EFFICIENT SAMPLING FOR GENERATIVE ADVERSARIAL NETWORKS WITH COUPLING MARKOV CHAINS

Anonymous authors
Paper under double-blind review

ABSTRACT

Recently, sampling methods have been successfully applied to enhance the sample quality of Generative Adversarial Networks (GANs). However, in practice, they typically have poor sample efficiency because of the independent proposal sampling from the generator. In this work, we propose CMC-GAN, a novel sampling method with dependent proposals reparametrized through a Coupling Markov Chain (CMC) in the latent space of the generator. Theoretically, we prove that our CMC proposal can converge to the true data distribution, given a perfect discriminator. Empirically, extensive experiments on synthetic and real datasets demonstrate that our CMC-GAN largely improves the sample efficiency and obtains better sample quality simultaneously.

1 INTRODUCTION

Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) have achieved a great success on generating realistic images in recent years (Karras et al., 2019; Brock et al., 2019). Unlike previous models that explicitly parameterize the data distribution, GANs rely on an alternative optimization between a generator and a discriminator to learn the data distribution implicitly. However, in practice, samples generated by GANs still suffer from problems such as mode collapse and bad artifacts.

Recently, sampling methods have shown promising results on enhancing the sample quality of GANs by making use of the information in the discriminator. Conventionally, after GAN training, the discriminator is discarded, and only the generator is left for image generation. While the sampling methods utilize the density ratio information (implied by the discriminator) to close the gap between the generative distribution and the data distribution. For example, DRS (Azadi et al., 2019) applies rejection sampling, and MH-GAN (Turner et al., 2019) adopts Markov chain Monte Carlo (MCMC) sampling for the improved sample quality of GANs. Nevertheless, these methods still suffer a lot from the sample efficiency problem. For example, as will be shown in Section 5, MH-GAN's average acceptance ratio on CIFAR10 can be lower than 5%, which makes the Markov chains slow to mix. As MH-GAN adopts an *independent* proposal q, i.e., $q(\mathbf{x}'|\mathbf{x}) = q(\mathbf{x}')$, the difference between samples can be so large that the proposal gets rejected easily.

To address this limitation, we propose to generalize the independent proposal to a general *dependent* proposal $q(\mathbf{x}'|\mathbf{x})$. To the end, the proposed sample can be a refinement of the previous one, which leads to a higher acceptance ratio and better sample quality. We can also balance between the exploration and exploitation of the Markov chains by tuning the step size. However, it is hard to design a proper dependent proposal in the high dimensional sample space $\mathcal{X}$ because the energy landscape could be very complex (Neal et al., 2010).

Nevertheless, we notice that the generative distribution $p_g(\mathbf{x})$ of GANs is implicitly defined as the push-forward of the latent prior distribution $p_0(\mathbf{z})$, and designing proposals in the low dimensional latent space is generally much easier. Hence, GAN's latent variable structure motivates us to design a *structured dependent proposal* with two Coupling Markov Chains (CMC), one in the sample space $\mathcal{X}$ and the other in the latent space $\mathcal{Z}$. As shown in Figure 1, given the current coupling samples $(\mathbf{z}_k, \mathbf{x}_k)$, we draw the next proposal $\mathbf{x}'$ in a bottom-to-up way: 1) drawing a latent proposal $\mathbf{z}'$ following $q(\mathbf{z}'|\mathbf{z}_k)$; 2) pushing it forward through the generator and getting the sample proposal $\mathbf{x}' = G(\mathbf{z}')$; 3) assigning $\mathbf{x}_{k+1} = \mathbf{x}'$ if the proposal $\mathbf{x}'$ is accepted, and $\mathbf{x}_{k+1} = \mathbf{x}_k$ otherwise. By utilizing the underlying structure of GANs, the proposed CMC sampling becomes more efficient in the low-dimensional latent space. We summarize our main contributions as follows:

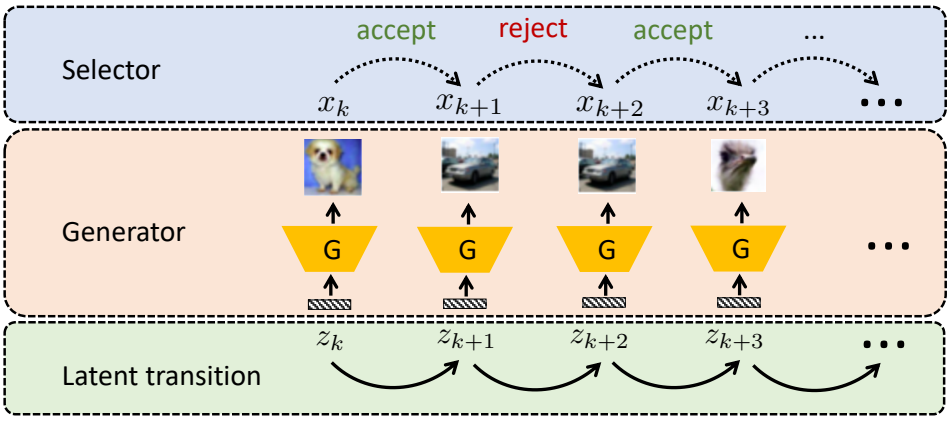

Figure 1: Illustration of CMC-GAN's structured dependent proposal with two coupling Markov chains, one in the latent space $\mathcal{Z}$, and the other in the sample space $\mathcal{X}$.

- We propose a structured dependent proposal of GANs, which reparameterizes the sample-level transition $\mathbf{x} \to \mathbf{x}'$ into the latent-level $\mathbf{z} \to \mathbf{z}'$ with two coupling Markov chains. We prove that it converges to the true data distribution with a tractable acceptance criterion.
- Our proposed method, called CMC-GAN, serves as a unified framework for the existing sampling methods of GANs. It provides a better balance between exploration and exploitation by the structured dependent proposal, and also guarantees the convergence of the Markov chain by the acceptance-rejection step.
- Empirical results demonstrate that CMC-GAN achieves better image quality and much higher sample efficiency than the state-of-the-art methods on both synthetic and real datasets.

2 Related Work

Although GANs are able to synthesize high-quality images, the minimax nature of GANs makes it quite unstable and leads to mode collapse. A series of works like DCGAN (Radford et al., 2015) and SNGAN (Miyato et al., 2018) improve the sample quality by designing new architectures. Methods like Unrolled GAN (Metz et al., 2017) and PACGAN (Lin et al., 2018), try to fix mode collapse with new training mechanisms. Other methods like WGAN (Arjovsky et al., 2017) and WGAN-GP (Gulrajani et al., 2017) instead propose alternative objectives for better training stability.

Moreover, there is another line of work using sampling methods to improve the sample quality of GANs. DRS (Azadi et al., 2019) firstly proposes to use rejection sampling. MH-GAN (Turner et al., 2019) instead uses the Metropolis-Hasting (MH) algorithm with an independent proposal. DDLS (Che et al., 2020) and DCD (Song et al., 2020) apply gradient-based proposals by viewing GAN as an energy-based model. Tanaka (2019) proposes a similar gradient-based method named DOT from the perspective of optimal transport.

Table 1: Comparison of sampling methods for GANs in terms of three effective sampling mechanisms.

Method	**Rejection step**	**Markov chain**	**Latent gradient proposal**
GAN	✗	✗	✗
DRS (Azadi et al., 2019)	✓	✗	✗
MH-GAN (Turner et al., 2019)	✓	✓	✗
DDLS (Che et al., 2020)	✗	✓	✓
CMC-GAN (ours)	✓	✓	✓

Different from them, our CMC-GAN introduces a structured dependent proposal with two coupling Markov chains, and includes all three effective sampling mechanisms, the Markov chain Monte Carlo

method, the acceptance-rejection step, and the latent gradient-based proposal, to further improve the sample efficiency. As shown in Table 1, many existing works are special cases of our CMC-GAN.

3 BACKGROUND

In this section, we briefly introduce the notations and backgrounds.

3.1 GAN

GAN models the data distribution $p_d(\mathbf{x})$ implicitly with a generator $G : \mathcal{Z} \to \mathcal{X}$ mapping from a low-dimensional latent space $\mathcal{Z}$ to a high-dimensional sample space $\mathcal{X}$,

$$\mathbf{x} = G(\mathbf{z}), \quad \mathbf{z} \sim p_0(\mathbf{z}), \tag{1}$$

where the sample $\mathbf{x}$ follows the generative distribution $p_g(\mathbf{x})$ and the latent variable $\mathbf{z}$ follows the prior distribution $p_0(\mathbf{z})$, e.g., a standard normal distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$. In GAN, a discriminator $D : \mathcal{X} \to [0, 1]$ is learned to distinguish samples from $p_d(\mathbf{x})$ and $p_g(\mathbf{x})$ in an adversarial way

$$\min_G \max_D \mathbb{E}_{\mathbf{x}\sim p_d(\mathbf{x})} \log(D(\mathbf{x})) + \mathbb{E}_{\mathbf{z}\sim p_0(\mathbf{z})} \log(1 - D(G(\mathbf{z}))). \tag{2}$$

Goodfellow et al. (2014) point out that an optimal discriminator D implies the density ratio between the data and generative distributions

$$D(\mathbf{x}) = \frac{p_d(\mathbf{x})}{p_d(\mathbf{x}) + p_g(\mathbf{x})} \quad \Rightarrow \quad \frac{p_d(\mathbf{x})}{p_g(\mathbf{x})} = \frac{1}{D(\mathbf{x})^{-1} - 1}. \tag{3}$$

3.2 MCMC

Markov chain Monte Carlo (MCMC) refers to a kind of sampling methods that draws a chain of samples $\mathbf{x}_{1:K} \in \mathcal{X}^K$ from a target distribution $p_t(\mathbf{x})$. We denote the initial distribution as $p_0(\mathbf{x})$ and the proposal distribution as $q(\mathbf{x}'|\mathbf{x}_k)$. With the Metropolis-Hastings (MH) algorithm, we accept the proposal $\mathbf{x}' \sim q(\mathbf{x}'|\mathbf{x}_k)$ with probability

$$\alpha\left(\mathbf{x}', \mathbf{x}_k\right) = \min\left(1, \frac{p_t\left(\mathbf{x}'\right) q\left(\mathbf{x}_k|\mathbf{x}'\right)}{p_t\left(\mathbf{x}_k\right) q\left(\mathbf{x}'|\mathbf{x}_k\right)}\right) \in [0, 1]. \tag{4}$$

If $\mathbf{x}'$ is accepted, $\mathbf{x}_{k+1} = \mathbf{x}'$, otherwise $\mathbf{x}_{k+1} = \mathbf{x}_k$. Under mild assumptions, the Markov chain is guaranteed to converge to $p_t(\mathbf{x})$ as $K \to \infty$. In practice, the sample efficiency of MCMC crucially depends on the proposal distribution to trade off between exploration and exploitation.

4 THE PROPOSED CMC-GAN

In this section, we first review MH-GAN and point out the limitations. We then propose our structured dependent proposal to overcome these obstacles, and finally discuss its theoretical properties as well as practical implementation.

4.1 FROM INDEPENDENT PROPOSAL TO DEPENDENT PROPOSAL

MH-GAN (Turner et al., 2019) first proposes to improve GAN sampling with MCMC. Specifically, given a perfect discriminator D and a descent (but imperfect) generator G after training, they take the data distribution $p_d(\mathbf{x})$ as the target distribution and use the generator distribution $p_g(\mathbf{x})$ as an independent proposal

$$\mathbf{x}' \sim q\left(\mathbf{x}'|\mathbf{x}_k\right) = q\left(\mathbf{x}'\right) = p_g(\mathbf{x}'). \tag{5}$$

With the MH criterion (Eqn. (4)) and the density ratio (Eqn. (3)), we should accept $\mathbf{x}'$ with probability

$$\alpha_{\mathrm{MH}}\left(\mathbf{x}', \mathbf{x}_k\right) = \min\left(1, \frac{p_d\left(\mathbf{x}'\right) q\left(\mathbf{x}_k\right)}{p_d\left(\mathbf{x}_k\right) q\left(\mathbf{x}'\right)}\right) = \min\left(1, \frac{D\left(\mathbf{x}_k\right)^{-1} - 1}{D\left(\mathbf{x}'\right)^{-1} - 1}\right). \tag{6}$$

As a result, the Markov chain is guaranteed to converge to the true data distribution $p_d(\mathbf{x})$. However, to achieve tractability, MH-GAN adopts an independent proposal $q(\mathbf{x}')$ with poor sample efficiency. As the proposed sample $\mathbf{x}'$ is independent of the current sample $\mathbf{x}_k$, the difference between the two samples can be so large that it results in a very low acceptance probability. Consequently, samples can be trapped in the same place for a long time, leading to a very slow mixing of the chain.

A natural solution is to take a *dependent* proposal $q(\mathbf{x}'|\mathbf{x}_k)$ that will propose a sample $\mathbf{x}'$ close to the current one $\mathbf{x}_k$, which is more likely to be accepted. Nevertheless, the problem of such a dependent proposal is that its MH acceptance criterion

$$\alpha_{\mathrm{DEP}}\left(\mathbf{x}', \mathbf{x}_k\right) = \min\left(1, \frac{p_d\left(\mathbf{x}'\right) q\left(\mathbf{x}_k|\mathbf{x}'\right)}{p_d\left(\mathbf{x}_k\right) q\left(\mathbf{x}'|\mathbf{x}_k\right)}\right), \tag{7}$$

is generally intractable because the data density $p_d(\mathbf{x})$ is unknown. Besides, it is hard to design a proper dependent proposal $q(\mathbf{x}'|\mathbf{x}_k)$ in the high dimensional sample space $\mathcal{X}$ with complex landscape. These obstacles prevent us from adopting a dependent proposal that is more suitable for MCMC.

4.2 A Tractable Structured Dependent Proposal with Coupling Markov Chains

As discussed above, the major difficulty of a general dependent proposal $q(\mathbf{x}'|\mathbf{x}_k)$ is to compute the MH criterion. We show that it can be made tractable by considering an additional coupling Markov chain in the latent space.

As we know, samples of GANs lie in a low-dimensional manifold induced by the push-forward of the latent variable. Suppose that at the k-th step of the Markov chain, we have a GAN sample $\mathbf{x}_k$ with latent $\mathbf{z}_k$. Instead of drawing a sample $\mathbf{x}'$ directly from a sample-level proposal distribution $q(\mathbf{x}'|\mathbf{x}_k)$, we first draw a latent proposal $\mathbf{z}'$ from a dependent latent proposal distribution $q(\mathbf{z}'|\mathbf{z}_k)$. Afterward, we push the latent $\mathbf{z}'$ forward through the generator and get the output $\mathbf{x}'$ as our sample proposal.

As illustrated in Figure 1, our bottom-to-up proposal relies on the interplay between two coupling Markov chains in the sample space $\mathcal{X}$ and the latent space $\mathcal{Z}$. Hence we call it a CMC (Coupling Markov Chain) proposal. Through a learned generator, we transport the transition $\mathbf{x}_k \to \mathbf{x}'$ in the high dimensional space $\mathcal{X}$ into the low dimensional space $\mathcal{Z}$, $\mathbf{z}_k \to \mathbf{z}'$, which enjoys a much better landscape and makes it easier to design proposals in MCMC algorithms. For example, the latent target distribution is nearly standard normal when the generator is nearly perfect. In fact, under mild conditions, the CMC proposal distribution $q_{\mathrm{CMC}}(\mathbf{x}'|\mathbf{x}_k)$ and the latent proposal distribution $q(\mathbf{z}'|\mathbf{z}_k)$ are tied with the following change of variables (Gemici et al., 2016; Ben-Israel, 1999)

$$\log q_{\mathrm{CMC}}(\mathbf{x}'|\mathbf{x}_k) = \log q(\mathbf{x}'|\mathbf{z}_k) = \log q(\mathbf{z}'|\mathbf{z}_k) - \frac{1}{2}\log\det J_{\mathbf{z}'}^{\top} J_{\mathbf{z}'}, \tag{8}$$

where $J_{\mathbf{z}}$ denotes the Jacobian matrix of the push-forward G at $\mathbf{z}$, i.e., $[J_{\mathbf{z}}]_{ij} = \partial\,\mathbf{x}_i / \partial\,\mathbf{z}_j, \mathbf{x} = G(\mathbf{z})$.

Nevertheless, it remains unclear whether we can perform the MH test to decide the acceptance of the proposal $\mathbf{x}'$. Note that a general dependent proposal distribution does not meet a tractable MH acceptance criterion (Eqn. (7)). Perhaps surprisingly, it can be shown that with our structured CMC proposal, the MH acceptance criterion is tractable for general latent proposals $q(\mathbf{z}'|\mathbf{z}_k)$.

Theorem 1. *Consider a Markov chain of GAN samples $\mathbf{x}_{1:K}$ with initial distribution $p_g(\mathbf{x})$. For step $k+1$, we accept our CMC proposal $\mathbf{x}' \sim q_{\mathrm{CMC}}(\mathbf{x}'|\mathbf{x}_k)$ with probability*

$$\alpha_{\mathrm{CMC}}\left(\mathbf{x}', \mathbf{x}_k\right) = \min\left(1,\ \frac{p_0(\mathbf{z}')q(\mathbf{z}_k|\mathbf{z}')}{p_0(\mathbf{z}_k)q(\mathbf{z}'|\mathbf{z}_k)} \cdot \frac{D(\mathbf{x}_k)^{-1} - 1}{D(\mathbf{x}')^{-1} - 1}\right), \tag{9}$$

i.e. let $\mathbf{x}_{k+1} = \mathbf{x}'$ if $\mathbf{x}'$ is accepted and $\mathbf{x}_{k+1} = \mathbf{x}_k$ otherwise. Further assume the chain is irreducible and aperiodic. Then, according to the Metropolis-Hastings algorithm, the stationary distribution of this Markov chain is the data distribution $p_d(\mathbf{x})$.

Proof. Note that similar to Eqn (8), we also have the change of variables between $p_g(\mathbf{x})$ and $p_0(\mathbf{z})$,

$$\log p_g(\mathbf{x})|_{\mathbf{x}=G(\mathbf{z})} = \log p_0(\mathbf{z}) - \frac{1}{2}\log\det J_{\mathbf{z}}^{\top} J_{\mathbf{z}}. \tag{10}$$

We only need to prove that Eqn. (9) holds. Together with Eqn. (3), (7) and (8), we have

$$\begin{aligned}\alpha_{\mathrm{CMC}}(\mathbf{x}', \mathbf{x}_k) &= \frac{p_d(\mathbf{x}')\, q(\mathbf{x}_k|\mathbf{x}')}{p_d(\mathbf{x}_k)\, q(\mathbf{x}'|\mathbf{x}_k)} = \frac{p_d(\mathbf{x}') q(\mathbf{z}_k|\mathbf{z}') \left(\det J_{\mathbf{z}_k}^\top J_{\mathbf{z}_k}\right)^{-\frac{1}{2}} p_g(\mathbf{x}_k) p_g(\mathbf{x}')}{p_d(\mathbf{x}_k) q(\mathbf{z}'|\mathbf{z}_k) \left(\det J_{\mathbf{z}'}^\top J_{\mathbf{z}'}\right)^{-\frac{1}{2}} p_g(\mathbf{x}') p_g(\mathbf{x}_k)} \\ &= \frac{q(\mathbf{z}_k|\mathbf{z}')\left(\det J_{\mathbf{z}_k}^\top J_{\mathbf{z}_k}\right)^{-\frac{1}{2}} p_0(\mathbf{z}')\left(\det J_{\mathbf{z}'}^\top J_{\mathbf{z}'}\right)^{-\frac{1}{2}} (D(\mathbf{x}_k)^{-1} - 1)}{q(\mathbf{z}'|\mathbf{z}_k)\left(\det J_{\mathbf{z}'}^\top J_{\mathbf{z}'}\right)^{-\frac{1}{2}} p_0(\mathbf{z}_k)\left(\det J_{\mathbf{z}_k}^\top J_{\mathbf{z}_k}\right)^{-\frac{1}{2}} (D(\mathbf{x}')^{-1} - 1)} \\ &= \frac{p_0(\mathbf{z}') q(\mathbf{z}_k|\mathbf{z}')(D(\mathbf{x}_k)^{-1} - 1)}{p_0(\mathbf{z}_k) q(\mathbf{z}'|\mathbf{z}_k)(D(\mathbf{x}')^{-1} - 1)}.\end{aligned} \tag{11}$$

Hence the proof is completed. □

The theorem above demonstrates the following favorable properties of our method:

- The discriminator score ratio is the same as $\alpha_{\mathrm{MH}}(\mathbf{x}', \mathbf{x}_k)$, but MH-GAN is restricted to a specific independent proposal. Our method instead works for any latent proposal $q(\mathbf{z}'|\mathbf{z}_k)$. When we take $q(\mathbf{z}'|\mathbf{z}_k) = p_0(\mathbf{z}')$, our method reduces to MH-GAN.
- Compared to $\alpha_{\mathrm{DEP}}(\mathbf{x}', \mathbf{x}_k)$ of a general dependent proposal (Eqn. (7)), the unknown data distributions terms are successfully cancelled in the reparameterized acceptance criterion.
- The reparameterized MH acceptance criterion becomes tractable as it only involves the latent priors, the latent proposal distributions, and the discriminator scores.

Combining the CMC proposal $q_{\mathrm{CMC}}(\mathbf{x}'|\mathbf{x}_k)$ and its tractable MH criterion $\alpha_{\mathrm{CMC}}(\mathbf{x}', \mathbf{x}_k)$, we have developed a novel sampling method for GANs, coined as CMC-GAN. See Appendix 1 for a detailed description.

4.3 A Practical Implementation

CMC-GAN enables us to utilize the vast literature of existing MCMC algorithms (Neal et al., 2010) to design dependent proposals for GANs. We take Langevin Monte Carlo (LMC) as an example. As an Euler-Maruyama discretization of the Langevin dynamics, LMC updates the Markov chain with

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \frac{\tau}{2}\nabla_{\mathbf{x}} \log p_t(\mathbf{x}_k) + \sqrt{\tau} \cdot \boldsymbol{\varepsilon}, \;\; \boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{12}$$

for a target distribution $p_t(\mathbf{x})$. Compared to MH-GAN, LMC utilizes the gradient information to explore the energy landscape more efficiently. However, if we directly take the (unknown) data distribution $p_d(\mathbf{x})$ as the target distribution $p_t(\mathbf{x})$, LMC does not meet a tractable update rule.

As discussed above, the reparameterization of CMC-GAN makes it easier to design transitions in the low-dimensional latent space. Hence, we instead propose to use LMC for the latent Markov chain. We assume that the data distribution also lies in the low-dimensional manifold induced by the generator, i.e., $\mathrm{Supp}\,(p_d) \subset \mathrm{Im}(G)$. This implies that the data distribution $p_d(\mathbf{x})$ also has a coupling distribution in the latent space, denoted as $p_t(\mathbf{z})$. They are tied with the change of variables

$$\log p_d(\mathbf{x})|_{\mathbf{x}=G(\mathbf{z})} = \log p_t(\mathbf{z}) - \frac{1}{2}\log\det\, J_{\mathbf{z}}^\top J_{\mathbf{z}}. \tag{13}$$

Taking $p_t(\mathbf{z})$ as the (unknown) target distribution of the latent Markov chain, we have the following Latent LMC (L2MC) proposal

$$\begin{aligned}\mathbf{z}' &= \mathbf{z}_k + \frac{\tau}{2}\nabla_{\mathbf{z}} \log p_t(\mathbf{z}_k) + \sqrt{\tau} \cdot \boldsymbol{\varepsilon} \\ &= \mathbf{z}_k + \frac{\tau}{2}\nabla_{\mathbf{z}} \log \frac{p_t(\mathbf{z}_k)\left(\det J_{\mathbf{z}_k}^\top J_{\mathbf{z}_k}\right)^{-\frac{1}{2}}}{p_0(\mathbf{z}_k)\left(\det J_{\mathbf{z}_k}^\top J_{\mathbf{z}_k}\right)^{-\frac{1}{2}}} + \frac{\tau}{2}\nabla_{\mathbf{z}} \log p_0(\mathbf{z}_k) + \sqrt{\tau} \cdot \boldsymbol{\varepsilon} \\ &= \mathbf{z} + \frac{\tau}{2}\nabla_{\mathbf{z}} \log \frac{p_d(\mathbf{x}_k)}{p_g(\mathbf{x}_k)} + \frac{\tau}{2}\nabla_{\mathbf{z}} \log p_0(\mathbf{z}_k) + \sqrt{\tau} \cdot \boldsymbol{\varepsilon} \\ &= \mathbf{z} - \frac{\tau}{2}\nabla_{\mathbf{z}} \log(D^{-1}(\mathbf{x}_k) - 1) + \frac{\tau}{2}\nabla_{\mathbf{z}} \log p_0(\mathbf{z}_k) + \sqrt{\tau} \cdot \boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}),\end{aligned} \tag{14}$$

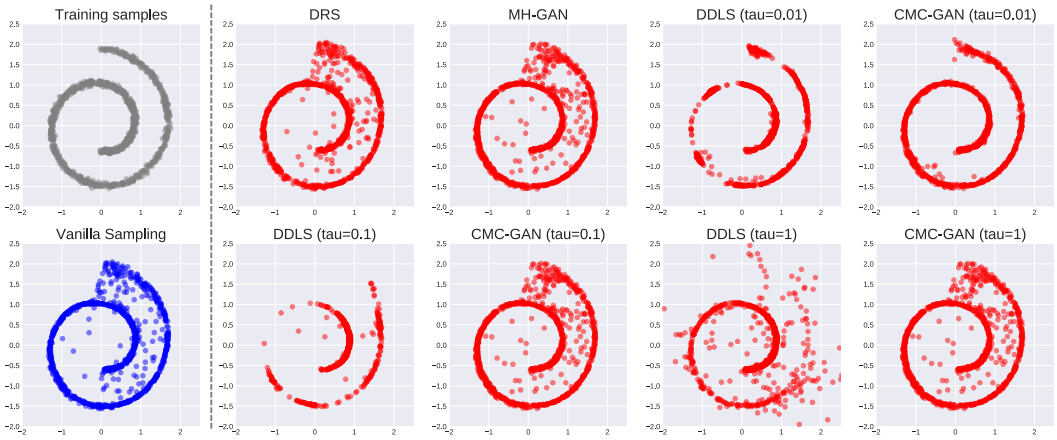

Figure 2: Visualization of samples with different sampling methods on the Swiss Roll dataset. Here tau denotes the Langevin step size in Eqn. (17).

where $\mathbf{x}_k = G(\mathbf{z}_k)$. As we can see, L2MC is made tractable by our structured dependent proposal with coupling Markov chains. DDLS (Che et al., 2020) proposes a similar Langevin proposal by formalizing GANs as an implicit energy-based model, while here we provide a straightforward derivation through reparameterization. Our major difference to DDLS is that CMC-GAN also includes a tractable MH correction step (Eqn. (9)), which accounts for the numerical errors introduced by the discretization and ensures that detailed balance holds.

4.4 Extension to WGAN

Our method can also be extended to other kinds of GAN, like Wasserstein GAN (WGAN) (Arjovsky et al., 2017). The WGAN objective is

$$\min_G \max_D \mathbb{E}_{\mathbf{x}\sim p_d(\mathbf{x})}[D(\mathbf{x})] - \mathbb{E}_{\mathbf{x}\sim p_g(\mathbf{x})}[D(\mathbf{x})], \tag{15}$$

where $D : \mathcal{X} \to \mathbb{R}$ is restricted to be a Lipschitz function. Under certain conditions, WGAN also implies an approximate estimation of the density ratio (Che et al., 2020),

$$D(\mathbf{x}) \approx \log \frac{p_d(\mathbf{x})}{p_g(\mathbf{x})} + \text{const} \quad \Rightarrow \quad \frac{p_d(\mathbf{x})}{p_g(\mathbf{x})} \approx \exp(D(\mathbf{x})) \cdot \text{const}. \tag{16}$$

Following the same derivations as in Eqn. (11) and (14), we will have the WGAN version of CMC-GAN. Specifically, with $\mathbf{x}_k = G(\mathbf{z}_k)$, the L2MC proposal follows

$$\mathbf{z}' = \mathbf{z}_k + \frac{\tau}{2}\nabla_{\mathbf{z}} D(\mathbf{x}_k) + \frac{\tau}{2}\nabla_{\mathbf{z}} \log p_0(\mathbf{z}_k) + \sqrt{\tau}\cdot\boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{17}$$

and the MH acceptance criterion is

$$\alpha_{CMC-W}(\mathbf{x}', \mathbf{x}_k) = \min\left(1, \frac{q(\mathbf{z}_k|\mathbf{z}')p_0(\mathbf{z}')}{q(\mathbf{z}'|\mathbf{z}_k)p_0(\mathbf{z}_k)} \cdot \frac{\exp\left(D(\mathbf{x}')\right)}{\exp\left(D(\mathbf{x}_k)\right)}\right). \tag{18}$$

5 Experiments

We show our empirical results both on synthetic and real image datasets.

5.1 Synthetic Data

Following DOT (Tanaka, 2019) and DDLS (Che et al., 2020), we apply CMC-GAN to the synthetic Swiss Roll dataset, where data samples lie on a Swiss roll manifold in the two-dimensional space. We construct the dataset by scikit-learn with 100,000 samples, and train a WGAN as in Tanaka (2019),

Table 2: Inception Scores (IS) of different sampling methods on CIFAR-10 and CelebA. * For GAN, DRS and MH-GAN, we report the results in Turner et al. (2019). ♯ The results of DDLS on these datasets are given by our re-implementation.

Method	CIFAR-10		CelebA	
	DCGAN	WGAN	DCGAN	WGAN
GAN*	2.879	3.073	2.332	2.788
DRS (Azadi et al., 2019)*	3.073	3.137	2.869	2.861
MH-GAN (Turner et al., 2019)	3.379	3.305	**3.106**	2.889
DDLS (Che et al., 2020)♯	3.518	3.547	2.534	2.862
CMC-GAN (ours)	**3.851**	**4.035**	2.686	**2.943**

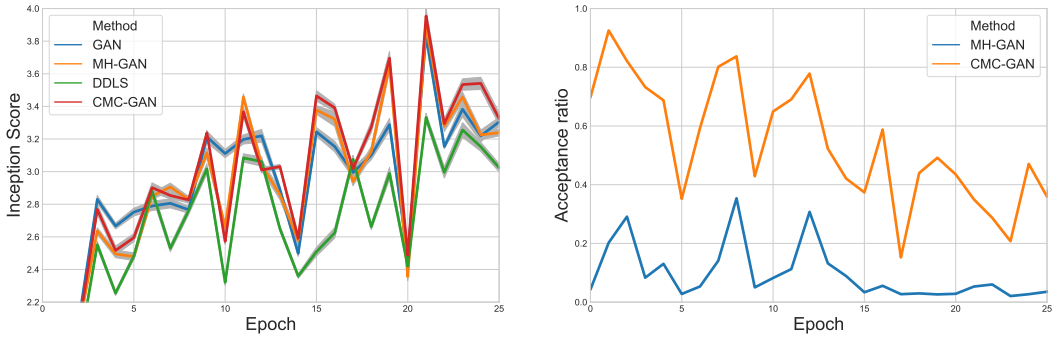

Figure 3: Average Inception Score (left) and acceptance ratio (right) vs. training epoch with DCGAN on CIFAR-10. The standard deviation of Inception Score is shown with shaded error bar.

where both the generator and discriminator are fully connected neural networks with leaky ReLU nonlinearities. We optimize the model using the Adam optimizer, with $\alpha = 0.0001, \beta_1 = 0.5, \beta_2 = 0.9$. After training, we draw $1,000$ samples with different sampling methods.

As shown in Figure 2, with appropriate step size ($\tau = 0.01$), the gradient-based methods (DDLS and CMC-GAN) outperform independent proposals (DRS and MH-GAN) by a large margin, while DDLS is more discontinuous on shape compared to CMC-GAN. In DDLS, when the step size becomes too large ($\tau = 0.1, 1$), the numerical error of the Langevin dynamics becomes so large that the chain either collapses or diverges. In contrast, those bad proposals are rejected by the MH correction steps of CMC-GAN, which prevents the misbehavior of the Markov chain.

5.2 Real Image Data

Following MH-GAN (Turner et al., 2019), we conduct experiments on two real-world image datasets, CIFAR-10 and CelebA, for DCGAN (Radford et al., 2015) and WGAN (Arjovsky et al., 2017). For a fair comparison of MCMC methods, each Markov chain starts from a GAN sample and runs 640 steps, and we take the last sample for evaluation. The step size τ of our L2MC proposal is 0.01 on CIFAR-10 and 0.1 on CelebA. We calibrate the discriminator with Logistic Regression as in Turner et al. (2019), and draw $50,000$ samples to compute the Inception Score (Salimans et al., 2016).

From Table 2, we can see our method outperforms state-of-the-art methods on both datasets. We also plot the Inception Score and acceptance ratio per epoch in Figure 3 based on our re-implementation. CMC-GAN and MH-GAN outperform DDLS by a large margin, showing that the MH correction steps also matter on real-world datasets. In most cases, CMC-GAN has higher Inception Scores and much higher acceptance ratios than MH-GAN, indicating the superiority of our structured dependent proposal. The average acceptance ratio of MH-GAN is lower than 0.2 in most cases, while CMC-GAN has an acceptance ratio of around 0.4 to 0.8, which is known to be a good tradeoff for MCMC algorithms.

Ablation study. From Table 3, we can see that without the MH correction step, the Langevin steps often result in worse sample quality. Meanwhile, the acceptance is very small on CIFAR-10

Table 3: Ablation study of our CMC-GAN with Inception Scores (IS) and acceptance ratios (Accept) averaged over five adjacent checkpoints. IND refers to the independent proposal of MH-GAN. CMC refers to our CMC proposal. MH denotes the MH rejection step of the corresponding sampler.

Proposal	MH	CIFAR10				CelebA			
		DCGAN		WGAN		DCGAN		WGAN	
		Accept	IS	Accept	IS	Accept	IS	Accept	IS
IND	✗	-	3.769	-	3.557	-	2.431	-	2.799
CMC	✗	-	3.312	-	3.500	-	2.465	-	2.868
IND	✓	0.033	3.735	0.084	3.509	**0.678**	**2.500**	0.747	2.879
CMC	✓	**0.363**	**3.820**	**0.447**	**3.598**	0.628	2.476	**0.955**	**2.894**

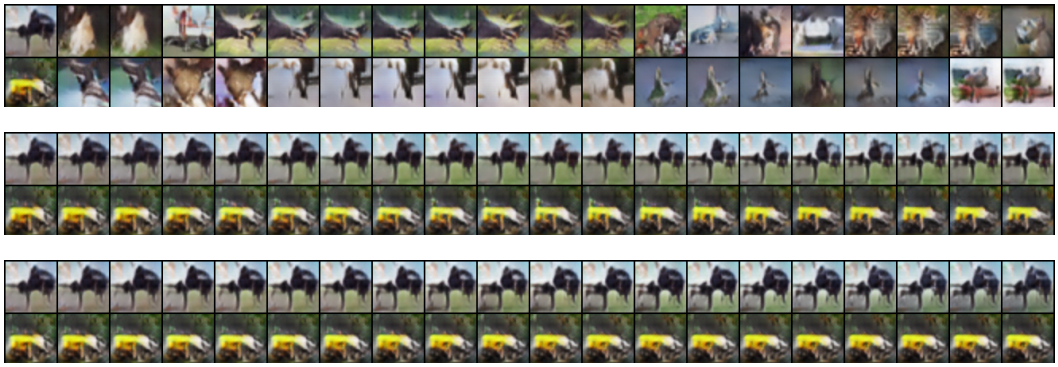

Figure 4: The first 20 steps of two Markov chains with the same initial samples. The chains are generated by MH-GAN (top), DDLS (middle) and CMC-GAN (bottom), respectively.

without the dependent CMC proposal. As a result, our CMC-GAN (CMC+MH) is the only setup that consistently improves over the baseline and obtains the best Inception Score on each dataset. The only exception is DCGAN on CelebA, where the independent proposal outperforms our CMC proposal with a higher acceptance ratio. We believe that this is because the human face samples of CelebA are very alike among each other, such that independent samples from the generator can also be easily accepted. Nevertheless, the acceptance ratio of the independent proposal can be much lower on datasets with diverse sources of images, like CIFAR-10.

Markov chain visualization. In Figure 4, we demonstrate two Markov chains sampled with different methods. We can see that MH-GAN is often trapped in the same place because of the independent proposals. DDLS and CMC-GAN instead gradually refine the samples with gradient steps. In addition, the MH rejection steps of CMC-GAN help avoid some bad artifacts in the images.

6 Conclusion

In this paper, we have proposed a novel method, CMC-GAN, to improve the sampling of GANs. We devise a structured dependent proposal that reparameterizes the sample-level transition of GANs into the latent-level transition. More importantly, we first prove that this general proposal meets a tractable MH criterion. Experiments show our method does not only improve sample efficiency but also demonstrate state-of-the-art sample quality on benchmark datasets over existing sampling methods.

References

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein GAN. *ICML*, 2017.

Samaneh Azadi, Catherine Olsson, Trevor Darrell, Ian Goodfellow, and Augustus Odena. Discriminator rejection sampling. *ICLR*, 2019.

Adi Ben-Israel. The change-of-variables formula using matrix volume. *SIAM Journal on Matrix Analysis and Applications*, 21(1):300–312, 1999.

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. *ICLR*, 2019.

Tong Che, Ruixiang Zhang, Jascha Sohl-Dickstein, Hugo Larochelle, Liam Paull, Yuan Cao, and Yoshua Bengio. Your GAN is secretly an energy-based model and you should use discriminator driven latent sampling. *ICML*, 2020.

Mevlana C Gemici, Danilo Rezende, and Shakir Mohamed. Normalizing flows on Riemannian manifolds. *arXiv preprint arXiv:1611.02304*, 2016.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *NeurIPS*, 2014.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of Wasserstein GANs. *NeurIPS*, 2017.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. *NeurIPS*, 2019.

Zinan Lin, Ashish Khetan, Giulia Fanti, and Sewoong Oh. PACGAN: The power of two samples in generative adversarial networks. *NeurIPS*, 2018.

Luke Metz, Ben Poole, David Pfau, and Jascha Sohl-Dickstein. Unrolled generative adversarial networks. *ICLR*, 2017.

Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *ICLR*, 2018.

Radford M Neal et al. MCMC using Hamiltonian dynamics. *Handbook of markov chain monte carlo*, 54:113–162, 2010.

Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training GANs. *NeurIPS*, 2016.

Yuxuan Song, Qiwei Ye, Minkai Xu, and Tie-Yan Liu. Discriminator contrastive divergence: semi-amortized generative modeling by exploring energy of the discriminator. *arXiv preprint arXiv:2004.01704*, 2020.

Akinori Tanaka. Discriminator optimal transport. *NeurIPS*, 2019.

Ryan Turner, Jane Hung, Yunus Saatci, and Jason Yosinski. Metropolis-Hastings generative adversarial networks. *ICML*, 2019.

A APPENDIX

A.1 ALGORITHM PROCEDURE

Algorithm 1 GAN sampling with coupling Markov chains (CMC-GAN)

Input: trained GAN with (calibrated) discriminator D and generator G, Markov chain length K, latent prior distribution $p_0(\mathbf{z})$, latent proposal distribution $q(\mathbf{z}'|\mathbf{z}_k)$;
Output: an improved GAN sample $\mathbf{x}_K$;

Draw an initial sample $\mathbf{x}_1$: 1) draw initial latent $\mathbf{z}_1 \sim p_0(\mathbf{z})$ and 2) push forward $\mathbf{x}_1 = G(\mathbf{z}_1)$;
for each step $k \in [1, K-1]$ **do**
  Draw a CMC proposal $\mathbf{x}' \sim q_{\mathrm{CMC}}(\mathbf{x}'|\mathbf{x}_k)$: 1) draw a latent proposal $\mathbf{z}' \sim q(\mathbf{z}'|\mathbf{z}_k)$, and 2) push forward $\mathbf{x}' = G(\mathbf{z}')$;
  Calculate the MH acceptance criterion $\alpha_{\mathrm{CMC}}(\mathbf{x}_k, \mathbf{x}')$ following Eqn. (9);
  Decide the acceptance of $\mathbf{x}'$ with probability $\alpha_{\mathrm{CMC}}(\mathbf{x}_k, \mathbf{x}')$;
  if $\mathbf{x}'$ is accepted **then**
    Let $\mathbf{x}_{k+1} = \mathbf{x}', \mathbf{z}_{k+1} = \mathbf{z}'$
  else
    Let $\mathbf{x}_{k+1} = \mathbf{x}_k, \mathbf{z}_{k+1} = \mathbf{z}_k$
  end if
end for

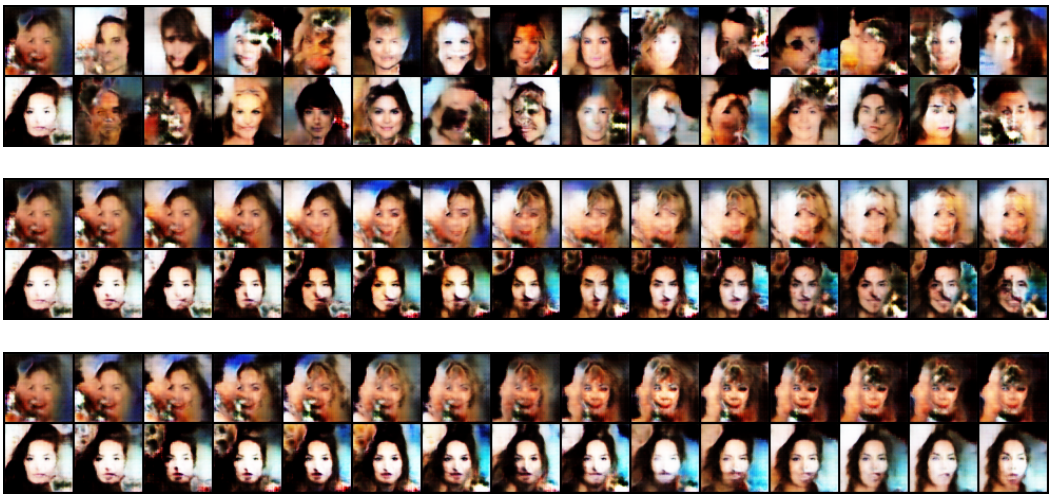

Figure 5: Visualization of the Markov chains generated by MH-GAN (top), DDLS (middle) and CMC-GAN (bottom) on CelebA.

A.2 ADDITIONAL RESULTS

We demonstrate two Markov chains on CelebA with different MCMC sampling methods of WGAN in Figure 5. We can see that on CelebA, the acceptance ratio of MH-GAN becomes much higher than that on CIFAR-10. Nevertheless, the sample quality is still relatively low. In comparison, the gradient-based method can gradually refine the samples with Langevin steps, and our CMC-GAN can alleviate image artifacts with MH correction steps.