace Approximation (Wolfinger 1993; Rue, Martino, and Chopin 2009). Those methods allow the simplification of the intractable quantities and result in the collection of good, yet approximate, samples. As seen in (3), training an EBM requires obtaining samples from the model itself. Given the nonconvexity of the structural model $f_\theta(\cdot)$ with respect to the model parameter θ, direct sampling is not an option. Besides, in order to update the model parameter θ, usually through gradient descent type of methods (Bottou and Bousquet 2008), exact samples from the EBM are needed in order to compute a good approximation of its (intractable) gradient, see (3). To do so, we generally have recourse to MCMC methods. MCMC are a class of inference algorithms that provide a principled iterative approach to obtain samples from any intractable distribution. While being exact, the samples generally represent a larger computation burden than methods such as VI. Increasing the efficiency of MCMC methods, by obtaining exact samples, in other words constructing a chain that converges faster, in fewer transitions is thus of utmost importance in the context of optimizing EBMs. Several attempts have been proposed for the standalone task of posterior sampling through the use of Langevin diffusion, see the Unadjusted Langevin in Brosse et al. (2017), the MALA algorithm in Roberts and Rosenthal (1997); Roberts and Tweedie (1996); Durmus et al. (2017) or leveraging Hamiltonian Dynamics as in Girolami and Calderhead (2011). We propose in the next section, an improvement of the Langevin diffusion with the ultimate goal of speeding the EBM training procedure. Our method includes this latter improvement in an end-to-end learning algorithms for Energy-Based models.

3 Gradient Informed Langevin Diffusion

We now introduce the main algorithmic contribution of our paper, namely STANLEY. STANLEY is a learning algorithm for EBMs, comprised of a novel MCMC method for sampling negative samples from the intractable model (1). We provide theoretical guarantees of our scheme in Section 4.

Preliminaries on Langevin MCMC based EBM

State-of-the-art MCMC sampling algorithm, particularly used during the training procedure of EBMs, is the discretized Langevin diffusion, cast as Stochastic Gradient Langevin Dynamics (SGLD), see Welling and Teh (2011). In particular, several applications using EBM and SGLD have thrived in image generation, natural language processing or even biology (Du et al. 2019). Yet, the choice of the proposal, generally Gaussian, is critical for improving the performances of both the sampling step (inner loop of the whole procedure) and the EBM training. We recall the vanilla discretized Langevin diffusion used in the related literature as follows:

$$z_k = z_{k-1} + \frac{\gamma}{2}\nabla \log \pi_\theta(z_k) + \sqrt{\gamma} B_k \ ,$$

where $\pi_\theta(\cdot) := p(\cdot, \theta)$ is the target potential one needs samples from and defined in (1), z_k represents the states of the chains at iteration k, *i.e.,* the generated samples in the context of EBM, k is the MCMC iteration index and B_k is the Brownian motion, usually set as a Gaussian noise and which can be written as $B_k := \epsilon\, \xi_k$ where ξ_k is a standard Gaussian random variable and ϵ is a scaling factor for implementation purposes. This method directs the proposed moves towards areas of high probability of the stationary distribution π_θ, for any $\theta \in \Theta$, using the gradient of $\log \pi_\theta$ and has been the object of several studies (Girolami and Calderhead 2011; Cotter et al. 2013). In high dimensional and highly nonlinear settings, the burden of computing this gradient for a certain number of MCMC transitions leads to a natural focus: the improvement of the behaviour of such sampling scheme by assimilating information about the landscape of the target distribution, also called stationary, while keeping its ease of implementation.

STANLEY, an Anisotropic Energy Based Modeling Approach

Given the drawbacks of current MCMC methods used for training EBMs, we introduce a new sampler based on the Langevin updates presented above in Step 4 of Algorithm 1.

Algorithm 1: STANLEY for Energy-Based model

1: **Input**: Total number of iterations T, number of MCMC transitions K and of samples M, sequence of global learning rate $\{\eta_t\}_{t>0}$, stepsize threshold th, initial value θ_0, MCMC initialization $\{z_0^m\}_{m=1}^M$ and observations $\{x_i\}_{i=1}^n$.
2: **for** $t = 1$ to T **do**
3: Compute the anisotropic stepsize as follows:

$$\gamma_t = \frac{\mathsf{th}}{\max(\mathsf{th}, |\nabla f_{\theta_t}(z_{t-1}^m)|)} \ . \tag{4}$$

4: Draw M samples $\{z_t^m\}_{m=1}^M$ from the objective potential (1) via Langevin diffusion:
5: **for** $k = 1$ to K **do**
6: Construct the Markov Chain as follows:

$$z_k^m = z_{k-1}^m + \gamma_t/2 \nabla f_{\theta_t}(z_{k-1}^m) + \sqrt{\gamma_t}\mathsf{B}_k \ , \tag{5}$$

where B_k denotes the Brownian motion.
7: **end for**
8: Sample m positive observations $\{x_i\}_{i=1}^m$ from the empirical data distribution.
9: Compute the gradient of the empirical log-EBM (1):

$$\nabla \log p(\theta_t) = \mathbb{E}_{p_{\text{data}}}\left[\nabla_\theta f_{\theta_t}(x)\right] - \mathbb{E}_{p(x,\theta)}\left[\nabla_{\theta_t} f_\theta(z_t)\right]$$
$$\approx \frac{1}{n}\sum_{i=1}^n \nabla_\theta f_{\theta_t}(x_i) - \frac{1}{M}\sum_{m=1}^M \nabla_\theta f_{\theta_t}(z_K^m) \ .$$

10: Update the vector of global parameters of the EBM:

$$\theta_{t+1} = \theta_t + \eta_t \nabla \log p(\theta_t) \ .$$

11: **end for**
12: **Output:** Vector of fitted parameters θ_{T+1}.

Intuitions behind the efficacy of STANLEY: Some past modifications have been proposed in particular to optimize the covariance matrix of the proposal of the general MCMC procedure in order to better stride the support of the target distribution. Langevin Dynamics is one example of those improvements where the proposal is a Gaussian distribution where the mean depends on the gradient of the log target distribution and the covariance depends on some Brownian motion. For instance, in Atchadé (2006); Marshall and Roberts (2012), the authors propose adaptive and geometrically ergodic Langevin chains. Yet, one important characteristic of our EBM problem, is that for each model parameter updated through the training iterations, the target distribution moves and the proposal should take that adjustment into account. The techniques in Atchadé (2006); Marshall and Roberts (2012) does not take the whole advantage of changing the proposal using the target distribution. In particular, the covariance matrix of the proposal is given by a stochastic approximation of the empirical covariance matrix. This choice seems completely relevant as soon as the convergence towards the stationary distribution is reached, in other words it would make sense towards the end of the EBM training, as the target distributions from a model parameter to the next one are similar. However, it does not provide a good guess of the variability during the first iterations since it is still very dependent on the initialization.

Moreover, in Girolami and Calderhead (2011), the authors consider the approximation of a constant. Even though this simplification leads to ease of implementation, the curvature metric chosen by the authors need to be inverted, step that can be a computational burden if not intractable. Especially in the case we are considering in our paper, *i.e.,* ConvNet-based EBM, where the high nonlinearity would lead to intractable expectations.

Therefore, in (4) and (5) of Algorithm 1, we propose a variant of Langevin Dynamics, in order to sample from a target distribution, using a full anisotropic covariance matrix based on the anisotropy and correlations of the target distribution, see the $\sqrt{\gamma_t}\mathsf{B}_k$ term.

4 Geometric ergodicity of STANLEY

We will present in this section, our theoretical analysis for the Markov Chain constructed using Line 3-4.

Let Θ be a subset of $\mathbb{R}^d$ for some integer $d > 0$. We denote by $\mathcal{Z}$ the measurable space of $\mathbb{R}^\ell$ for some integer $\ell > 0$. We define a family of stationary distribution $(\pi_\theta(z))_{\theta\in\Theta}$, probability density functions with respect to the Lebesgue measure on the measurable space $\mathcal{Z}$. This family of p.d.f. defines the stationary distributions of our newly introduced sampler.

Notations and Assumptions

For any chain state $z \in \mathcal{Z}$, we denote by $\Pi_\theta(z, \cdot)$ the transition kernel as defined in the STANLEY update in Line 4. The objective of this section is to rigorously show that each transition kernel π_θ, for any parameter $\theta \in \Theta$ is geometrically ergodic and that this result is true uniformly on the model parameter θ and on any compact subset $\mathcal{C} \in \mathcal{Z}$. As a background note, a Markov chain, as constructed Line 4, is said to be geometrically ergodic when k iterations of the same transition kernel is converging to the stationary distribution of the chain with a geometric dependence on k.

As in Allassonniere and Kuhn (2015), we state the assumptions required for our analysis. The first one is related to the continuity of the gradient of the log posterior distribution and the unit vectors pointing in the direction of the sample z and in the direction of the gradient of the log posterior distribution at z:

H1. *For all $\theta \in \Theta$, the structural model $f_\theta(\cdot)$ satisfies:*

$$\lim_{|z|\to\infty} \frac{z}{|z|}\nabla f_\theta(z) = -\infty \ ,$$
$$\lim\sup_{|z|\to\infty} \frac{z}{|z|}\frac{\nabla f_\theta(z)}{|\nabla f_\theta(z)|} < 0 \ .$$

We assume also some regularity conditions of the stationary distributions with respect to the model parameter θ:

H2. *$\theta \to \pi_\theta$ and $\theta \to \nabla \log \pi_\theta$ are continuous on Θ.*

For a positive and finite function noted $V : \mathcal{Z} \mapsto \mathbb{R}$, we define the V-norm distance between two arbitrary transition kernels Π_1 and Π_2 as follows:

$$\|\Pi_1 - \Pi_2\|_V := \sup_{z \in \mathcal{Z}} \frac{\|\Pi_1(z, \cdot) - \Pi_2(z, \cdot)\|_V}{V(z)} .$$

The definition of this norm allows us to establish a convergence rate for our sampling method by deriving an upper bound of $\|\Pi_\theta^k - \pi_\theta\|_V$ where $k > 0$ denotes the number of MCMC transitions. We also recall that Π_θ is the transition kernel defined by Line 4 and π_θ is the stationary distribution of our Markov chain at a given EBM model θ. This quantity characterizes how close to the target distribution, our chain is getting after a finite time of iterations and will eventually formalize *V-uniform ergodicity* of our method. We specify that strictly speaking π_θ is a probability measure, and not a transition kernel. However $\|\Pi_\theta^k - \pi_\theta\|_V$ is well-defined if we consider the probability π_θ as a kernel:

$$\pi(z, \mathcal{C}) := \pi(\mathcal{C}) \quad \text{for} \quad \mathcal{C} \subset \mathcal{Z}, \quad z \in \mathcal{Z} .$$

Here, for some $\beta \in]0, 1[$ we define the V_θ function, also know as the *drift*, for all $z \in \mathcal{Z}$ as follows:

$$V_\theta(z) := c_\theta \pi_\theta(z)^{-\beta} , \tag{6}$$

where c_θ is a constant, with respect to the chain state z, such that for all $z \in \mathcal{Z}$, $V_\theta(z) \geq 1$. Note that the V norm depends on the chain state noted z *and* of the global model parameter θ varying through the optimization procedure. Yet, in both main results, the ergodicity and the convergence rate, including the underlying drift condition, are established uniformly on the parameter space Θ. We also define the auxiliary functions, independent of the parameter θ as:

$$V_1(z) := \inf_{\theta \in \Theta} V_\theta(z) \quad \text{and} \quad V_2(z) := \sup_{\theta \in \Theta} V_\theta(z) , \tag{7}$$

and assume the following:

H3. *There exists a constant $a_0 > 0$ such that for all $\theta \in \Theta$ and $z \in \mathcal{Z}$, the function $V_2^{a_0}(z)$, defined in* (7)*, is integrable against the kernel $\Pi_\theta(z, \cdot)$ and we have*

$$\limsup_{a \to 0} \sup_{\theta \in \Theta, z \in \mathcal{Z}} \Pi_\theta V_2^a(z) = 1 .$$

Convergence Results

The result consists of showing V-uniform ergodicity of the chain, the irreducibility of the transition kernels and their aperiodicity, see Meyn and Tweedie (2012); Allassonniere and Kuhn (2015) for more details. We also prove a drift condition which states that the transition kernels tend to bring back elements into a small set. Then, V-uniform ergodicity of the transition kernels $(\Pi_\theta)_{\theta \in \Theta}$ boils down from the latter proven drift condition.

Important Note: The stationary distributions depends on $\theta \in \Theta$ as they vary at each model update during the EBM optimization phase. Thus uniformity of convergence of the chain is important in order to characterize the sampling phase *throughout the entire training phase*. Particularly at the beginning, the shape of the distributions one needs to sample from vary a lot from a parameter to another.

Theorem 1. *Assume H1-H3. For any $\theta \in \Theta$, there exists a drift function V_θ, a set $\mathcal{O} \subset \mathcal{Z}$, a constant $0 < \epsilon \leq 1$ such that*

$$\Pi_\theta(z, \mathcal{B}) \geq \epsilon \int_{\mathcal{B}} 1_{\mathcal{X}}(z) dy . \tag{8}$$

Moreover there exists $0 < \mu < 1$, $\delta > 0$ and a drift function V, now independent of θ such that for all $z \in \mathcal{Z}$:

$$\Pi_\theta V(z) \leq \mu V(z) + \delta 1_{\mathcal{O}}(z) . \tag{9}$$

Theorem 1 shows two important convergence results for our sampling method. First, it established the existence of a small set $\mathcal{O}$ leading to the crucially needed aperiodicity of the chain and ensuring that each transition moves toward a better state. Then, it provide a uniform ergodicity result of our sampling method in STANLEY, via the so-called *drift condition* providing the guarantee that our user-designed transition kernels $(\Pi_\theta)_{\theta \in \Theta}$ attracts the states into the small set $\mathcal{O}$.

Moreover, the independence on the EBM model parameter θ of V in (9) leads to *uniform* ergodicity as shown in the following Corollary. While Theorem 1 is critical for proving the aperiodicity and irreducibility of the chain, we now establish the geometric speed of convergence of the chain. We do not only show the importance of the *uniform* ergodicity of the chain, which makes it appealing for the EBM training since the model parameter θ is often updated, but we also derive a geometrical rate in the following Corollary:

Corollary 1. *Assume H1-H3. A direct consequence of Theorem 1 is that the family of transition kernels $(\Pi_\theta)_{\theta \in \Theta}$ are uniformly ergodic,i.e., for any compact $\mathcal{C} \subset \mathcal{Z}$, there exist constants $\rho \in]0, 1[$ and $e > 0$ such for any MCMC iteration $k > 0$,we have:*

$$\sup_{z \in \mathcal{C}} \|\Pi_\theta^k u(\cdot) - \pi_\theta u(\cdot)\|_V \leq e \rho^k \|u\|_V , \tag{10}$$

where V is the drift function used in Theorem 1 and $u(\cdot)$ is any bounded function we apply a transition to.

We encourage the readers to read through the sketch of the main Theorem of our paper provided on the first page of the supplemental as we give the important details leading to the desired ergodicity results. Those various techniques are common in the MCMC literature and we refer the readers to several MCMC handbooks such as Neal et al. (2011); Meyn and Tweedie (2012) for more understanding.

5 Numerical Experiments

We present in this section a collection of numerical experiments to show the effectiveness of our method, both on synthetic and real datasets After verifying the advantage of STANLEY on a Gaussian Mixture Model (GMM) retrieving the synthetic data observations, we then investigate its performance when learning a distribution over high-dimensional natural images such as pictures of flowers, see the Flowers dataset in Nilsback and Zisserman (2008), or

general concepts featured in CIFAR-10 (Krizhevsky and Hinton 2009). For both methods, we use the Frechet Inception Distance (FID), as a reliable performance metrics as detailed in Heusel et al. (2017). In the sequel, we tune the learning rates over a fine grid and report the best result for all methods. For our method STANLEY, the threshold parameter th, crucial for the implementation of the stepsize (4) is tuned over a grid search as well. As mentioned in the above, we also define a Brownian motion as $B_k := \epsilon\, \xi_k$, and tune the scaling factor ϵ for better performances.

Toy Example: Gaussian Mixture Model

Datasets. We first demonstrate the outcomes of both methods including our newly proposed STANLEY for low-dimensional toy distributions. We generate synthetic 2D rings data and use an EBM to learn the true data distribution and put it to the test of generating new synthetic samples.

Methods and Settings. We consider two methods. Methods are ran with *nonconvergent* MCMC, *i.e.,*, we do not necessitate the convergence to the stationary distribution of the Markov chains. The number of transitions of the MCMC is set to $K = 100$ per EBM iteration. We use a standard deviation of 0.15 as in Nijkamp et al. (2020). Both methods have a constant learning rate of 0.14. The value of the threshold th for our STANLEY method is set to $\mathsf{th} = 0.01$. The total number of EBM iterations is set to $T = 10\,000$. The global learning rate η is set to a constant equal to 0.0001.

Network architectures. For the backbone of the EBM model, noted $f_\theta(\cdot)$ in (1), we chose a CNN of 5 2D convolutional layers and Leaky ReLU activation functions, with the leakage parameter set to 0.05. The number of hidden neurons varies between 32 and 64.

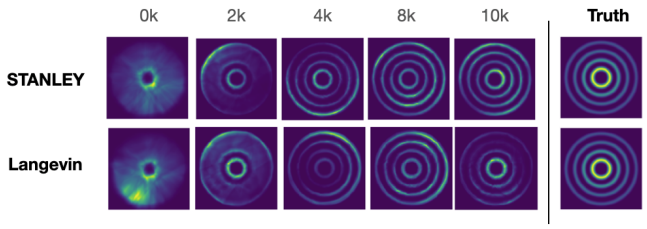

Figure 1: (Rings Toy Dataset) Top: our method, namely STANLEY Bottom: vanilla Langevin Dynamics. Methods are used with the same backbone architecture. Generated samples are plotted through the iterations ever 2 000 steps.

Results. We observe Figure 1 the outputs using both methods on the toy dataset. While they achieve a great representation of the truth after a large number of iterations, we notice that STANLEY learns an energy that closely approximates the true density during the first thousands of iterations if the training process. The sharpness of the data generated by STANLEY in the first iterations shows an empirically better ability to sample from the 2D toy dataset.

Image Generation

Datasets. We run our method and several baselines detailed below on the *CIFAR-10* dataset (Krizhevsky and Hinton 2009) and the *Oxford Flowers 102* dataset (Nilsback and Zisserman 2008). *CIFAR-10* is a popular computer-vision dataset of $50\,000$ training images and $10\,000$ test images, of size 32×32. It is composed of tiny natural images representing a wide variety of objects and scenes, making the task of self supervision supposedly harder. The *Oxford Flowers 102* dataset is composed of 102 flower categories.

Methods and Settings for the Flowers dataset. Nonconvergent MCMC are also used in this experiment and the number of MCMC transitions is set to $K = 50$. Global learning parameters of the gradient descent update is set to 0.001 for both methods. We run each method during $T = 100\,000$ iterations and plot the results using the final vector of fitted parameters.

Methods and Settings for CIFAR-10. We employ the same nonconvergent MCMC strategies for this experiment. The value of the threshold th for our STANLEY method is set to $\mathsf{th} = 0.0002$. The total number of EBM iterations is set to $T = 100\,000$. The global learning rate η is set to a constant equal to 0.0001. In this experiment, we slightly change the last step of our method described in Algorithm 1. Indeed, Line 10 in Algorithm 1 is not a plain Stochastic Gradient Descent here but we rather use the ADAM optimizer (Kingma and Ba 2015). The scaling factor of the Brownian motion is equal to 0.01.

Network architectures for both. The backbone of the energy function for this experiment is a vanilla ConvNet composed of 3×3 convolution layers with stride 1. 5 Convolutional Layers using ReLU activation functions are stacked.

Results. *(Flowers)* Visual results are provided in Figure 2 where we have used both methods to generate synthetic images of flowers. For each threshold iterations number ($5\,000$ iterations) we sample $10\,000$ synthetic images from the EBM model under the current vector of parameters and use the same number of data observations to compute the FID similarity score as advocated in Heusel et al. (2017). The evolution of the FID values are reported in Figure 4 (Left) through the iterations. We note that our method outperforms the other baselines for all iterations threshold, including the Vanilla Langevin (in blue) which is an ablated form our STANLEY (no adaptive stepsize).

Figure 2: (Flowers Dataset). Left: Langevin Method. Right: STANLEY method. After 100k iterations.

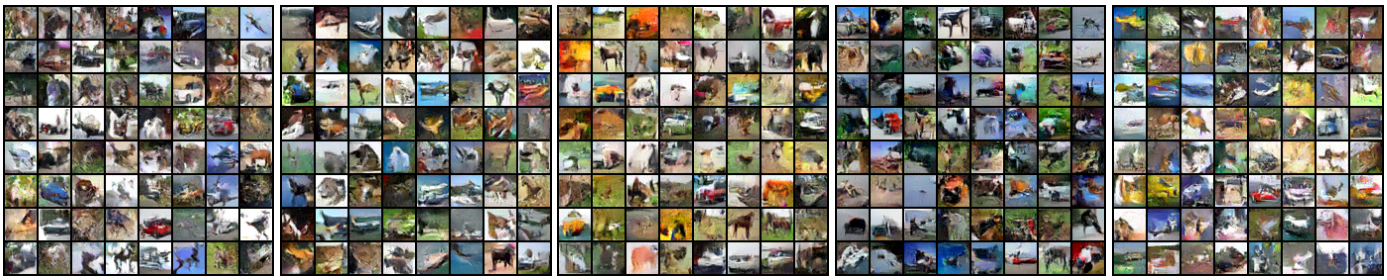

Figure 3: (CIFAR Dataset). 1: Langevin 2: STANLEY 3: MH 4: HMC 5: GD without noise. After 100k iterations.

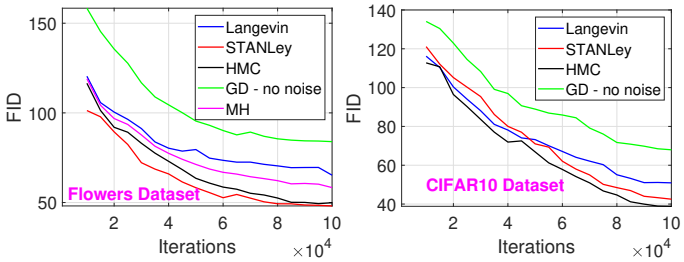

Figure 4: (FID values per method against 100k iterations elapsed). Left: Oxford Flowers dataset. Right: CIFAR-10.

(CIFAR-10) Visual results are provided in Figure 3 where we have used both methods to generate synthetic images of flowers. The FID values are reported in Figure 4 (Right) and have been computed using $10\,000$ synthetic images from each model. The similarity score is then evaluated every $5\,000$ iterations. While the FID curves for the Flowers dataset exhibits a superior performance of our method throughout the training procedure, we notice that in the case of CIFAR-10, vanilla method seems to be slightly better than STANLEY during the first iterations, *i.e.,* when the model is still learning the representation of the images. Yet, after a certain number of iterations, we observe that STANLEY leads to more accurate synthetic images. This behavior can be explained by the importance of incorporating curvature informed metrics into the training process when the parameter reaches a neighborhood of the optimal solution.

Image Inpainting

Datasets. We use the CelebA dataset (Liu et al. 2015) to evaluate our learning algorithm. CelebA dataset contains more than 200k RGB color facial image. We use 100k images for training and 100 images for testing.

Methods and Settings. Nonconvergent MCMC are also used in this experiment and the number of MCMC transitions is set to $K = 50$. Global learning parameters of the gradient descent update is set to 0.01 for all methods. We run each method during $T = 50\,000$ iterations and plot the results using the final vector of fitted parameters.

Results. Figure 5 displays the FID curves for all methods. We note that along the iterations, our method STANLEY outperforms the other baseline and is similar to HMC, while only requiring first order information for the computation of the stepsize whereas HMC computes second order quantity.

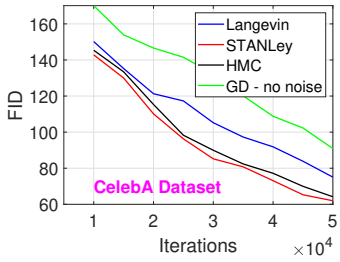

Figure 5: (FID values per method against 50k iterations elapsed). CelebA (Liu et al. 2015) dataset.

Figure 6: (Image inpainting). CelebA dataset. **Top row:** STANLEY **Bottom row:** Vanilla Langevin.

Figure 6 shows the visual check on different samples between our method and its ablated form, *i.e.,* the vanilla Langevin sampler based EBM.

6 Conclusion

Given the growing interest in self-supervised learning, we propose in this paper, an improvement of the so-called MCMC based Energy-Based models. In the particular case of a highly nonlinear structural model of the EBM, more precisely a Convolutional Neural Network in our paper, we tackle the complex task of sampling negative samples from the energy function. The multi-modal and highly curved landscape one must sample from inspire our technique called STANLEY, and based on a Stochastic Gradient Anisotropic Langevin Dynamics, that updates the Markov Chain using an anisotropic stepsize in the vanilla Langevin update. We provide strong theoretical guarantees for our novel method, including uniform ergodicity and geometric convergence rate of the transition kernels to the stationary distribution of the chain. Our method is tested on several benchmarks data and image generation tasks including toy and real datasets such as CIFAR-10, Flowers and CelebA.

References

Allassonniere, S.; and Kuhn, E. 2015. Convergent Stochastic Expectation Maximization algorithm with efficient sampling in high dimension. Application to deformable template model estimation. *CSDA*, 91: 4–19.

Atchadé, Y. F. 2006. An adaptive version for the Metropolis adjusted Langevin algorithm with a truncated drift. *Methodology and Computing in applied Probability*, 8(2): 235–254.

Bottou, L.; and Bousquet, O. 2008. The Tradeoffs of Large Scale Learning. In Platt, J. C.; Koller, D.; Singer, Y.; and Roweis, S. T., eds., *Advances in Neural Information Processing Systems 20*, 161–168. Curran Associates, Inc.

Brosse, N.; Durmus, A.; Moulines, É.; and Sabanis, S. 2017. The Tamed Unadjusted Langevin Algorithm. *arXiv preprint arXiv:1710.05559*.

Cotter, S. L.; Roberts, G. O.; Stuart, A. M.; and White, D. 2013. MCMC methods for functions: modifying old algorithms to make them faster. *Statistical Science*, 424–446.

De Freitas, N.; Højen-Sørensen, P.; Jordan, M. I.; and Russell, S. 2001. Variational MCMC. *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence*.

Deng, Y.; Bakhtin, A.; Ott, M.; Szlam, A.; and Ranzato, M. 2020. Residual energy-based models for text generation. *arXiv preprint arXiv:2004.11714*.

Doucet, A.; Godsill, S.; and Andrieu, C. 2000. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and computing*, 10(3): 197–208.

Du, Y.; Li, S.; Tenenbaum, J.; and Mordatch, I. 2020. Improved Contrastive Divergence Training of Energy Based Models. *arXiv preprint arXiv:2012.01316*.

Du, Y.; Meier, J.; Ma, J.; Fergus, R.; and Rives, A. 2019. Energy-based models for atomic-resolution protein conformations. In *ICLR*.

Du, Y.; and Mordatch, I. 2019. Implicit Generation and Modeling with Energy Based Models. In Wallach, H.; Larochelle, H.; Beygelzimer, A.; d Alche-Buc, F.; Fox, E.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Durmus, A.; Roberts, G. O.; Vilmart, G.; and Zygalakis, K. C. 2017. Fast Langevin based algorithm for MCMC in high dimensions. *Ann. Appl. Probab.*, 27(4): 2195–2237.

Gao, R.; Lu, Y.; Zhou, J.; Zhu, S.-C.; and Wu, Y. N. 2018. Learning generative convnets via multi-grid modeling and sampling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 9155–9164.

Gao, R.; Nijkamp, E.; Kingma, D. P.; Xu, Z.; Dai, A. M.; and Wu, Y. N. 2020. Flow contrastive estimation of energy-based models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7518–7528.

Girolami, M.; and Calderhead, B. 2011. Riemann manifold langevin and hamiltonian monte carlo methods. *Journal of the Royal Statistical Society: Series B*, 73(2): 123–214.

Goodfellow, I. J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*.

Grenander, U.; and Miller, M. I. 1994. Representations of knowledge in complex systems. *Journal of the Royal Statistical Society: Series B (Methodological)*, 56(4): 549–581.

Gustafsson, F. K.; Danelljan, M.; Bhat, G.; and Schön, T. B. 2020. Energy-based models for deep probabilistic regression. In *ECCV*, 325–343. Springer.

Gutmann, M.; and Hyvärinen, A. 2010. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*.

Haarnoja, T.; Tang, H.; Abbeel, P.; and Levine, S. 2017. Reinforcement learning with deep energy-based policies. In *International Conference on Machine Learning*, 1352–1361.

Han, T.; Nijkamp, E.; Fang, X.; Hill, M.; Zhu, S.-C.; and Wu, Y. N. 2019. Divergence triangle for joint training of generator model, energy-based model, and inferential model. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 8670–8679.

Han, T.; Nijkamp, E.; Zhou, L.; Pang, B.; Zhu, S.-C.; and Wu, Y. N. 2020. Joint training of variational auto-encoder and latent energy-based model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7978–7987.

Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; and Hochreiter, S. 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *arXiv preprint arXiv:1706.08500*.

Hinton, G. E. 2002. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8).

Ingraham, J.; Riesselman, A.; Sander, C.; and Marks, D. 2018. Learning protein structure with a differentiable simulator. In *ICLR*.

Jacob, P. E.; O Leary, J.; and Atchadé, Y. F. 2020. Unbiased Markov chain Monte Carlo methods with couplings. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 82(3): 543–600.

Jarner, S. F.; and Hansen, E. 2000. Geometric ergodicity of Metropolis algorithms. *Stochastic processes and their applications*, 85(2): 341–361.

Jin, L.; Lazarow, J.; and Tu, Z. 2017. Introspective classification with convolutional nets. In *Advances in Neural Information Processing Systems*, 823–833.

Kingma, D. P.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. *ICLR*.

Kingma, D. P.; and Welling, M. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Krizhevsky, A.; and Hinton, G. 2009. Learning multiple layers of features from tiny images. *Master's thesis, Department of Computer Science, University of Toronto*.

Lazarow, J.; Jin, L.; and Tu, Z. 2017. Introspective neural networks for generative modeling. In *ICCV*, 2774–2783.

LeCun, Y.; Chopra, S.; Hadsell, R.; Ranzato, M.; and Huang, F. 2006. A tutorial on energy-based learning. *Predicting structured data*, 1(0).

Lee, K.; Xu, W.; Fan, F.; and Tu, Z. 2018. Wasserstein introspective neural networks. In *ICCV*, 3702–3711.

Liu, W.; Wang, X.; Owens, J. D.; and Li, Y. 2020. Energy-based Out-of-distribution Detection. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*.

Liu, Z.; Luo, P.; Wang, X.; and Tang, X. 2015. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, 3730–3738.

Marshall, T.; and Roberts, G. 2012. An adaptive approach to Langevin MCMC. *Statistics and Computing*, 22(5).

Meyn, S. P.; and Tweedie, R. L. 2012. *Markov chains and stochastic stability*. Springer Science & Business Media.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. *arXiv preprint arXiv:1310.4546*.

Neal, R. M.; et al. 2011. MCMC using Hamiltonian dynamics. *Handbook of markov chain monte carlo*, 2(11): 2.

Ngiam, J.; Chen, Z.; Koh, P. W.; and Ng, A. Y. 2011. Learning deep energy models. In *Proceedings of the 28th international conference on machine learning (ICML-11)*.

Nijkamp, E.; Hill, M.; Han, T.; Zhu, S.; and Wu, Y. N. 2020. On the Anatomy of MCMC-Based Maximum Likelihood Learning of Energy-Based Models. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence*.

Nijkamp, E.; Hill, M.; Zhu, S.-C.; and Wu, Y. N. 2019. Learning non-convergent non-persistent short-run MCMC toward energy-based model. *arXiv preprint arXiv:1904.09770*.

Nilsback, M.-E.; and Zisserman, A. 2008. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, 722–729. IEEE.

Qiu, Y.; Zhang, L.; and Wang, X. 2019. Unbiased contrastive divergence algorithm for training energy-based latent variable models. In *ICLR*.

Robbins, H.; and Monro, S. 1951. A stochastic approximation method. *Annals of Mathematical Statistics*, 22.

Robert, C. P.; and Casella, G. 2010. *Metropolis–Hastings Algorithms*, 167–197. New York, NY: Springer New York. ISBN 978-1-4419-1576-4.

Roberts, G. O.; and Rosenthal, J. S. 1997. Optimal scaling of discrete approximations to Langevin diffusions. *J. R. Statist. Soc. B*, 60: 255–268.

Roberts, G. O.; Rosenthal, J. S.; et al. 2004. General state space Markov chains and MCMC algorithms. *Probability surveys*, 1.

Roberts, G. O.; and Tweedie, R. L. 1996. Exponential convergence of Langevin distributions and their discrete approximations. *Bernoulli*, 2(4): 341–363.

Rue, H.; Martino, S.; and Chopin, N. 2009. Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. *Journal of the royal statistical society: Series b (statistical methodology)*.

Song, Y.; Garg, S.; Shi, J.; and Ermon, S. 2020. Sliced score matching: A scalable approach to density and score estimation. In *Uncertainty in Artificial Intelligence*, 574–584.

Tieleman, T. 2008. Training restricted Boltzmann machines using approximations to the likelihood gradient. In *ICML*.

Wainwright, M. J.; and Jordan, M. I. 2008. Graphical Models, Exponential Families, and Variational Inference. *Found. Trends Mach. Learn.*, 1(1-2): 1–305.

Welling, M.; and Hinton, G. E. 2002. A new learning algorithm for mean field Boltzmann machines. In *International Conference on Artificial Neural Networks*, 351–357.

Welling, M.; and Teh, Y. W. 2011. Bayesian learning via stochastic gradient Langevin dynamics. In *ICML*, 681–688.

Wenliang, L.; Sutherland, D.; Strathmann, H.; and Gretton, A. 2019. Learning deep kernels for exponential family densities. In *International Conference on Machine Learning*.

Wolfinger, R. 1993. Laplace's approximation for nonlinear mixed models. *Biometrika*, 80(4): 791–795.

Xie, J.; Lu, Y.; Zhu, S.-C.; and Wu, Y. 2016. A theory of generative convnet. In *International Conference on Machine Learning*, 2635–2644. PMLR.

Xie, J.; Zheng, Z.; Fang, X.; Zhu, S.-C.; and Wu, Y. N. 2021a. Cooperative Training of Fast Thinking Initializer and Slow Thinking Solver for Conditional Learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Xie, J.; Zheng, Z.; Fang, X.; Zhu, S.-C.; and Wu, Y. N. 2021b. Learning cycle-consistent cooperative networks via alternating MCMC teaching for unsupervised cross-domain translation. In *AAAI*.

Xie, J.; Zheng, Z.; Gao, R.; Wang, W.; Zhu, S.-C.; and Nian Wu, Y. 2018. Learning descriptor networks for 3D shape synthesis and analysis. In *CVPR*, 8629–8638.

Xie, J.; Zheng, Z.; Gao, R.; Wang, W.; Zhu, S.-C.; and Wu, Y. N. 2020. Generative VoxelNet: Learning Energy-Based Models for 3D Shape Synthesis and Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Xie, J.; Zheng, Z.; and Li, P. 2021. Learning Energy-Based Model with Variational Auto-Encoder as Amortized Sampler. In *AAAI*.

Xie, J.; Zhu, S.-C.; and Wu, Y. N. 2017. Synthesizing dynamic patterns by spatial-temporal generative ConvNet. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 7093–7101.

Xie, J.; Zhu, S.-C.; and Wu, Y. N. 2019. Learning energy-based spatial-temporal generative convnets for dynamic patterns. *IEEE TPAMI*.

Xu, Y.; Xie, J.; Zhao, T.; Baker, C.; Zhao, Y.; and Wu, Y. N. 2019. Energy-based continuous inverse optimal control. *arXiv preprint arXiv:1904.05453*.

Yu, L.; Song, Y.; Song, J.; and Ermon, S. 2020. Training deep energy-based models with f-divergence minimization. In *International Conference on Machine Learning*.

Zhu, S. C.; Wu, Y.; and Mumford, D. 1998. Filters, random fields and maximum entropy (FRAME): Towards a unified theory for texture modeling. *International Journal of Computer Vision*, 27(2): 107–126.