

---

# Zeroth-Order Sampling Methods for Non-Log-Concave Distributions: Alleviating Metastability by Denoising Diffusion

---

**Ye He**

Georgia Institute of Technology  
yhe367@gatech.edu

**Kevin Rojas**

Georgia Institute of Technology  
kevin.rojas@gatech.edu

**Molei Tao**

Georgia Institute of Technology  
mtao@gatech.edu

## Abstract

This paper considers the problem of sampling from non-logconcave distribution, based on queries of its unnormalized density. It first describes a framework, Denoising Diffusion Monte Carlo (DDMC), based on the simulation of a denoising diffusion process with its score function approximated by a generic Monte Carlo estimator. DDMC is an oracle-based meta-algorithm, where its oracle is the assumed access to samples that generate a Monte Carlo score estimator. Then we provide an implementation of this oracle, based on rejection sampling, and this turns DDMC into a true algorithm, termed Zeroth-Order Diffusion Monte Carlo (ZOD-MC). We provide convergence analyses by first constructing a general framework, i.e. a performance guarantee for DDMC, **without** assuming the target distribution to be log-concave or satisfying any isoperimetric inequality. Then we prove that ZOD-MC admits an inverse polynomial dependence on the desired sampling accuracy, albeit still suffering from the curse of dimensionality. Consequently, for low dimensional distributions, ZOD-MC is a very efficient sampler, with performance exceeding latest samplers, including also-denoising-diffusion-based RDMC and RSDMC. Last, we experimentally demonstrate the insensitivity of ZOD-MC to increasingly higher barriers between modes or discontinuity in non-convex potential.

## 1 Introduction

The problem of drawing samples from a distribution based on unnormalized density  $\propto \exp(-V)$  (described by the potential  $V$ ) is a fundamental statistical and algorithmic problem. This classical problem nevertheless remains as a research frontier, providing pivotal tools to applications such as decision making, statistical inference / estimation, uncertainty quantification, data assimilation, and molecular dynamics. Worth mentioning is that machine learning could benefit vastly from progress in sampling as well, not only because of its connection to inference, optimization and approximation, but also through modern domains such as diffusion generative modeling & differential privacy.

Recent years have seen rapid developments of sampling algorithms with quantitative and non-asymptotic theoretical guarantees. Many of the results are either based on discretizations of diffusion processes [12, 13, 50, 16, 34, 33] or gradient flows [38, 10, 22]. In order to develop such guarantees, it is necessary to make assumptions about the target distributions, for instance, that it satisfies an isoperimetric property, where standard requirements are log-concavity or functional inequalities

[12, 56, 21, 8, 48]. However, there is empirical evidence that the corresponding algorithms struggle to sample from targets that have high barriers between modes that create metastability. Overcoming such issues is highly nontrivial and researchers have continued to develop new methods to tackle these problems.

Diffusion models have lately shown remarkable ability in the generative modeling setting, with applications including image, video, audio, and macromolecule generations. This created a wave of theoretical work that showed the ability of diffusion models to sample from distributions under minimal assumptions [14, 57, 5, 32, 29, 4, 11, 3]. However, these works all started with the assumption that there is access to an approximation of the score function with some accuracy. This is a reasonable assumption for the task of generative modeling when one spends enough efforts on the training of the score, but the task of sampling is different. A natural question is: can we leverage the insensitivity of diffusion models to multimodality to efficiently sample from unnormalized, non-log-concave density? This would require approximating the score, which is then used as an inner loop inside an outer loop that integrates reverse diffusion process to transport, e.g., Gaussian initial condition, to nearly the target distribution.

The seminal works by [26, 27, 20] try to answer this question using Monte Carlo estimators of the score function and to provide theoretical guarantees. We also mention earlier work by [53] which learns parameterized scores and more work along the same line by [58, 44, 54, 55], whose theoretical guarantees are less clear but are based on other interesting ideas. [26] proposed Reverse Diffusion Monte Carlo (RDMC), which estimates the score via LMC algorithm and relaxes the isoperimetric assumptions in the analysis of traditional sampling algorithms. [20] proposed a similar method, stochastic localization via iterative posterior sampling (SLIPS), which approximate the score via Metropolis-adjusted Langevin algorithm (MALA). However, both methods rely on the usage of a small time window where isoperimetric properties hold. This leaves the problem of finding a good initialization for the diffusion process. To alleviate this issue, [27] developed an acceleration of RDMC, the Recursive Score Diffusion-based Monte Carlo (RSDMC), which improves the non-asymptotic complexity to be quasi-polynomial in both dimension and inverse accuracy and gets rid of any isoperimetric assumption. Such work provides strong theoretical guarantees, however it requires a lot of computational power to get a high accuracy sampler. Additionally, RDMC, SLIPS and RSDMC are all based on first-order queries (i.e. gradients of  $V$ ), which brings extra computational and memory costs, in addition to requiring a continuous differentiable  $V$ . Motivated by these two observations, we create a sampler that only makes use of **zeroth-order** queries without assuming any isoperimetric conditions on the target distribution. Our contributions can be summarized as follows.

- We introduce an oracle-based meta-algorithm **DDMC (Denosing Diffusion Monte Carlo)** and provide a non-asymptotic guarantee in KL-divergence in Theorem 1. Our result provides theoretical insight on the choice of optimal step-size in DDMC as well as in denosing diffusion models (Sec. 3.2).
- We develop a novel algorithm **ZOD-MC (Zeroth Order Diffusion-Monte Carlo)** that uses zeroth-order queries and the global minimal value of the potential function to generate samples approximating the target distribution. In Corollary 3.1, we establish a zeroth-order query complexity upper bound for general target distributions satisfying mild smoothness and moment conditions. Our result is summarized and compared to other sampling algorithms in Table 5.
- The advantages of our algorithm are experimentally verified for non-log-concave target distributions. We demonstrate the insensitivity of our algorithm to various high barriers between modes, and the ability of correctly account for discontinuities in the potential.

## 2 Preliminaries

### 2.1 Diffusion Model

Diffusion model generates samples that are similar to training data, by requiring the generated data to follow the same latent distribution  $p$  as the training data. To do so, it considers a forward noising

<sup>1</sup>Assumption [A4] in [26] is a soft version of strongly log-concave outside a ball.

<sup>2</sup>This criterion measures the KL-divergence from the output distribution to a distribution that is closed to the target distribution in Wasserstein-2 distance. This criterion is considered in analyzing denosing diffusion models with step-size that accommodates the early stopping technique, see [4, 3]. This criterion does not apply to RDMC/RSDMC since they don't use early stopping and instead assume the target distribution to be smooth and fully supported on  $\mathbb{R}^d$ . See Sec.3.3 for more discussions.

Algorithms	Queries	Assumptions	Criterion	Oracle Complexity
LMC	first-order	LSI	KL	$\mathcal{O}(d\varepsilon^{-1})$
RDMC	first order	soft log-concave <sup>1</sup>	TV	$\exp(\mathcal{O}(\log(d))\tilde{\mathcal{O}}(\varepsilon^{-1}))$
RSDMC	first-order	None	KL	$\exp(\mathcal{O}(\log^3(d\varepsilon^{-1})))$
Proximal Sampler	zeroth-order	log-concave	KL	$\mathcal{O}(d\varepsilon^{-1})$
ZOD-MC	zeroth-order	None	KL + $W_2$ <sup>2</sup>	$\exp(\tilde{\mathcal{O}}(d)\mathcal{O}(\log(\varepsilon^{-1})))$

Table 1: Comparison of ZOD-MC to LMC, RDMC, RSDMC and the Proximal Sampler: Summary of isoperimetric assumptions and oracle complexities to generate a  $\varepsilon$ -accurate sample under different criterion.  $\tilde{\mathcal{O}}$  hides polylog( $d\varepsilon^{-1}$ ) factors. The zeroth-order oracle complexity of ZOD-MC is from Corollary 3.1 for achieving both  $\varepsilon$  KL and  $\varepsilon W_2$  errors. These theoretical results suggest that in the absence of isoperimetric assumptions, ZOD-MC excels in low-dimensions.

process that transforms a random variable into Gaussian noise. One most commonly used forward process is (a time reparameterization of) the Ornstein-Uhlenbeck (OU) process, given by the SDE:

$$dX_t = -X_t dt + \sqrt{2}dB_t, \quad X_0 \sim p, \quad (1)$$

where  $\{B_t\}_{t \geq 0}$  is the standard Brownian motion in  $\mathbb{R}^d$ . The OU process that solves (1) is in distribution equivalent to a sum of two independent random vectors:  $X_t = e^{-t}X_0 + \sqrt{1 - e^{-2t}}Z$  where  $(X_0, Z) \sim p \otimes \gamma^d$  and  $\gamma^d$  is the standard Gaussian distribution in  $\mathbb{R}^d$ . Denote  $p_t = \text{Law}(X_t)$  for all  $t \geq 0$ . If we consider a large, fixed terminal time  $T$  of (1), then  $p_T$  is close to  $\gamma^d$ . Then, the denoising or backwards diffusion process,  $\{\tilde{X}_t\}_{0 \leq t \leq T}$ , can be constructed by reversing the OU process from time  $T$ , meaning that  $\text{Law}(\tilde{X}_t) := \text{Law}(X_{T-t})$  for all  $t \in [0, T]$ . By doing so we obtain the denoising diffusion process which solves the following SDE:

$$d\tilde{X}_t = (\tilde{X}_t + 2\nabla \log p_{T-t}(\tilde{X}_t))dt + \sqrt{2}d\tilde{B}_t, \quad \tilde{X}_0 \sim p_T, \quad 0 \leq t \leq T, \quad (2)$$

where  $\{\tilde{B}_t\}_{0 \leq t \leq T}$  is a Brownian motion in  $\mathbb{R}^d$ , independent of  $\{B_t\}_{0 \leq t \leq T}$  and  $\nabla \log p_t$  is usually referred as the score function for  $p_t$ . Although the denoising process initializes at  $p_T$ , we can't generate exact samples from  $p_T$ . In practice, people consider the standard Gaussian initialization  $\gamma^d$  due to the fact that  $p_T$  is close to  $\gamma^d$  when  $T$  is large. The denosing process with the standard Gaussian initialization is given by

$$d\tilde{X}_t = (\tilde{X}_t + 2\nabla \log p_{T-t}(\tilde{X}_t))dt + \sqrt{2}d\tilde{B}_t, \quad \tilde{X}_0 \sim \gamma^d, \quad 0 \leq t \leq T. \quad (3)$$

By simulating this denoising process (3), we can achieve the goal of generating new samples. However, the denoising process (3) can't be simulated directly due to the fact that the score function is not explicitly known. A widely applied method to solve this issue is to learn the score function through denoising score matching [51, 24, 52]. Given a learned score, denoted as  $s(t, x)$ , one can simulate the denoising diffusion process using discretizations like the Euler Maruyama or some exponential integrator. From a theoretical perspective, assuming the learned score satisfies

$$\mathbb{E}_{x \sim p_t} [\|s(t, x) - \nabla \log p_t(x)\|^2] \leq \epsilon_{\text{score}}^2, \quad \forall 0 \leq t \leq T, \quad (4)$$

non-asymptotic convergence guarantees for diffusion models are obtained in [5, 3, 4, 11]. For instance, in [3], polynomial iteration complexities were proved without assuming any isoperimetric property of the data distribution and only assuming the data distribution has a finite second moment and a score estimator satisfying (4) is available.

In this work, we consider instead the sampling setting, in which no existing samples from the target distribution is available. Our sampling algorithm and theoretical analysis are motivated from the denoising diffusion process given by (2) and its corresponding discretization through the exponential integrator in Algorithm 1. In particular, we first introduce an oracle-based meta-algorithm, DDMC, which integrates Algorithm 1 and Algorithm 2, where the exponential integrator scheme of (2) is applied to generate samples and the score function is approximated by a Monte Carlo estimator assuming independent samples from a conditional distribution are available.

## 2.2 Rejection Sampling and Restricted Gaussian Oracle

Rejection sampling is a popular Monte Carlo method for sampling a target distribution,  $p$ , based on the zeroth-order queries of the potential  $V$ . It requires that we have access to the potential function

---

**Algorithm 1:** Denoising Diffusion Sampling via Exponential Integrator

---

**Input :**  $N \in \mathbb{Z}_+$ ,  $0 = t_0 < \dots < t_N = T - \delta$ , score estimator  $\{s(T - t_k, \cdot)\}_{k=0}^{N-1}$ .

**Output :**  $x_N$ .

generate a sample  $x_0 \sim \gamma^d$ ;

**for**  $k = 0, 1, \dots, N - 1$  **do**

    generate  $\xi_k \sim \gamma^d$  such that  $\xi_k$  is independent to  $\xi_0, \dots, \xi_{k-1}$ ;  
     $x_{k+1} \leftarrow e^{t_{k+1}-t_k} x_k + 2(e^{t_{k+1}-t_k} - 1)s(T - t_k, x_k) + \sqrt{e^{2(t_{k+1}-t_k)} - 1}\xi_k$ .

**end**

---

$V_\mu$  of some other distribution  $\mu$ , such that  $\mu$  is easy to sample from and  $\exp(-V) \leq \exp(-V_\mu)$  globally. Such a distribution  $\mu$  is typically called an envelope for the distribution  $p$ . With an envelope  $\nu$ , rejection sampling generates samples from  $p$  by running the following algorithm till acceptance:

1. Sample  $X \sim \mu$ ,
2. Accept  $X$  with probability  $\exp(-V(X) + V_\mu(X))$ .

The rejection sampling is considered as a high-accuracy algorithm as it outputs a unbiased sample from the target distribution. However, despite such a remarkable property, it has drawbacks. First, it is a nontrivial task to find an envelope for a general target distribution. Second, rejection sampling usually suffers from ‘‘curse of dimensionality’’. Even for strongly logconcave target distributions, the complexity of the rejection sampling increases exponentially fast with the dimension: in expectation it requires  $\kappa^{d/2}$  many rejections before one acceptance, where  $\kappa$  is the condition number for the potential  $V$ , see [9].

The Restricted Gaussian Oracle (RGO), which was first introduced in [31], assumes that an accurate sample from distribution  $\pi(\cdot|y) \propto \exp(-V(\cdot) - \frac{1}{2\eta}\|\cdot - y\|^2)$  can be generated for any  $y \in \mathbb{R}^d$ ,  $\eta > 0$  and any potential  $V$ . Implementing the RGO is challenging. It is usually done by rejection sampling. However, most proposed methods [36, 18], are only suitable for small  $\eta$ .

Our proposed sampling algorithm, ZOD-MC applies the rejection sampling (Algorithm 3) to implement the RGO with a large value of  $\eta$ . Details on ZOD-MC are introduced in Section 3.1.

### 3 Denoising Diffusion Monte Carlo Sampling

In this section, we first introduce DDMC and ZOD-MC in Section 3.1. Then we provide a convergence guarantee for DDMC in Section 3.2. Last, in Section 3.3, we establish the zeroth-order query complexity of ZOD-MC. Note DDMC is a meta-algorithm that still requires an implementation of its oracle, and ZOD-MC is an actual algorithm that contains such an implementation. The theoretical guarantee of ZOD-MC (Sec. 3.3), therefore, is based on the analysis framework of DDMC (Sec. 3.2).

#### 3.1 Denoising Diffusion Monte Carlo and Zeroth-Order Diffusion Monte Carlo

**Denoising Diffusion Monte Carlo (DDMC).** Let’s start with a known but helpful lemma on score representation, derivable from Tweedie’s formula [45].

**Lemma 1.** *Let  $\{X_t\}_{t \geq 0}$  be the solution to the OU process (1) and  $p_t = \text{Law}(X_t)$ . Then for all  $t > 0$ ,*

$$\nabla \ln p_t(x) = \mathbb{E}_{x_0 \sim p_{0|t}(\cdot|x)} \left[ \frac{e^{-t} x_0 - x}{1 - e^{-2t}} \right], \quad (5)$$

where  $p_{0|t}(\cdot|x) \propto \exp(-V(\cdot) - \frac{1}{2} \frac{\|\cdot - e^t x\|^2}{e^{2t} - 1})$  is the distribution of  $X_0$  conditioned on  $\{X_t = x\}$ .

This lemma was for example applied in [26] to do sampling based on the denoising diffusion process in (2). For the sake of completeness, we include its proof in Appendix C.5.

Due to (5), to approximate the score function  $\nabla \ln p_t(x)$ , it suffices to generate samples that approximate  $p_{0|t}(\cdot|x)$ . [26, 27] proposed to use Langevin-based algorithms to sample from  $p_{0|t}(\cdot|x)$ . The first step of our work is to generalize this, with refined and more general theoretical analysis later on, by considering an oracle algorithm, DDMC, which assumes independent samples  $\{z_{t,i}\}_{i=1}^{n(t)}$  that

---

**Algorithm 2:** Monte Carlo Score Estimation

---

**Input :**  $t \in (0, T]$ ,  $x \in \mathbb{R}^d$ ,  $n(t) \in \mathbb{Z}_+$ ,  $\delta(t) > 0$ .

**Output :**  $s(t, x)$ .

**Oracle :** generate independent  $\{z_{t,i}\}_{i=1}^{n(t)}$  such that  $W_2(\text{Law}(z_{t,i}), p_{0|t}(\cdot|x)) \leq \delta(t)$ .

$$s(t, x) \leftarrow \frac{1}{n(t)} \sum_{i=1}^{n(t)} \frac{x - e^{-t} z_{t,i}}{1 - e^{-2t}}.$$

---

approximate  $p_{0|t}(\cdot|x)$  are available. The Monte Carlo score estimator in Algorithm 2 is given by

$$s(t, x) = \frac{1}{n(t)} \sum_{i=1}^{n(t)} \frac{e^{-t} z_{t,i} - x}{1 - e^{-2t}}, \quad (6)$$

where  $n(t)$  is the number of samples and  $\delta(t)$  is such that  $W_2(\text{Law}(z_{t,i}), p_{0|t}(\cdot|x)) \leq \delta(t)$  for all  $i$ . In Section 3.2, we will discuss how the performance of sampling depends on  $n(t)$ ,  $\delta(t)$ .

**Zeroth-Order Diffusion Monte Carlo (ZOD-MC).** Noticing that in Lemma 1, the conditional distribution has a structured potential function: a summation of the target potential and a quadratic function. Therefore, implementing the oracle in DDMC is equivalent to implementing RGO with  $y = e^t x$  and  $\eta = e^{2t} - 1$ . Based on this, we propose ZOD-MC, a novel methodology based on rejection sampling and DDMC. Rejection sampling (Algorithm 3) can generate i.i.d. Monte Carlo samples required in Algorithm 2. Therefore, ZOD-MC, as a combination of rejection sampling and DDMC, can efficiently sample from non-logconcave distributions. See Appendix B for more details.

---

**Algorithm 3:** Rejection Sampling: generating  $\{z_{t,i}\}_{i=1}^{n(t)}$  in Algorithm 2

---

**Input :**  $x \in \mathbb{R}^d$ , zeroth-order queries of  $V$ .

**Output :**  $z$ .

**while TRUE do**

    Generate  $(\xi, u) \sim \gamma^d \otimes U[0, 1]$ ;

$z \leftarrow e^t x + \sqrt{e^{2t} - 1} \xi$ ;

**return**  $z$  **if**  $u \leq \exp(-V(z) + V^*)$ ;

**end**

---

**Remark 1.** (Remark on the optimization step) In theory, we assume an oracle access to the minimum value of  $V$ . However, in practice we use Newton's method to find a local minimum. Throughout the sampling process we update the local minimum as we explore the search space.

**Remark 2.** (Parallelization) Notice that Algorithm 3 can be run in parallel to generate all the  $n(t)$  samples required to compute the score. Contrary to methods like LMC that have a sequential nature, this allows our method to be more computationally efficient and reduce the running time. This is a feature that RDMC or RSDMC doesn't benefit as much from.

### 3.2 Convergence of DDMC

Our oracle-based meta-algorithm, DDMC, provides a framework for designing and analyzing sampling algorithms that integrate the denoising diffusion model and the Monte Carlo score estimation. In this section, we first present an error analysis to the Monte Carlo score estimation in Proposition 3.1, whose proof is in Appendix C.3. After that, we leverage our result in Proposition 3.1 and provide a non-asymptotic convergence result for DDMC in Theorem 1, whose proof is in Appendix C.4.

**Proposition 3.1.** Let  $\{X_t\}_{t \geq 0}$  be the solution of the OU process (1) and  $p_t = \text{Law}(X_t)$  for all  $t > 0$ . If we define  $s(t, x) = \frac{1}{n(t)} \sum_{i=1}^{n(t)} \frac{e^{-t} z_{t,i} - x}{1 - e^{-2t}}$  with  $\{z_{t,i}\}_{i=1}^{n(t)}$  being a sequence of independent random vectors such that  $W_2(\text{Law}(z_{t,i}), p_{0|t}(\cdot|x)) \leq \delta(t)$  for all  $t > 0$  and  $x \in \mathbb{R}^d$ , then we have

$$\mathbb{E}[\|\nabla \ln p_t(X_t) - s(t, X_t)\|^2] \leq \frac{e^{-2t}}{(1 - e^{-2t})^2} \delta(t)^2 + \frac{1}{n(t)} \frac{e^{-2t}}{(1 - e^{-2t})^2} \text{Cov}_p(x). \quad (7)$$

**Choice of  $\delta(t)$  and  $n(t)$ .** The error bound in (7) helps choose the accuracy threshold  $\delta(t)$  and the number of samples  $n(t)$  to control the score estimation error over different time. In fact, when

$t$  increases, it requires less samples and allows larger sample errors to get a good Monte Carlo score estimator. If we assume  $\text{Cov}_p(x) = \mathcal{O}(d)$  for simplicity, then when  $t$  is small, the factor  $\frac{e^{-2t}}{(1-e^{-2t})^2} = \mathcal{O}(t^{-2})$  and the choice of  $\delta(t) = \mathcal{O}(t\varepsilon)$  and  $n(t) = \Omega(dt^{-2}\varepsilon^{-2})$  will lead to the  $L^2$ -error of order  $\mathcal{O}(\varepsilon^2)$ . When  $t$  is large, the factor  $\frac{e^{-2t}}{(1-e^{-2t})^2} = \mathcal{O}(e^{-2t})$  and it only requires  $\delta(t) = \mathcal{O}(e^t\varepsilon)$  and  $n(t) = \Omega(de^{-2t}\varepsilon^{-2})$  to ensure the  $L^2$ -error is of order  $\mathcal{O}(\varepsilon^2)$ . In the latter case, the  $\delta(t)$  is of a larger order and  $n(t)$  is of a smaller order than the first case.

We now analyze the convergence of DDMC. Recall that Algorithm 1 is an exponential integrator discretization scheme of (2) with the time schedule  $0 = t_0 < t_1 < \dots < t_N = T - \delta$  for some  $\delta > 0$ . In each iteration,  $x_{k+1} = e^{t_{k+1}-t_k}x_k + 2(e^{t_{k+1}-t_k} - 1)s(T - t_k, x_k) + \sqrt{e^{2(t_{k+1}-t_k)} - 1}\xi_k$ , where  $\xi_k \sim \gamma^d$  and  $s(T - t_k, \cdot)$  is the Monte Carlo score estimator generated by Algorithm 2. The trajectory of Algorithm 1 can be piece-wisely characterized by the following SDEs: for all  $t \in [t_k, t_{k+1})$ ,

$$d\tilde{X}_t = (\tilde{X}_t + 2s(T - t_k, \tilde{X}_{t_k}))dt + \sqrt{2}d\tilde{B}_t, \quad \tilde{X}_0 \sim \gamma^d, \quad \tilde{X}_{t_k} = x_k. \quad (8)$$

Therefore, the convergence of DDMC is equivalent to the convergence of the process  $\{\tilde{X}_t\}_{0 \leq t \leq t_N}$ , which could be quantified under mild assumptions on the target distribution. Next, we present the moment assumption on the target distribution and our non-asymptotic convergence theorem.

**Assumption 3.1.** *The distribution  $p$  has a finite second moment:  $\mathbb{E}_{x \sim p}[\|x\|^2] = m_2^2 < \infty$ .*

**Theorem 1.** *Assume that the target distribution satisfies Assumption 3.1. Let  $\{X_t\}_{t \geq 0}$  be the solution of (1) with  $p_t := \text{Law}(X_t)$  and  $\{\tilde{X}_t\}_{t \geq 0}$  be the solution of (8) with  $q_t := \text{Law}(\tilde{X}_t)$ . For any  $\delta \in (0, 1)$  and  $T > 1$ , let  $0 = t_0 < t_1 < \dots < t_N = T - \delta$  be a time schedule such that  $\gamma_k = \Theta(\gamma_{k-1})$  for all  $k = 0, 1, \dots, N - 1$ , where  $\gamma_k := t_{k+1} - t_k$ . Then*

$$\begin{aligned} KL(p_\delta | q_{t_N}) &\lesssim \underbrace{(d + m_2^2)e^{-2T}}_I + \underbrace{\sum_{k=0}^{N-1} \frac{\gamma_k e^{-2(T-t_k)}}{(1-e^{-2(T-t_k)})^2} \left( \delta(T-t_k)^2 + \frac{m_2^2}{n(T-t_k)} \right)}_{II} \\ &\quad + \underbrace{\frac{m_2^2 e^{-2T} \gamma_0}{(1-e^{-2T})^2} + \sum_{k=1}^{N-1} \frac{(d+m_2^2)\gamma_k \gamma_{k-1}^2}{(1-e^{-2(T-t_k)})(1-e^{-2(T-t_{k-1}})})^2}}_{III} + \sum_{k=0}^{N-1} \frac{d\gamma_k^2}{(1-e^{-2(T-t_k)})^2}, \quad (9) \end{aligned}$$

where  $\delta(t), n(t)$  are parameters in Algorithm 2.

**Remark 3.** *In Theorem 1, we characterize  $KL(p_\delta | q_{t_N})$  instead of  $KL(p | q_{t_N})$  due to the fact that  $KL(p | q_{t_N})$  is not well-defined when the target distribution  $p$  is not smooth w.r.t. the Lebesgue measure. It turns out that  $p_\delta$  is an alternative distribution to look at because  $p_\delta$  is smooth for all  $\delta > 0$  and  $p_\delta$  is close to  $p$  when  $\delta$  is small (see Proposition C.1). This is a standard treatment, referred to as early stopping, in the score-based generative modeling literature [e.g., 5, 3].*

The terms I, II, III in (9) correspond to the three types of errors in Algorithm 1, the initialization error, the score estimation error and the discretization error, respectively. Such a decomposition is very common in analyses of diffusion models, see [5, 6, 3]. Since we consider the sampling setting, the score estimation error is derived from the error analysis to the Monte Carlo score estimator, i.e., Proposition 3.1. A detailed discussion on these three types of errors is provided in Appendix C.4.

Compared to existing analyses on diffusion models, Theorem 1 extends result in [3] from exponential-decay step-size to general choices of step-size, and recovers their sharp linear dimension dependence in the discretization error, with minimal assumptions on the target distribution. By assuming two consecutive step sizes are of the same order, we perform asymptotic estimation on the accumulated discretization errors and obtain a bound that depends on the step-size. Such kind of result helps to understand the optimal time schedule, as we will discuss soon.

**Discussion on the choices of time schedule.** Under different choices of step-size, the discretization errors in the denoising diffusion model have the same linear dependence on  $d$ , but different dependence on  $\delta$ . The linear dimension dependence improves the results in [4], where  $\mathcal{O}(d^2)$  discretization error bounds are proved for different choices of step-size. It is also an extension of the result in [3], where  $\mathcal{O}(d)$  discretization error is only proved for the exponential-decay step-size. In fact, Theorem 1 implies that the exponential-decay step-size induces an optimal discretization error up to some constant in term of the inverse early-stopping time  $\delta^{-1}$ . Detailed discussions on different choices of time schedules is provided in Appendix C.5.

### 3.3 Complexity of ZOD-MC

With the convergence result for DDMC in Theorem 1, we introduce the query complexity bound of ZOD-MC. Our analysis assumes a relaxation of the commonly used gradient-Lipschitz condition on the potential. The formal statement is presented in Corollary 3.1, whose proof is provided in Appendix C.6.

**Assumption 3.2.** *There exists a constant  $L > 0$  such that for any  $x^* \in \arg \min_{y \in \mathbb{R}^d} V(y)$  and  $x \in \mathbb{R}^d$ ,  $V$  satisfies  $V(x) - V(x^*) \leq \frac{L}{2} \|x - x^*\|^2$ .*

**Corollary 3.1.** *Under the assumptions in Theorem 1 and Assumption 3.2, if we set  $T = \frac{1}{2} \ln(\frac{d+m_2}{\varepsilon_{KL}})$ ,  $\gamma_k = \kappa \min(1, T - t_k)$ ,  $\delta = \min(\frac{\varepsilon_{w_2}^2}{d}, \frac{\varepsilon_{w_2}}{m_2})$ ,  $\kappa = \Theta(\frac{T + \ln(\delta^{-1})}{N})$ , then to obtain an output (with distribution  $q_{t_N}$ ) in ZOD-MC such that  $W_2(p, p_\delta) \lesssim \varepsilon_{w_2}$  and  $KL(p_\delta, q_{t_N}) \lesssim \varepsilon_{KL}$ , the zeroth-order query complexity is of order*

$$\tilde{O}\left(\max\left(\frac{d+m_2}{\varepsilon_{KL}}, \frac{d^2}{\varepsilon_{KL}^2}\right) \varepsilon_{KL}^{-\frac{d-2}{2}} (d+m_2)^{\frac{d-2}{2}} L^{\frac{d}{2}} d^{-1}\right) \max_{0 \leq k \leq N-1} \exp(L\|x^*\|^2 + \|x_k\|^2), \quad (10)$$

where the  $\tilde{O}$  hides polylog( $\frac{d+m_2}{\varepsilon_{w_2}}$ ) factors.

**Remark 4.** *If we assume WLOG that the minimizer of the potential is at the origin, i.e.,  $x^* = 0$ , and further make reasonable assumptions that  $m_2^2$ ,  $L$  and  $\{\|x_k\|^2\}$  are all of order  $\mathcal{O}(d)$ , where  $\{x_k\}$  are the iterates in Algorithm 1, then the query complexity of ZOD-MC is of order  $\exp(\tilde{O}(d) \log(\varepsilon_{KL}^{-1}))$ . Even though this complexity bound has an exponential dimension dependence, it only depends polynomially on the inverse accuracy. Since it applies to any target distribution satisfying Assumptions 3.1 and 3.2, this complexity bound suggests that with the same overall complexity, ZOD-MC can generate samples more accurate than other algorithms in Table 5, for a large class of **low-dimensional non-logconcave** target distributions.*

**Comparison to LMC, RDMC and RSDMC.** When no isoperimetric condition is assumed, we compare convergence for ZOD-MC to convergence for LMC, RDMC and RSDMC.

In the absence of the isoperimetric condition, [1] demonstrated that LMC is capable of producing samples that are close to the target in FI assuming the target potential is smooth. However, FI is a weaker divergence than KL divergence/ Wasserstein-2 distance. It has been observed that, in certain instances, the KL divergence/Wasserstein-2 distance may still be significantly different from zero, despite a minimal FI value. This observation implies that the convergence criteria based on FI may not be as stringent as our result which is based on KL divergence/Wasserstein-2 distance. [26] proved that RDMC produces samples that are  $\varepsilon$ -close to the target in KL divergence with high probability. Assuming the potential is smooth and a tail-growth condition, the first order oracle complexity is shown to be of order  $\exp(\varepsilon^{-1} \log d)$ . [27] introduced RSDMC as an acceleration of RDMC. They were able to show that if the potential is smooth, RSDMC produces a sample that is  $\varepsilon$ -close to the target in KL divergence with high probability. The first order oracle complexity is shown to be of order  $\exp(\log^3(d/\varepsilon))$ . Compared to RDMC and RSDMC, our result on ZOD-MC doesn't require the potential to be smooth as our Assumption 3.2 is only a growth condition of the potential. This indicates that our convergence result applies to targets with non-smooth, or even discontinuous potentials. Our result in Corollary 3.1 shows the zeroth-order oracle complexity for ZOD-MC is of order  $\exp(d \log(\varepsilon^{-1}))$ , which achieves a better  $\varepsilon$ -dependence compared to RDMC and RSDMC, at the price of a worse dimension dependence. This suggests that, for any low-dimensional target, ZOD-MC produces a more accurate sample than RDMC/RSDMC when the overall oracle complexity are the same. Last, zeroth-order queries cost less computationally than first-order queries in practice, which also makes ZOD-MC a more suitable sampling algorithm when the gradients of the potential are hard to compute.

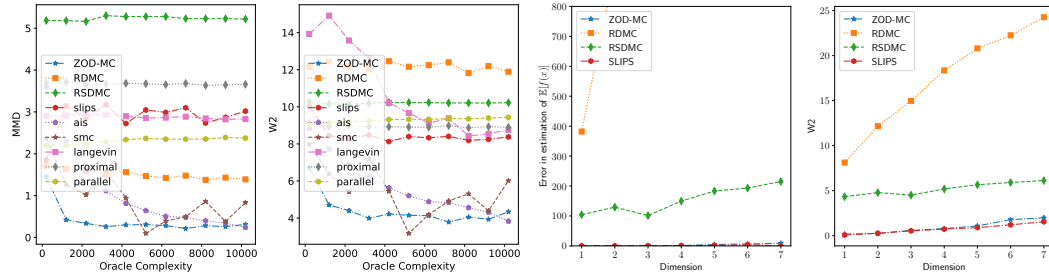
## 4 Experiments

We will demonstrate ZOD-MC on three examples, namely Gaussian mixtures, Gaussian mixtures plus discontinuities, and Müller-Brown which is a highly-nonlinear, nonconvex test problem popular in computational chemistry and material sciences. Multiple Gaussian mixtures will be considered, for showcasing the robustness of our method under worsening isoperimetric properties. The baselines we consider include RDMC [26], RSDMC [27], SLIPS [20], the proximal sampler [37], annealed

importance sample [42], sequential Monte Carlo [15], a parallel tempering approach with MALA proposals [30] and naive unadjusted Langevin Monte Carlo. All the experiments are conducted using a NVIDIA GeForce RTX 4070 Laptop GPU with 8GB of VRAM and Pytorch.

#### 4.1 Results for Gaussian Mixtures

**Matched Oracle Complexity.** We modify a 2D Gaussian mixture example frequently considered in the literature to make it more challenging, by making its modes unbalanced with non-isotropic variances, resulting in a highly asymmetrical, multi-modal problem. We include the full details of the parameters in Appendix D. We fix the same oracle complexity (total number of  $0^{th}$  and  $1^{st}$  order  $V$  queries) for different methods, and show the generated samples in Figure 2. Note matching oracle complexity puts our method at a disadvantage, since other techniques require querying the gradient, which results in more function evaluations. Despite this, we see in Figure 1a that our method achieves both the lowest MMD and  $W_2$  using the least number of oracle complexity.



(a) *Sampling accuracy against oracle complexity.* For any fixed oracle complexity, ZOD-MC has the least error both in MMD and  $W_2$ . Note diffusion based methods do not have an initialization. Thus curves don't start at the same  $y$ -value. (b) *Sampling accuracy against dimension* We demonstrate that other diffusion based methods scale poorly with dimension. On the left we plot the error when evaluating statistics of the generated samples and on the right we analyze the  $W_2$  metric.

Figure 1: Accuracies of different methods for sampling Gaussian Mixture

**Robustness Against Mode Separation.** Now let's further separate the modes in the mixture to investigate the robustness of our method to increasing nonconvexity/metastability. More precisely, we scale the means of each mode by a constant factor to have a mode located at  $(0, R)$ ; doing so increases the barriers between the modes and exponentially worsens the isoperimetric properties of the target distribution [49]. Figure 4a shows our method is the most insensitive to mode separation. Being the only one that can successfully sample from all modes, as observed in Figure 3, ZOD-MC suffers less from metastability. Note there is still some dependence on mode separation due to the  $x_k$  dependence in the complexity bound in Corollary 3.1.

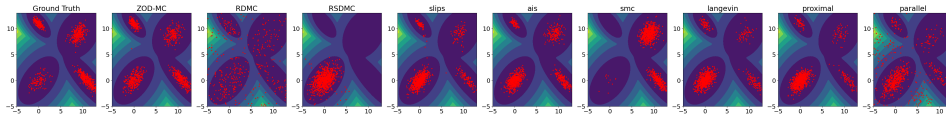


Figure 2: *Sampling from asymmetric, unbalanced Gaussian Mixture.* All diffusion-based methods (ZOD-MC, RDMC, RSDMC) use 2200 oracles per score evaluation. Langevin and the proximal sampler are set to use the same total amount of oracles as diffusion based methods. While other methods suffer from metastability, ZOD-MC correctly samples all modes.

**Dimension Dependence Against Other Diffusion Based Methods.** One drawback of our method, is its bad dimension dependence when compared to diffusion based methods. For instance, RDMC and RSDMC have a dependence of  $\exp(\mathcal{O}(\log(d))\tilde{\mathcal{O}}(\varepsilon^{-1}))$  and  $\exp(\mathcal{O}(\log^3(d\varepsilon^{-1})))$  respectively, in comparison to our  $\exp(\tilde{\mathcal{O}}(d)\mathcal{O}(\log(\varepsilon^{-1})))$ . Despite this theoretical disadvantage, we find empirically that these methods don't scale well with dimension either. To demonstrate this we sample 5 points on the positive quadrant and use them as means for a GMM. We then evaluate statistics on the generated samples and  $W_2$  as a function of dimension. We observe in Figure 1b that under a fixed number of function evaluations our method results in the lowest  $W_2$ . More details are in Appendix D. **Discontinuous Potentials.** The use of zeroth-order queries allows ZOD-MC to solve problems that would be completely infeasible to first order methods. To demonstrate this, we modify the



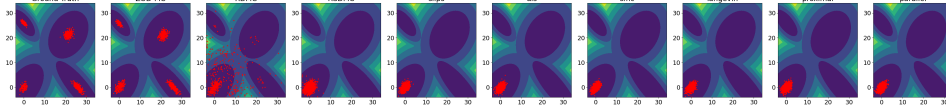
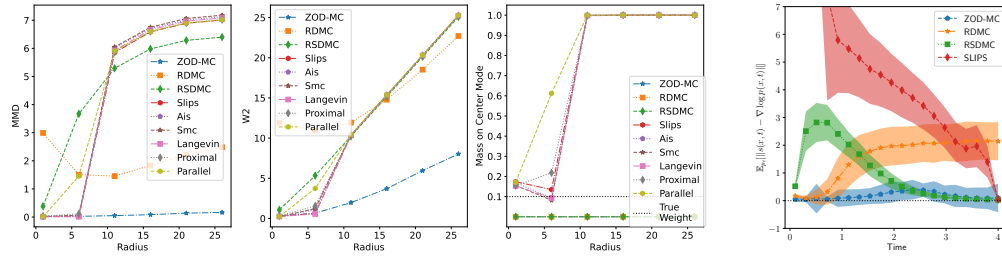


Figure 3: *Gaussian Mixture with further separated modes* ( $R = 26$ ). ZOD-MC can overcome strengthened metastability and sample from every mode, while other methods are stuck at the mode at the origin, where every method is initialized.

potential in Figure 2. We consider  $V(x) + U(x)$  where  $U$  is a discontinuous function given by  $U(x) = 8 \mathbb{1}_{\{5 < \|x\| < 11\}}$ . This creates an annulus of much lower probability and a strong potential barrier. In the original problem, the mode centered at the origin was chosen to have the smallest weight (0.1), but adding this discontinuity significantly changes the problem. As observed in Figure 5, our method is still able to correctly sample from the target distribution, while other methods not only continue to suffer from metastability but also fail to see the discontinuities. We quantitatively evaluate the sampling accuracy by using rejection sampling (slow but unbiased) to obtain ground truth samples, and then compute MMD and  $W_2$ . See Appendix D.2 for details.



(a) *Sampling accuracy against how separated modes are.* ZOD-MC is the least sensitive to mode separation. The oracle complexity is fixed, independent of how modes are separated. (b) *Average Score Error as a function of time.* Shaded is the standard deviation of the errors.

Figure 4: *Accuracies of generated samples against dimension and Score Error.* On the right, the result for SLIPS is not directly comparable as it has a different forward process.

**Score Approximation of Diffusion Based Methods.** One explanation of our method’s great success in comparison with RDMC and RSDMC is the ability to approximate the score correctly. We select an unbalanced assymetrical 5d GMM and evaluate the average  $L^2$  score error between methods. On Figure 4b we show that the best approximations of the score are found by using ZODMC as an estimator as opposed to other methods. Even as  $t$  increases and the approximation gets harder we are able to retain accuracy and therefore generate high quality samples.

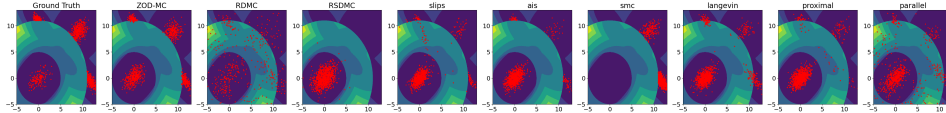


Figure 5: *Generated samples for discontinuous Gaussian Mixture.* Our method can recover the target distribution even under the presence of discontinuities. The same oracle complexity is again used in each method, 3200 per score evaluation in diffusion-based approaches.

## 4.2 Results of Müller Brown Potential

The Müller Brown potential is a toy model for molecular dynamics. Its highly nonlinear potential has 3 modes despite of being the sum of 4 exponentials. The original version has 2 of its modes corresponding to negligible probabilities when compared to the 3rd, which is not good to visualization and comparison across different methods. Thus we consider a balanced version [35] and further translate and dilate  $x$  and  $y$  so that one of the modes is centered near the origin. The details of the potential can be found in Appendix D.5. Our method is the only one that can correctly sample from all 3 modes as observed in Figure 6 (note they are leveled).

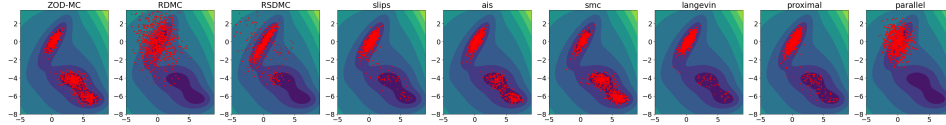


Figure 6: *Generated samples for the Müller Brown potential.* We overlay the generated samples on top of the level curves of  $V(x)$ . All methods use 1100 oracles.

## Acknowledgments and Disclosure of Funding

The authors are grateful for the partially support by NSF DMS-1847802, Cullen-Peck Scholarship, and GT-Emory Humanity.AI Award. We thank the anonymous reviewers for their helpful comments.

## References

- [1] K. Balasubramanian, S. Chewi, M. A. Erdogdu, A. Salim, and S. Zhang. Towards a theory of non-log-concave sampling: first-order stationarity guarantees for langevin monte carlo. In *Conference on Learning Theory*, pages 2896–2923. PMLR, 2022.
- [2] C. J. Bélisle, H. E. Romeijn, and R. L. Smith. Hit-and-run algorithms for generating multivariate distributions. *Mathematics of Operations Research*, 18(2):255–266, 1993.
- [3] J. Benton, V. De Bortoli, A. Doucet, and G. Deligiannidis. Linear convergence bounds for diffusion models via stochastic localization. *ICLR*, 2024.
- [4] H. Chen, H. Lee, and J. Lu. Improved analysis of score-based generative modeling: User-friendly bounds under minimal smoothness assumptions. In *International Conference on Machine Learning*, pages 4735–4763. PMLR, 2023.
- [5] S. Chen, S. Chewi, J. Li, Y. Li, A. Salim, and A. Zhang. Sampling is as easy as learning the score: theory for diffusion models with minimal data assumptions. *ICLR*, 2022.
- [6] Y. Chen, S. Chewi, A. Salim, and A. Wibisono. Improved analysis for a proximal algorithm for sampling. In *Conference on Learning Theory*, pages 2984–3014. PMLR, 2022.
- [7] Y. Chen and R. Eldan. Localization schemes: A framework for proving mixing bounds for markov chains. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 110–122. IEEE, 2022.
- [8] Y. Chen and K. Gatmiry. A simple proof of the mixing of metropolis-adjusted langevin algorithm under smoothness and isoperimetry. *arXiv preprint arXiv:2304.04095*, 2023.
- [9] S. Chewi. *Log-concave sampling*. 2023. Book draft available at <https://chewisinho.github.io/>.
- [10] S. Chewi, T. Le Gouic, C. Lu, T. Maunu, and P. Rigollet. Sgvd as a kernelized wasserstein gradient flow of the chi-squared divergence. *Advances in Neural Information Processing Systems*, 33:2098–2109, 2020.
- [11] G. Conforti, A. Durmus, and M. G. Silveri. Score diffusion models without early stopping: finite fisher information is all you need. *arXiv preprint arXiv:2308.12240*, 2023.
- [12] A. S. Dalalyan and A. Karagulyan. User-friendly guarantees for the langevin monte carlo with inaccurate gradient. *Stochastic Processes and their Applications*, 129(12):5278–5311, 2019.
- [13] A. S. Dalalyan and L. Riou-Durand. On sampling from a log-concave density using kinetic langevin diffusions. *Bernoulli*, 26(3):1956–1988, 2020.
- [14] V. De Bortoli. Convergence of denoising diffusion models under the manifold hypothesis. *TMLR*, 2022.
- [15] P. Del Moral, A. Doucet, and A. Jasra. Sequential monte carlo samplers. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 68(3):411–436, 2006.
- [16] R. Dwivedi, Y. Chen, M. J. Wainwright, and B. Yu. Log-concave sampling: Metropolis-hastings algorithms are fast. *Journal of Machine Learning Research*, 20(183):1–42, 2019.
- [17] M. Dyer, A. Frieze, and R. Kannan. A random polynomial-time algorithm for approximating the volume of convex bodies. *Journal of the ACM (JACM)*, 38(1):1–17, 1991.
- [18] J. Fan, B. Yuan, and Y. Chen. Improved dimension dependence of a proximal algorithm for sampling. In *The Thirty Sixth Annual Conference on Learning Theory*, pages 1473–1521. PMLR, 2023.
- [19] A. Garbuno-Inigo, F. Hoffmann, W. Li, and A. M. Stuart. Interacting langevin diffusions: Gradient structure and ensemble kalman sampler. *SIAM Journal on Applied Dynamical Systems*, 19(1):412–441, 2020.

- [20] L. Grenioux, M. Noble, M. Gabri e, and A. Oliviero Durmus. Stochastic localization via iterative posterior sampling. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pages 16337–16376. PMLR, 21–27 Jul 2024.
- [21] Y. He, K. Balasubramanian, and M. A. Erdogdu. On the ergodicity, bias and asymptotic normality of randomized midpoint sampling method. *Advances in Neural Information Processing Systems*, 33:7366–7376, 2020.
- [22] Y. He, K. Balasubramanian, B. K. Sriperumbudur, and J. Lu. Regularized stein variational gradient flow. *arXiv preprint arXiv:2211.07861*, 2022.
- [23] Y. He, T. Farghly, K. Balasubramanian, and M. A. Erdogdu. Mean-square analysis of discretized it  diffusions for heavy-tailed sampling. *Journal of Machine Learning Research*, 25(43):1–44, 2024.
- [24] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.
- [25] D. Holzm uller and F. Bach. Convergence rates for non-log-concave sampling and log-partition estimation. *arXiv preprint arXiv:2303.03237*, 2023.
- [26] X. Huang, H. Dong, Y. Hao, Y. Ma, and T. Zhang. Reverse diffusion monte carlo. *ICLR*, 2024.
- [27] X. Huang, D. Zou, H. Dong, Y. Ma, and T. Zhang. Faster sampling without isoperimetry via diffusion-based monte carlo. *COLT*, 2024.
- [28] M. A. Iglesias, K. J. Law, and A. M. Stuart. Ensemble kalman methods for inverse problems. *Inverse Problems*, 29(4):045001, 2013.
- [29] H. Lee, J. Lu, and Y. Tan. Convergence of score-based generative modeling for general data distributions. In *International Conference on Algorithmic Learning Theory*, pages 946–985. PMLR, 2023.
- [30] H. Lee and Z. Shen. Improved bound for mixing time of parallel tempering. *arXiv preprint arXiv:2304.01303*, 2023.
- [31] Y. T. Lee, R. Shen, and K. Tian. Structured logconcave sampling with a restricted gaussian oracle. In *Conference on Learning Theory*, pages 2993–3050. PMLR, 2021.
- [32] G. Li, Y. Wei, Y. Chen, and Y. Chi. Towards faster non-asymptotic convergence for diffusion-based generative models. *ICLR*, 2024.
- [33] R. Li, M. Tao, S. S. Vempala, and A. Wibisono. The mirror Langevin algorithm converges with vanishing bias. In *International Conference on Algorithmic Learning Theory*, pages 718–742. PMLR, 2022.
- [34] R. Li, H. Zha, and M. Tao. Sqrt(d) Dimension Dependence of Langevin Monte Carlo. In *ICLR*, 2021.
- [35] X. H. Li and M. Tao. Automated construction of effective potential via algorithmic implicit bias. *arXiv preprint arXiv:2401.03511*, 2024.
- [36] J. Liang and Y. Chen. A proximal algorithm for sampling. *arXiv preprint arXiv:2202.13975*, 2022.
- [37] J. Liang and Y. Chen. A proximal algorithm for sampling. *arXiv preprint arXiv:2202.13975*, 2022.
- [38] Q. Liu. Stein variational gradient descent as gradient flow. *Advances in neural information processing systems*, 30, 2017.
- [39] L. Lov asz and M. Simonovits. The mixing rate of markov chains, an isoperimetric inequality, and computing the volume. In *Proceedings [1990] 31st annual symposium on foundations of computer science*, pages 346–354. IEEE, 1990.

- [40] L. Lovász and S. Vempala. Hit-and-run from a corner. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, pages 310–314, 2004.
- [41] K. L. Mengersen and R. L. Tweedie. Rates of convergence of the hastings and metropolis algorithms. *The annals of Statistics*, 24(1):101–121, 1996.
- [42] R. M. Neal. Annealed importance sampling. *Statistics and computing*, 11:125–139, 2001.
- [43] L. Pardo. *Statistical inference based on divergence measures*. CRC press, 2018.
- [44] L. Richter, J. Berner, and G.-H. Liu. Improved sampling via learned diffusions. *ICLR*, 2024.
- [45] H. E. Robbins. An empirical bayes approach to statistics. In *Breakthroughs in Statistics: Foundations and basic theory*, pages 388–394. Springer, 1992.
- [46] G. O. Roberts and R. L. Tweedie. Geometric convergence and central limit theorems for multidimensional hastings and metropolis algorithms. *Biometrika*, 83(1):95–110, 1996.
- [47] A. Roy, L. Shen, K. Balasubramanian, and S. Ghadimi. Stochastic zeroth-order discretizations of langevin diffusions for bayesian inference. *Bernoulli*, 28(3):1810–1834, 2022.
- [48] A. Salim, L. Sun, and P. Richtarik. A convergence theory for svgd in the population limit under talagrand’s inequality t1. In *International Conference on Machine Learning*, pages 19139–19152. PMLR, 2022.
- [49] A. Schlichting. Poincaré and log–sobolev inequalities for mixtures. *Entropy*, 21(1):89, 2019.
- [50] R. Shen and Y. T. Lee. The randomized midpoint method for log-concave sampling. *Advances in Neural Information Processing Systems*, 32, 2019.
- [51] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. *ICML*, 2015.
- [52] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.
- [53] F. Vargas, W. Grathwohl, and A. Doucet. Denoising diffusion samplers. *arXiv preprint arXiv:2302.13834*, 2023.
- [54] F. Vargas, A. Ovsianas, D. Fernandes, M. Girolami, N. D. Lawrence, and N. Nüsken. Bayesian learning via neural schrödinger–föllmer flows. *Statistics and Computing*, 33(1):3, 2023.
- [55] F. Vargas, S. Padhy, D. Blessing, and N. Nüsken. Transport meets variational inference: Controlled monte carlo diffusions. In *The Twelfth International Conference on Learning Representations*, 2024.
- [56] S. Vempala and A. Wibisono. Rapid convergence of the unadjusted langevin algorithm: Isoperimetry suffices. *Advances in neural information processing systems*, 32, 2019.
- [57] K. Yingxi Yang and A. Wibisono. Convergence of the inexact langevin algorithm and score-based generative models in kl divergence. *arXiv e-prints*, pages arXiv–2211, 2022.
- [58] Q. Zhang and Y. Chen. Path integral sampler: a stochastic control approach for sampling. *ICLR*, 2022.

## A Related Works on Zeroth-Order Sampling

The zeroth-order sampling algorithms have been widely studied in the past decades. There is a class of zeroth-order sampling algorithms, including the Ensemble Kalman Inversion [28] and the Ensemble Kalman Sampler [19], that are based on moving a set of easy-to-sample particle according to certain dynamics. However, these methods require (noisy) observations from the target distribution rather than queries of the potential function. Within the zeroth-order sampling algorithms using queries of the potential function, one type of methods make use of the zeroth-order queries to approximate the gradient and apply it to some first-order sampling algorithm [12, 47, 23]. Since it is based on the first-order methods, the analysis of this type of algorithms assumes the target distribution satisfies certain isoperimetric property in general. The other type of methods utilize the zeroth-order queries directly without relating to the gradient. Such methods include the Rejection sampling algorithm, the Metropolized Random Walk (MRW) [41, 46], Ball Walk [39, 17], Hit-and-Run algorithm [2, 40]. The rejection sampling algorithm requires to finding an envelope function which is easy to sample from. This could be difficult. MRW requires sufficient smooth and light tail of the target distribution to mix fast. Ball walk and Hit-and-Run algorithms assume the target distribution is compactly supported. In this paper, we develop a zeroth-order sampling algorithm based on the reverse OU process. Our algorithm does not suffer from the difficulties in the rejection sampling and MRW, and our analysis does not assume isoperimetric property and compact support of the target distribution.

[25] also studies the complexity of a zeroth-order sampling algorithm that combines an approximation technique and rejection sampling. For target distributions that are  $m$ -differentiable with compact support, [25][Theorem 12] implies a complexity of order  $\Omega_d(\varepsilon^{-d/m})$  to reach an  $\varepsilon$ -accuracy in KL-divergence, where the dimension dependence is implicit. Compare to our result in Corollary 3.1, both complexities are polynomial in  $\varepsilon$  and exponential in  $d$ . Our complexity is smaller for less smooth targets ( $m < 2$ ) while their complexity is smaller for smoother targets ( $m > 2$ ). However, result in [25] only applies to smooth targets ( $m > 0$ ) with compact supports, while our Corollary 3.1 applies to more general target distributions which can be with full support with even discontinuous potentials.

## B More Details on ZOD-MC

In this section, we provide more details on how rejection sampling (Algorithm 3) in ZOD-MC implements the oracle in DDMC, i.e., generating Monte Carlo samples required in Algorithm 2. **Construction of an envelope.** If we have  $V^*$  as a minimum value of  $V$ , then by noting that:

$$-V(z) - \frac{1}{2} \frac{\|z - e^t x\|^2}{e^{2t} - 1} \leq -V^* - \frac{1}{2} \frac{\|z - e^t x\|^2}{e^{2t} - 1}, \quad \forall z \in \mathbb{R}^d.$$

We are able to construct an envelope for rejection sampling. In particular we propose a samples  $z$  from  $\mathcal{N}(\cdot; e^t x, e^{2t} - 1)$  and accept proposal  $z$  with probability  $\exp(-V(z) + V^*)$ .

**Sampling from the target distribution.** Algorithm 3, implements the oracle in Algorithm 2 with  $\delta(t) = 0$ . When  $n(t)$  increases, Algorithm 2 outputs unbiased Monte Carlo score estimators with smaller variance, hence closer to the true score. We will quantify the convergence of DDMC next and consequently demonstrate that ZOD-MC can sample general non-logconcave distributions.

## C Proofs

### C.1 Properties of the OU-Process

In this section, we introduce and prove some useful properties of the OU-process. Throughout this section, we denote  $\{X_t\}_{t \geq 0}$  as the solution of (1) with  $p_t := \text{Law}(X_t)$ . For any  $s, t > 0$ ,  $p_{t|s}$  denotes the conditional probability measure of  $X_t$  given the value of  $X_s$ .

**Proposition C.1.** (Decay along the OU-Process) *Let  $\{X_t\}_{t \geq 0}$  be the solution of (1) with  $p_t := \text{Law}(X_t)$ . Assume that the initial distribution  $p$  satisfies Assumption 3.1. Then we have*

$$W_2(p_t, p)^2 \leq (1 - e^{-t})^2 m_2^2 + (1 - e^{-2t})d, \quad (11)$$

$$\text{and } KL(p_t | \gamma^d) \leq \frac{1}{2} \frac{e^{-4t}}{1 - e^{-2t}} d + \frac{1}{2} e^{-2t} m_2^2. \quad (12)$$

*Proof of Proposition C.1.* The proof for (11) is based on the fact that the solution to (1) can be represented by

$$X_t = e^{-t}X_0 + \sqrt{1 - e^{-2t}}Z, \quad \forall t \geq 0. \quad (13)$$

We have

$$\begin{aligned} W_2(p_t, p)^2 &\leq \mathbb{E}[\|X_t - X_0\|^2] \leq \mathbb{E}_{(X_0, Z) \sim p \otimes \gamma^d}[\|(e^{-t} - 1)X_0 + \sqrt{1 - e^{-2t}}Z\|^2] \\ &= (1 - e^{-t})^2 m_2^2 + (1 - e^{-2t})d. \end{aligned}$$

Next, to prove (12), we have

$$\text{KL}(p_t | \gamma^d) = \text{KL}\left(\int p_{t|0}(\cdot|y)p(dy) | \gamma^d(\cdot)\right) \leq \int \text{KL}(p_{t|0}(\cdot|y) | \gamma^d)p(dy),$$

where the inequality follows from the convexity of KL divergence. According to (13),  $p_{t|0}(\cdot|y)$  is a Gaussian measure with mean  $e^{-t}y$  and covariance matrix  $(1 - e^{-2t})I_d$ . According to [43], we have

$$\begin{aligned} \text{KL}(p_{t|0}(\cdot|y) | \gamma^d) &= \text{KL}(\mathcal{N}(e^{-t}y, (1 - e^{-2t})I_d) | \mathcal{N}(0, I_d)) \\ &= \frac{1}{2}(-d \ln(1 - e^{-2t}) - e^{-2t}d + e^{-2t}\|y\|^2). \end{aligned}$$

As a result, we get

$$\text{KL}(p_t | \gamma^d) \leq -\frac{d}{2} \ln(1 - e^{-2t}) - \frac{d}{2}e^{-2t} + \frac{1}{2}e^{-2t} \int \|y\|^2 p(dy) \leq \frac{e^{-4t}}{2(1 - e^{-2t})}d + \frac{1}{2}e^{-2t}m_2^2,$$

where the last inequality follows from the fact that  $\ln(1 + x) \leq x$  for all  $x > 0$ .  $\square$

**Proposition C.2.** (*Stochastic Dynamics along the OU-Process*) Let  $\{X_t\}_{t \geq 0}$  be the solution of (1). Define  $m_t(X_t) := \mathbb{E}_{X_0 \sim p_{0|t}(\cdot|X_t)}[X_0]$  and  $\Sigma_t(X_t) := \text{Cov}_{X_0 \sim p_{0|t}(\cdot|X_t)}(X_0) = \mathbb{E}_{X_0 \sim p_{0|t}(\cdot|X_t)}[(X_0 - m_t(X_t))^{\otimes 2}]$ . Then we have for all  $t \geq 0$ ,

$$\frac{d}{dt} \mathbb{E}[\Sigma_t(X_t)] = \frac{2e^{-2t}}{(1 - e^{-2t})^2} \mathbb{E}[\Sigma_t(X_t)^2].$$

The above proposition is known in stochastic localization literature [7] and diffusion model literature [3]. We present its proof for the sake of completeness.

*Proof of Proposition C.2.* For any  $T > 0$ , from (13), we have the conditional distribution

$$p_{0|t}(dx | X_t) \propto \exp\left(-\frac{1}{2} \frac{\|X_t - e^{-t}x\|^2}{1 - e^{-2t}}\right) p(dx),$$

where  $\{X_t\}_{0 \leq t \leq T}$  is the solution of (1). Noticing that the solution of (2),  $\{\bar{X}_t\}_{0 \leq t \leq T}$  is the reverse process of  $\{X_t\}_{0 \leq t \leq T}$  and it satisfies  $\bar{X}_t = X_{T-t}$  in distribution for all  $t \in [0, T]$ . Therefore, it suffices to study

$$\begin{aligned} q_{0|t}(dx | \bar{X}_t) &:= Z^{-1} \exp\left(-\frac{1}{2} \frac{\|\bar{X}_t - e^{-(T-t)}x\|^2}{1 - e^{-2(T-t)}}\right) p(dx) \\ &= Z_t^{-1} \exp\left(-\frac{1}{2} \frac{e^{-2(T-t)}}{1 - e^{-2(T-t)}} \|x\|^2 + \frac{e^{-(T-t)}}{1 - e^{-2(T-t)}} \langle x, \bar{X}_t \rangle\right) p(dx) \\ &:= Z_t^{-1} \exp(h_t(x)) p(dx), \end{aligned} \quad (14)$$

where the normalization constant  $Z_t = \int_{\mathbb{R}^d} \exp(h_t(x)) p(dx)$ . We have  $q_{0|t}(dx | \bar{X}_t) = p_{0|T-t}(dx | X_{T-t})$  in distribution for all  $t \in [0, T]$  and

$$\bar{m}_t(\bar{X}_t) := \mathbb{E}_{X_0 \sim q_{0|t}(\cdot|\bar{X}_t)}[X_0] = \mathbb{E}_{X_0 \sim p_{0|T-t}(\cdot|X_{T-t})}[X_0] = m_{T-t}(X_{T-t}), \quad (15)$$

$$\bar{\Sigma}_t(\bar{X}_t) := \text{Cov}_{X_0 \sim q_{0|t}(\cdot|\bar{X}_t)}(X_0) = \text{Cov}_{X_0 \sim p_{0|T-t}(\cdot|X_{T-t})}(X_0) = \Sigma_{T-t}(X_{T-t}). \quad (16)$$

where the above two identities hold in distribution. For simplicity, we denote  $\sigma_t = \sqrt{1 - e^{-2t}}$ . Then  $h_t(x) = -\frac{1}{2}(\sigma_{T-t}^{-2} - 1)\|x\|^2 + \sigma_{T-t}^{-1} \sqrt{\sigma_{T-t}^{-2} - 1} \langle x, \bar{X}_t \rangle$  is a stochastic process linearly depending on

$\{\bar{X}_t\}_{t \geq 0}$ . The conditional measure  $\{q_{0|t}(x|\bar{X}_t)\}_{t \geq 0}$  is a measure-valued stochastic process, whose dynamics can be studied by applying Itô's formula. First we have

$$\begin{aligned} dh_t(x) &= \sigma_{T-t}^{-3} \dot{\sigma}_{T-t} \|x\|^2 dt + (\sigma_{T-t}^{-2} - 1)^{-\frac{1}{2}} (-2\sigma_{T-t}^{-4} \dot{\sigma}_{T-t} + \sigma_{T-t}^{-2} \dot{\sigma}_{T-t}) \langle x, \bar{X}_t \rangle dt \\ &\quad + \sigma_{T-t}^{-1} (\sigma_{T-t}^{-2} - 1)^{\frac{1}{2}} \langle x, d\bar{X}_t \rangle, \end{aligned} \quad (17)$$

$$d[h(x), h(x)]_t = \sigma_{T-t}^{-2} (\sigma_{T-t}^{-2} - 1) \|x\|^2 d[\bar{X}, \bar{X}]_t, \quad (18)$$

Since  $\{\bar{X}_t\}_{0 \leq t \leq T}$  solves (2), according to Lemma 1, it satisfies that

$$\begin{aligned} d\bar{X}_t &= (\bar{X}_t + 2\mathbb{E}_{x \sim p_{0|T-t}(\cdot|\bar{X}_t)} \left[ \frac{e^{-(T-t)x} - \bar{X}_t}{1 - e^{-2(T-t)}} \right]) dt + \sqrt{2} d\bar{B}_t \\ &= (-\sigma_{T-t}^{-2} (2 - \sigma_{T-t}^{-2}) \bar{X}_t + 2(1 - \sigma_{T-t}^2)^{\frac{1}{2}} \bar{m}_t(\bar{X}_t)) dt + \sqrt{2} d\bar{B}_t. \end{aligned} \quad (19)$$

Based on (17), (18) and (19), we have

$$\begin{aligned} dZ_t &= \int_{\mathbb{R}^d} \exp(h_t(x)) (dh_t(x) + \frac{1}{2} d[h(x), h(x)]_t) p(dx) \\ &= (\sigma_{T-t}^{-3} \dot{\sigma}_{T-t} + \sigma_{T-t}^{-2} (\sigma_{T-t}^{-2} - 1)) \mathbb{E}_{q_{0|t}(\cdot|\bar{X}_t)} [\|x\|^2] Z_t dt \\ &\quad + (\sigma_{T-t}^{-2} - 1)^{-\frac{1}{2}} (-2\sigma_{T-t}^{-4} \dot{\sigma}_{T-t} + \sigma_{T-t}^{-2} \dot{\sigma}_{T-t}) \langle \bar{m}_t(\bar{X}_t), \bar{X}_t \rangle Z_t dt \\ &\quad + \sigma_{T-t}^{-1} (\sigma_{T-t}^{-2} - 1)^{\frac{1}{2}} \langle \bar{m}_t(\bar{X}_t), d\bar{X}_t \rangle Z_t, \\ \text{and } d \ln Z_t &= Z_t^{-1} dZ_t - \frac{1}{2} Z_t^{-2} d[Z, Z]_t \\ &= (\sigma_{T-t}^{-3} \dot{\sigma}_{T-t} + \sigma_{T-t}^{-2} (\sigma_{T-t}^{-2} - 1)) \mathbb{E}_{q_{0|t}(\cdot|\bar{X}_t)} [\|x\|^2] dt \\ &\quad + (\sigma_{T-t}^{-2} - 1)^{-\frac{1}{2}} (-2\sigma_{T-t}^{-4} \dot{\sigma}_{T-t} + \sigma_{T-t}^{-2} \dot{\sigma}_{T-t}) \langle \bar{m}_t(\bar{X}_t), \bar{X}_t \rangle dt \\ &\quad + \sigma_{T-t}^{-1} (\sigma_{T-t}^{-2} - 1)^{\frac{1}{2}} \langle \bar{m}_t(\bar{X}_t), d\bar{X}_t \rangle \\ &\quad + \sigma_{T-t}^{-2} (\sigma_{T-t}^{-2} - 1) \|\bar{m}_t(\bar{X}_t)\|^2 dt. \end{aligned} \quad (20)$$

If we define  $R_t(\bar{X}_t) = \frac{q_{0|t}(dx|\bar{X}_t)}{p(dx)} = Z_t^{-1} \exp(h_t(x))$ , then apply Itô's formula again and we have

$$\begin{aligned} dR_t(\bar{X}_t) &= d \exp(\ln R_t(\bar{X}_t)) \\ &= R_t(\bar{X}_t) d(\ln R_t(\bar{X}_t)) + \frac{1}{2} R_t(\bar{X}_t) d[\ln R_t(\bar{X}_t), \ln R_t(\bar{X}_t)] \\ &= R_t(\bar{X}_t) dh_t(x) - R_t(\bar{X}_t) d \ln Z_t + \frac{1}{2} R_t(\bar{X}_t) d[h_t(x) - \ln Z_t, h_t(x) - \ln Z_t]. \end{aligned} \quad (21)$$

Now combine the results in (17), (18), (20) and (21), we can derive the differential equation of  $\bar{m}_t(\bar{X}_t)$ :

$$\begin{aligned} d\bar{m}_t(\bar{X}_t) &= d \int_{\mathbb{R}^d} x R_t(\bar{X}_t) p(dx) \\ &= \int_{\mathbb{R}^d} x (dh_t(x) - d \ln Z_t + \frac{1}{2} d[h_t(x) - \ln Z_t, h_t(x) - \ln Z_t]) q_{0|t}(dx|\bar{X}_t) \\ &= \sigma_{T-t}^{-3} \dot{\sigma}_{T-t} \mathbb{E}_{q_{0|t}(\cdot|\bar{X}_t)} [\|x\|^2 x] dt \\ &\quad + (\sigma_{T-t}^{-2} - 1)^{-\frac{1}{2}} (-2\sigma_{T-t}^{-4} \dot{\sigma}_{T-t} + \sigma_{T-t}^{-2} \dot{\sigma}_{T-t}) \mathbb{E}_{q_{0|t}(\cdot|\bar{X}_t)} [x^{\otimes 2}] \bar{X}_t dt \\ &\quad + \sigma_{T-t}^{-1} (\sigma_{T-t}^{-2} - 1)^{\frac{1}{2}} \mathbb{E}_{q_{0|t}(\cdot|\bar{X}_t)} [x^{\otimes 2} d\bar{X}_t] \\ &\quad - \sigma_{T-t}^{-3} \dot{\sigma}_{T-t} \mathbb{E}_{q_{0|t}(\cdot|\bar{X}_t)} [\|x\|^2] \bar{m}_t(\bar{X}_t) dt \\ &\quad - (\sigma_{T-t}^{-2} - 1)^{-\frac{1}{2}} (-2\sigma_{T-t}^{-4} \dot{\sigma}_{T-t} + \sigma_{T-t}^{-2} \dot{\sigma}_{T-t}) \bar{m}_t(\bar{X}_t)^{\otimes 2} \bar{X}_t dt \\ &\quad - \sigma_{T-t}^{-1} (\sigma_{T-t}^{-2} - 1)^{\frac{1}{2}} \bar{m}_t(\bar{X}_t)^{\otimes 2} d\bar{X}_t \\ &\quad - \sigma_{T-t}^{-2} (\sigma_{T-t}^{-2} - 1) \|\bar{m}_t(\bar{X}_t)\|^2 \bar{m}_t(\bar{X}_t) dt \\ &\quad + \sigma_{T-t}^{-2} (\sigma_{T-t}^{-2} - 1) \mathbb{E}_{q_{0|t}(\cdot|\bar{X}_t)} [\|x - \bar{m}_t(\bar{X}_t)\|^2 x] dt. \end{aligned} \quad (22)$$



Utilize (19) and the definition of  $\sigma_t$ , all terms with factor  $dt$  in the above equation cancel and (22) can be simplified as

$$\begin{aligned} d\bar{m}_t(\bar{X}_t) &= \frac{\sqrt{2}e^{-(T-t)}}{1 - e^{-2(T-t)}} \mathbb{E}_{q_{0|t}(\cdot|\bar{X}_t)} [x \otimes (x - \bar{m}_t(\bar{X}_t)) dB_t] \\ &= \frac{\sqrt{2}e^{-(T-t)}}{1 - e^{-2(T-t)}} \bar{\Sigma}_t(\bar{X}_t) d\bar{B}_t. \end{aligned} \quad (23)$$

Last, we derive the differential equation that  $\mathbb{E}_{X_t \sim p_t} [\Sigma_t(X_t)]$  satisfies. Let  $f(t) := \mathbb{E}_{X_t \sim p_t} [\Sigma_t(X_t)]$  and  $g(t) := \mathbb{E}_{X_t \sim p_t} [\Sigma_t(X_t)^2]$  be two deterministic functions on  $[0, T]$ . According to (16) and (23), we have

$$\begin{aligned} \frac{d}{dt} f(T-t) &= \frac{d}{dt} \mathbb{E}_{X_{T-t} \sim p_{T-t}} [\Sigma_{T-t}(X_{T-t})] \\ &= \frac{d}{dt} \mathbb{E}_{\bar{X}_t \sim p_{T-t}} [\bar{\Sigma}_t(\bar{X}_t)] \\ &= \frac{d}{dt} \mathbb{E}_{\bar{X}_t \sim p_{T-t}} [\mathbb{E}_{q_{0|t}(\cdot|\bar{X}_t)} [x^{\otimes 2}] - \bar{m}_t(\bar{X}_t)^{\otimes 2}] \\ &= \frac{d}{dt} \mathbb{E}_{x \sim p} [x^{\otimes 2}] - \frac{d}{dt} \mathbb{E}_{X_t \sim p_t} [\bar{m}_t(\bar{X}_t)^{\otimes 2}] \\ &= -\frac{2e^{-2(T-t)}}{(1 - e^{-2(T-t)})^2} \mathbb{E}_{\bar{X}_t \sim p_{T-t}} [\bar{\Sigma}_t(\bar{X}_t)^2] \\ &= -\frac{2e^{-2(T-t)}}{(1 - e^{-2(T-t)})^2} g(T-t). \end{aligned}$$

where the last inequality follows from the Itô isometry. Proposition C.2 is then proved by reverse the time in  $f$  and  $g$ .  $\square$

## C.2 Proofs of Section 3.1

*Proof of Lemma 1.* Based on (13), we have  $p_t = (e^{-t})_{\#p} * (\sqrt{1 - e^{-2t}})_{\#\gamma^d}$  where  $(e^{-t})_{\#p}$  is the pushforward measure of  $p$  via map  $x \in \mathbb{R}^d \mapsto e^{-t}x$  and  $(\sqrt{1 - e^{-2t}})_{\#\gamma^d}$  is the pushforward measure of  $\gamma^d$  via map  $x \in \mathbb{R}^d \mapsto \sqrt{1 - e^{-2t}}x \in \mathbb{R}^d$ . The pushforward measures  $(e^{-t})_{\#p}$  and  $(\sqrt{1 - e^{-2t}})_{\#\gamma^d}$  can be written as

$$\begin{aligned} (e^{-t})_{\#p}(dx) &= e^{td} p(e^t dx) \quad \text{and} \\ (\sqrt{1 - e^{-2t}})_{\#\gamma^d}(dx) &= (2\pi(1 - e^{-2t}))^{-\frac{d}{2}} \exp\left(-\frac{\|x\|^2}{2(1 - e^{-2t})}\right) dx, \end{aligned}$$

respectively. Therefore the score function  $\nabla \ln p_t(x)$  can be written as

$$\begin{aligned} \nabla \ln p_t(x) &= p_t(x)^{-1} e^{td} (2\pi(1 - e^{-2t}))^{-\frac{d}{2}} \nabla_x \int \exp\left(-\frac{\|x - z\|^2}{2(1 - e^{-2t})}\right) p(e^t dz) \\ &= p_t(x)^{-1} (2\pi(1 - e^{-2t}))^{-\frac{d}{2}} \nabla_x \int \exp\left(-\frac{\|x - e^{-t}z\|^2}{2(1 - e^{-2t})}\right) p(dz) \\ &= \int \frac{x - e^{-t}z}{1 - e^{-2t}} \frac{p_{t|0}(x|z)p(dz)}{p_t(x)} \\ &= \int \frac{x - e^{-t}z}{1 - e^{-2t}} p_{0|t}(dz|x), \end{aligned}$$

where the last step follows from the Bayesian rule and

$$p_{0|t}(\cdot|x) \propto p_{t|0}(x|\cdot)p(\cdot) \propto \exp\left(-V(\cdot) - \frac{1}{2} \frac{\|x - e^{-t}\cdot\|^2}{1 - e^{-2t}}\right).$$

$\square$

### C.3 Proofs of Section 3.2

*Proof of Proposition 3.1.* With the score estimator given in Algorithm 2, we have

$$\begin{aligned} & \mathbb{E}[\|\nabla \ln p_t(X_t) - s(t, X_t)\|^2] \\ &= \mathbb{E}_{x \sim p_t}[\|\nabla \ln p_t(x) - \frac{1}{n(t)} \sum_{i=1}^{n(t)} \frac{e^{-t} z_{t,i} - x}{1 - e^{-2t}}\|^2] \\ &= \mathbb{E}_{x \sim p_t}[\|\nabla \ln p_t(x) - \frac{1}{n(t)} \sum_{i=1}^{n(t)} \frac{e^{-t} x_{t,i} - x}{1 - e^{-2t}} + \frac{1}{n(t)} \sum_{i=1}^{n(t)} \frac{x_{t,i} - z_{t,i}}{1 - e^{-2t}}\|^2], \end{aligned}$$

where  $\{x_{t,i}\}_{i=1}^{n(t)}$  is a sequence of i.i.d. samples following  $p_{0|t}(\cdot|x)$  that are chosen such that  $\mathbb{E}[\|x_{t,i} - z_{t,i}\|^2|x] = W_2(p_{0|t}(\cdot|x), \text{Law}(z_{t,i}))$  for all  $t > 0$  and  $i = 1, 2, \dots, n(t)$ . Based on Lemma 1,  $\{\frac{e^{-t} x_{t,i} - x}{1 - e^{-2t}}\}_{i=1}^{n(t)}$  is a sequence of unbiased i.i.d. Monte Carlo estimator of  $\nabla \ln p_t(x)$  for all  $t > 0$  and  $x \in \mathbb{R}^d$ . Therefore, we get

$$\begin{aligned} & \mathbb{E}[\|\nabla \ln p_t(X_t) - s(t, X_t)\|^2] \\ &= \mathbb{E}_{x \sim p_t}[\|\nabla \ln p_t(x) - \frac{1}{n(t)} \sum_{i=1}^{n(t)} \frac{e^{-t} x_{t,i} - x}{1 - e^{-2t}} + \|^2] + \mathbb{E}[\|\frac{1}{n(t)} \sum_{i=1}^{n(t)} \frac{x_{t,i} - z_{t,i}}{1 - e^{-2t}}\|^2] \\ &= \underbrace{\frac{1}{n(t)^2} \sum_{i=1}^{n(t)} \mathbb{E}_{x \sim p_t}[\|\frac{e^{-t} x_{t,i} - x}{1 - e^{-2t}} - \mathbb{E}_{x_{t,i} \sim p_{0|t}(\cdot|x)}[\frac{e^{-t} x_{t,i} - x}{1 - e^{-2t}}]\|^2]}_{N_1} \\ & \quad + \underbrace{\frac{e^{-2t}}{(1 - e^{-2t})^2} \frac{1}{n(t)^2} \sum_{i,j=1}^{n(t)} \mathbb{E}[\langle x_{t,i} - z_{t,i}, x_{t,j} - z_{t,j} \rangle]}_{N_2}. \end{aligned}$$

The first term in the above equation,  $N_1$ , is related to the covariance of  $p_{0|t}(\cdot|X_t)$ , which is studied in Proposition C.2. We have

$$\begin{aligned} N_1 &= \frac{e^{-2t}}{(1 - e^{-2t})^2} \frac{1}{n(t)^2} \sum_{i=1}^{n(t)} \mathbb{E}_{x \sim p_t}[\text{trace}(\text{Cov}_{x_{t,i} \sim p_{0|t}(\cdot|x)}(x_{t,i}))] \\ &= \frac{e^{-2t}}{(1 - e^{-2t})^2} \frac{1}{n(t)} \mathbb{E}_{x \sim p_t}[\text{trace}(\Sigma_t(x))] \\ &\leq \frac{e^{-2t}}{(1 - e^{-2t})^2} \frac{1}{n(t)} \mathbb{E}_{x \sim \gamma^d}[\text{trace}(\Sigma_\infty(x))] \\ &= \frac{e^{-2t}}{(1 - e^{-2t})^2} \frac{1}{n(t)} \text{Cov}_p(x), \end{aligned}$$

where the inequality follows from Proposition C.2 indicating that  $t \mapsto \mathbb{E}_{x \sim p_t}[\text{trace}(\Sigma_t(x))]$  is a increasing function.

The second term  $N_2$  characterize the bias from the Monte Carlo samples and the bias can be measured by the Wasserstein-2 distance:

$$\begin{aligned} N_2 &\leq \frac{e^{-2t}}{(1 - e^{-2t})^2} \frac{1}{n(t)^2} \sum_{i,j=1}^{n(t)} \mathbb{E}[\|x_{t,i} - z_{t,i}\|^2]^{\frac{1}{2}} \mathbb{E}[\|x_{t,j} - z_{t,j}\|^2]^{\frac{1}{2}} \\ &= \frac{e^{-2t}}{(1 - e^{-2t})^2} \frac{1}{n(t)^2} \sum_{i,j=1}^{n(t)} \mathbb{E}_{x \sim p_t}[W_2(\text{Law}(z_{t,i}), p_{0|t}(\cdot|x))] \mathbb{E}_{x \sim p_t}[W_2(\text{Law}(z_{t,j}), p_{0|t}(\cdot|x))] \\ &\leq \frac{e^{-2t}}{(1 - e^{-2t})^2} \delta(t)^2. \end{aligned}$$

(4) follows from the estimation on  $N_1$  and  $N_2$ . □

### C.4 Proof of Theorem 1

In this section, we introduce the proof of our main convergence results, Theorem 1. Recall that in the convergence result in Theorem 1, three types of errors appear in the upper bound: the initialization

error, the discretization error and the score estimation error. Our proof compares the trajectory of  $\{\tilde{X}_t\}_{0 \leq t \leq T}$  that solves (8) and the trajectory of  $\{\bar{X}_t\}_{0 \leq t \leq T}$  that solves (2). We denote the path measures of  $\{\bar{X}_t\}_{0 \leq t \leq T}$  and  $\{\tilde{X}_t\}_{0 \leq t \leq T}$  by  $P^{p_T}$ , and  $Q^{\gamma^d}$ , respectively. Next, we introduce a high level idea on how the three types of errors are handled .

1. **Initialization error:** the initialization error comes from the comparison between  $\{\tilde{X}_t\}_{0 \leq t \leq T}$  and  $\{\tilde{X}_t^{p_T}\}_{0 \leq t \leq T}$ . To characterize this error, we introduce the intermediate process  $\{\tilde{X}_t^{p_T}\}_{0 \leq t \leq T}$

$$d\tilde{X}_t^{p_T} = (\tilde{X}_t^{p_T} + 2s(T - t_k, \tilde{X}_{t_k}^{p_T})dt + \sqrt{2}d\tilde{B}_t, \quad \tilde{X}_0 \sim p_T, \quad t \in [t_k, t_{k+1}), \quad (24)$$

in (24) and denote the path measure of  $\{\tilde{X}_t^{p_T}\}_{0 \leq t \leq T}$  by  $Q^{p_T}$ . Both processes are driven by the estimated scores and only the initial conditions are different. We factor out the initialization error from  $\text{KL}(p_\delta | q_{t_N})$  by the following argument:

$$\begin{aligned} \text{KL}(p_\delta | q_{t_N}) &= \text{KL}(p_\delta | q_{T-\delta}) \\ &\leq \int \ln \frac{dP^{p_T}}{dQ^{\gamma^d}} dP^{p_T} = \int \ln \left( \frac{dP^{p_T}}{dQ^{p_T}} \frac{dQ^{p_T}}{dQ^{\gamma^d}} \right) dP^{p_T} \\ &= \text{KL}(P^{p_T} | Q^{p_T}) + \int \ln \frac{dQ^{p_T}}{dQ^{\gamma^d}} dP^{p_T} \\ &= \text{KL}(P^{p_T} | Q^{p_T}) + \text{KL}(p_T | \gamma^d), \end{aligned}$$

where the inequality follows from the data processing inequality and the last identity follows from the fact that  $\frac{dQ^{p_T}}{dQ^{\gamma^d}} = \frac{dQ_0^{p_T}}{dQ_0^{\gamma^d}} = \frac{dp_T}{d\gamma^d}$ , which is true because the processes  $\{\tilde{X}_t\}_{0 \leq t \leq T}$  and  $\{\tilde{X}_t^{p_T}\}_{0 \leq t \leq T}$  have the same transition kernel function.  $\text{KL}(p_T | \gamma^d)$  is the initialization error and it is bounded based on (12) in Proposition C.1.

2. **Discretization error:** the discretization error arises from the evaluations of the scores at the discrete times. We factor out the discretization error from the  $\text{KL}(P^{p_T} | Q^{p_T})$  via the Girsanov's Theorem.

$$\begin{aligned} \text{KL}(P^{p_T} | Q^{p_T}) &\leq \sum_{k=0}^{N-1} \int_{t_k}^{t_{k+1}} \mathbb{E}_{P^{p_T}} [\|\nabla \ln p_{T-t}(\bar{X}_t) - s(T - t_k, \bar{X}_{t_k})\|^2] dt \\ &\lesssim \underbrace{\sum_{k=0}^{N-1} \int_{t_k}^{t_{k+1}} \mathbb{E}_{P^{p_T}} [\|\nabla \ln p_{T-t}(\bar{X}_t) - \nabla \ln p_{T-t_k}(\bar{X}_{t_k})\|^2] dt}_{\text{discretization error}} \\ &\quad + \underbrace{\sum_{k=0}^{N-1} \gamma_k \mathbb{E}_{P^{p_T}} [\|\nabla \ln p_{T-t_k}(\bar{X}_{t_k}) - s(T - t_k, \bar{X}_{t_k})\|^2]}_{\text{score estimation error}} \end{aligned}$$

We bound the discretization error term in the above equation by checking the dynamical properties of the process  $\{\nabla \ln p_t(\bar{X}_t)\}_{0 \leq t \leq T}$ . Similar approach was used in the analysis of denoising diffusion models, see [3]. For the sake of completeness, we include the proof in Appendix C.7.

3. **Score estimation error:** as discussed in the discretization error, the score estimation error is the accumulation of the  $L^2$ -error between the true score and score estimator at the time schedules  $\{T - t_k\}_{k=0}^{N-1}$ . In the analysis of denoising diffusion models, [5, 4, 3], it is usually assumed that such a  $L^2$  score error is small. In this paper, we consider to do sampling via the Reverse OU-process and score estimation. One of our main contribution is that we prove the  $L^2$  score error can be guaranteed small for the class of Monte Carlo score estimators given in Algorithm 2. The  $L^2$  score error upper bound is stated in Proposition 3.1.

*Proof of Theorem 1.* First we can decompose  $\text{KL}(p_\delta|q_{t_N})$  into summation of the three types of error.

$$\begin{aligned}
\text{KL}(p_\delta|q_{t_N}) &= \text{KL}(p_\delta|q_{T-\delta}) \\
&\leq \int \ln \frac{dP^{p_T}}{dQ^{\gamma^d}} dP^{p_T} = \int \ln \left( \frac{dP^{p_T}}{dQ^{p_T}} \frac{dQ^{p_T}}{dQ^{\gamma^d}} \right) dP^{p_T} \\
&= \text{KL}(P^{p_T}|Q^{p_T}) + \int \ln \frac{dQ^{p_T}}{dQ^{\gamma^d}} dP^{p_T} \\
&= \text{KL}(P^{p_T}|Q^{p_T}) + \text{KL}(p_T|\gamma^d) \\
&\leq \text{KL}(p_T|\gamma^d) + \sum_{k=0}^{N-1} \int_{t_k}^{t_{k+1}} \mathbb{E}_{P^{p_T}} [\|\nabla \ln p_{T-t}(\bar{X}_t) - s(T-t_k, \bar{X}_{t_k})\|^2] dt \\
&\lesssim \underbrace{\text{KL}(p_T|\gamma^d)}_{\text{initialization error}} + \underbrace{\sum_{k=0}^{N-1} \int_{t_k}^{t_{k+1}} \mathbb{E}_{P^{p_T}} [\|\nabla \ln p_{T-t}(\bar{X}_t) - \nabla \ln p_{T-t_k}(\bar{X}_{t_k})\|^2] dt}_{\text{discretization error}} \\
&\quad + \underbrace{\sum_{k=0}^{N-1} \gamma_k \mathbb{E}_{P^{p_T}} [\|\nabla \ln p_{T-t_k}(\bar{X}_{t_k}) - s(T-t_k, \bar{X}_{t_k})\|^2]}_{\text{score estimation error}},
\end{aligned}$$

where the first inequality follows from the data processing inequality. The second inequality follows from Girsanov's theorem and [6, Section 3.1]. According to Proposition C.1 and the assumption that  $T > 1$ , the initialization error satisfies

$$\text{KL}(p_T|\gamma^d) \leq \frac{1}{2} \frac{e^{-4T}}{1 - e^{-2T}} d + \frac{1}{2} e^{-2T} m_2^2 \lesssim (d + m_2^2) e^{-2T}. \quad (25)$$

According to Lemma 2, the discretization error satisfies

$$\begin{aligned}
&\sum_{k=0}^{N-1} \int_{t_k}^{t_{k+1}} \mathbb{E}_{P^{p_T}} [\|\nabla \ln p_{T-t}(\bar{X}_t) - \nabla \ln p_{T-t_k}(\bar{X}_{t_k})\|^2] dt \\
&\lesssim d \sum_{k=0}^N \frac{\gamma_k^2}{(1 - e^{-2(T-t_k)})^2} \\
&\quad + \sum_{k=0}^N \frac{e^{-2(T-t_k)} \gamma_k}{(1 - e^{-2(T-t_k)})^2} \left( \mathbb{E}[\text{trace}(\Sigma_{T-t_k}(X_{T-t_k}))] - \mathbb{E}[\text{trace}(\Sigma_{T-t_{k+1}}(X_{T-t_{k+1}}))] \right). \quad (26)
\end{aligned}$$

Reordering the summation in the second term and we have

$$\begin{aligned}
&\sum_{k=0}^N \frac{e^{-2(T-t_k)} \gamma_k}{(1 - e^{-2(T-t_k)})^2} \left( \mathbb{E}[\text{trace}(\Sigma_{T-t_k}(X_{T-t_k}))] - \mathbb{E}[\text{trace}(\Sigma_{T-t_{k+1}}(X_{T-t_{k+1}}))] \right) \\
&\leq \sum_{k=1}^{N-1} \left( \frac{e^{-2(T-t_k)} \gamma_k}{(1 - e^{-2(T-t_k)})^2} - \frac{e^{-2(T-t_{k-1})} \gamma_{k-1}}{(1 - e^{-2(T-t_{k-1})})^2} \right) \mathbb{E}[\text{trace}(\Sigma_{T-t_k}(X_{T-t_k}))] \\
&\quad + \frac{e^{-2T} \gamma_0}{(1 - e^{-2T})^2} \mathbb{E}[\text{trace}(\Sigma_T(X_T))] \\
&\lesssim \sum_{k=1}^{N-1} \frac{\gamma_k \gamma_{k-1}^2}{(1 - e^{-2(T-t_k)})^2 (1 - e^{-2(T-t_{k-1})})^2} \mathbb{E}[\text{trace}(\Sigma_{T-t_k}(X_{T-t_k}))] \\
&\quad + \frac{e^{-2T} \gamma_0}{(1 - e^{-2T})^2} \mathbb{E}[\text{trace}(\Sigma_T(X_T))]. \quad (27)
\end{aligned}$$

Recall that  $\Sigma_t(X_t) = \text{Cov}(X_0|X_t)$  for all  $0 \leq t \leq T$ . We have

$$\mathbb{E}[\text{trace}(\Sigma_t(X_t))] = \mathbb{E}[\text{Cov}(X_0|X_t)] \leq \mathbb{E}[\|X_0\|^2] \leq \mathbf{m}_2^2, \quad (28)$$

$$\begin{aligned}
\text{and } \mathbb{E}[\text{trace}(\Sigma_t(X_t))] &= \mathbb{E}[\text{Cov}(X_0|X_t)] = \mathbb{E}[\text{Cov}(X_0 - e^t X_t|X_t)] \\
&\leq \mathbb{E}[\mathbb{E}[\|X_0 - e^t X_t\|^2 | X_t]] \\
&= (e^{2t} - 1)d. \quad (29)
\end{aligned}$$

where the last identity follows from (13). (29) and (28) implies that  $\mathbb{E}[\text{trace}(\Sigma_t(X_t))] \lesssim (1 - e^{-2t})(d + m_2^2)$  for all  $0 \leq t \leq T$ . Therefore, from (26) and (27), the overall discretization error can be bounded as

$$\begin{aligned} & \sum_{k=0}^{N-1} \int_{t_k}^{t_{k+1}} \mathbb{E}_{P^{PT}} [\|\nabla \ln p_{T-t}(\bar{X}_t) - \nabla \ln p_{T-t_k}(\bar{X}_{t_k})\|^2] dt \\ & \lesssim \sum_{k=0}^{N-1} \frac{d\gamma_k^2}{(1 - e^{-2(T-t_k)})^2} + \sum_{k=1}^{N-1} \frac{(d + m_2^2)\gamma_k\gamma_{k-1}^2}{(1 - e^{-2(T-t_k)})(1 - e^{-2(T-t_{k-1})})^2} + \frac{m_2^2 e^{-2T}\gamma_0}{(1 - e^{-2T})^2}. \end{aligned} \quad (30)$$

Last, according to Proposition 3.1, the score estimation error satisfies

$$\begin{aligned} & \sum_{k=0}^{N-1} \gamma_k \mathbb{E}_{P^{PT}} [\|\nabla \ln p_{T-t_k}(\bar{X}_{t_k}) - s(T - t_k, \bar{X}_{t_k})\|^2] \\ & \leq \sum_{k=0}^{N-1} \frac{\gamma_k e^{-2(T-t_k)}}{(1 - e^{-2(T-t_k)})^2} \left( \delta(T - t_k)^2 + \frac{m_2^2}{n(T - t_k)} \right), \end{aligned} \quad (31)$$

and (9) follows from (25), (30) and (31).  $\square$

### C.5 Discussion on the Step-size

In this section, we first state error bounds of DDMC under different choices of step-size. Then we provide the detailed calculations. Last we compare our results in Theorem 1 to existing results on convergence of denoising diffusion models.

In the following discussion, we assume  $\delta(T - t_k)^2 \leq d\gamma_k e^{2(T-t_k)}$  and  $n(T - t_k) \geq \gamma_k^{-1} e^{-2(T-t_k)}$  for all  $k = 0, 1, \dots, N - 1$ , so that the score estimation error is dominated by the discretization error. For different choices of step-size, we discuss the parameter dependence of the error bound in (9) under the assumptions on  $\delta(t)$  and  $n(t)$ .

1. **constant step-size:** the constant step-size is widely considered in sampling algorithms and denoising diffusion generative models. It requires  $\gamma_k = \gamma$  for all  $0 \leq k \leq N - 1$ . Then

$$\text{KL}(p_\delta | q_{t_N}) \lesssim (d + m_2^2)e^{-2T} + \frac{(d+m_2^2)T^2}{N^2}(T + \delta^{-2}) + \frac{dT}{N}(T + \delta^{-1}).$$

2. **linear step-size:** the linear step-size is considered by [6] as an interpretation of the uniform discretization of a diffusion model with non-constant diffusion coefficient [52]. It requires  $t_k = T - (\delta + (N - k)\gamma)^2$  with  $\gamma = \frac{\sqrt{T-\delta}}{N}$  for all  $0 \leq k \leq N - 1$ . Then

$$\text{KL}(p_\delta | q_{t_N}) \lesssim (d + m_2^2)e^{-2T} + \frac{(d+m_2^2)T}{N^2}(T^2 + \delta^{-1}) + \frac{dT^{\frac{1}{2}}}{N}(T^{\frac{3}{2}} + \delta^{-\frac{1}{2}}).$$

3. **exponential-decay step-size:** the exponential-decay step-size is considered to be optimal in SGMs [6, 3]. It requires  $\gamma_k = \kappa \min(1, T - t_k)$  for some  $\kappa \in (0, 1)$ . Then

$$\text{KL}(p_\delta | q_{t_N}) \lesssim (d + m_2^2)e^{-2T} + \frac{(d+m_2^2)}{N^2}(T + \ln(\frac{1}{\delta}))^3 + \frac{d}{N}(T + \ln(\frac{1}{\delta}))^2.$$

The purple terms are denoting the discretization errors. For all of the above choices of step-size, the error bounds have the same linear dimension dependence and different dependence on the early stopping parameter  $\delta$ . Next, we provide a detailed calculation of these error bounds and a derivation of optimal  $\delta$ -dependence.

1. **constant step-size:** when  $\gamma_k = \gamma$  for all  $k = 0, 1, \dots, N - 1$ , we have  $T - t_k = \delta + (N - k)\gamma$  and

$$\frac{\gamma_k}{1 - e^{-2(T-t_k)}} = \begin{cases} \Theta(\gamma), & \text{if } T - t_k > 1, \\ \Theta(\frac{\gamma}{T - t_k}), & \text{if } T - t_k < 1. \end{cases}$$

Therefore

$$\begin{aligned}
& \sum_{k=1}^{N-1} \frac{(d+m_2^2)\gamma_k\gamma_{k-1}^2}{(1-e^{-2(T-t_k)})(1-e^{-2(T-t_{k-1})})^2} + \sum_{k=0}^{N-1} \frac{d\gamma_k^2}{(1-e^{-2(T-t_k)})^2} \\
&= \Theta\left(\sum_{1 < T-t_k < T} (d+m_2^2)\gamma^3 + d\gamma^2 + \sum_{\delta < T-t_k < 1} \frac{(d+m_2^2)\gamma^3}{(T-t_k)(T-t_{k-1}^2)} + \frac{d\gamma^2}{(T-t_k)^2}\right) \\
&= \Theta\left(\frac{(d+m_2^2)T^3}{N^2} + \frac{dT^2}{N} + (d+m_2^2)\gamma^2 \int_{\delta}^1 t^{-3} dt + d\gamma \int_{\delta}^1 t^{-2} dt\right) \\
&= \Theta\left(\frac{(d+m_2^2)T^3}{N^2} + \frac{dT^2}{N} + \frac{(d+m_2^2)T^2}{N^2\delta^2} + \frac{dT}{N\delta}\right).
\end{aligned}$$

2. **linear step-size:** when  $T-t_k = (\delta + (N-k)\gamma)^2$  with  $\gamma = \frac{\sqrt{T}-\delta}{N}$ , we have  $\gamma_k = (2\delta + (2N-2k-1)\gamma)\gamma = \Theta(\sqrt{T-t_k}\gamma)$  and

$$\frac{\gamma_k}{1-e^{-2(T-t_k)}} = \begin{cases} \Theta(\gamma\sqrt{T-t_k}), & \text{if } T-t_k > 1, \\ \Theta\left(\frac{\gamma}{\sqrt{T-t_k}}\right), & \text{if } T-t_k < 1. \end{cases}$$

Therefore

$$\begin{aligned}
& \sum_{k=1}^{N-1} \frac{(d+m_2^2)\gamma_k\gamma_{k-1}^2}{(1-e^{-2(T-t_k)})(1-e^{-2(T-t_{k-1})})^2} + \sum_{k=0}^{N-1} \frac{d\gamma_k^2}{(1-e^{-2(T-t_k)})^2} \\
&= \Theta\left(\sum_{1 < T-t_k < T} (d+m_2^2)\gamma^3\sqrt{T-t_k}(T-t_{k-1}) + d\gamma^2(T-t_{k-1})\right) \\
&+ \Theta\left(\sum_{\delta < T-t_k < 1} \frac{(d+m_2^2)\gamma^3}{\sqrt{T-t_k}(T-t_{k-1})} + \frac{d\gamma^2}{T-t_k}\right) \\
&= \Theta\left(\gamma^2(d+m_2^2) \int_1^T t dt + \gamma d \int_1^T t^{\frac{1}{2}} dt + \gamma^2(d+m_2^2) \int_{\delta}^1 t^{-2} dt + \gamma d \int_{\delta}^1 t^{-\frac{3}{2}} dt\right) \\
&= \Theta\left(\frac{(d+m_2^2)T^3}{N^2} + \frac{dT^2}{N} + \frac{(d+m_2^2)T}{N^2\delta} + \frac{dT^{\frac{1}{2}}}{N\delta^{\frac{1}{2}}}\right).
\end{aligned}$$

3. **exponential-decay step-size:** when  $\gamma_k = \kappa \min(1, T-t_k)$  with  $\kappa = \frac{T+\ln(1/\delta)}{N}$ , we have

$$\frac{\gamma_k}{1-e^{-2(T-t_k)}} = \begin{cases} \Theta(\kappa), & \text{if } T-t_k > 1, \\ \Theta(\kappa), & \text{if } T-t_k < 1. \end{cases}$$

Therefore

$$\begin{aligned}
& \sum_{k=1}^{N-1} \frac{(d+m_2^2)\gamma_k\gamma_{k-1}^2}{(1-e^{-2(T-t_k)})(1-e^{-2(T-t_{k-1})})^2} + \sum_{k=0}^{N-1} \frac{d\gamma_k^2}{(1-e^{-2(T-t_k)})^2} \\
&= \Theta\left(\sum_{1 < T-t_k < T} (d+m_2^2)\kappa^3 + d\kappa^2 + \sum_{\delta < T-t_k < 1} (d+m_2^2)\kappa^2\gamma_k(T-t_k)^{-1} + d\kappa\gamma_k(T-t_k)^{-1}\right) \\
&= \Theta\left((d+m_2^2)\kappa^3 N + d\kappa^2 N + \kappa^2(d+m_2^2) \int_{\delta}^1 t^{-1} dt + \kappa d \int_{\delta}^1 t^{-1} dt\right) \\
&= \Theta\left(\frac{(d+m_2^2)(T+\ln(1/\delta))^3}{N^2} + \frac{d(T+\ln(1/\delta))^2}{N}\right).
\end{aligned}$$

4. **Optimality of the exponential step-size:** assuming that  $\gamma_k = \Theta(\gamma_{k-1})$  for all  $k = 0, 1, \dots, N-1$ , the exponential step-size actually provides the optimal order estimation for the error terms. Noticing that the error terms all depend on the quantity  $\frac{\gamma_k}{1-e^{-2(T-t_k)}}$  which is of order

$$\frac{\gamma_k}{1-e^{-2(T-t_k)}} = \begin{cases} \Theta(\gamma_k), & \text{if } T-t_k > 1, \\ \Theta\left(\frac{\gamma_k}{T-t_k}\right), & \text{if } T-t_k < 1. \end{cases}$$

Therefore

$$\begin{aligned} & \sum_{k=1}^{N-1} \frac{(d + m_2^2)\gamma_k \gamma_{k-1}^2}{(1 - e^{-2(T-t_k)})(1 - e^{-2(T-t_{k-1})})^2} + \sum_{k=0}^{N-1} \frac{d\gamma_k^2}{(1 - e^{-2(T-t_k)})^2} \\ &= \Theta\left(\sum_{1 < T-t_k < T} (d + m_2^2)\gamma_k^3 + d\gamma_k^2 + \sum_{\delta < T-t_k < 1} \frac{(d + m_2^2)\gamma_k^3}{(T-t_k)^3} + \frac{d\gamma_k^2}{(T-t_k)^2}\right) \end{aligned}$$

Noticing that  $x \mapsto x^2$  and  $x \mapsto x^3$  are both convex functions on the domain  $x \in (0, \infty)$ . Since  $\sum_{1 < T-t_k < T} \gamma_k = T - 1$  is fixed, according to Jensen's inequality,  $\sum_{1 < T-t_k < T} \gamma_k^2$  and  $\sum_{1 < T-t_k < T} \gamma_k^3$  reach their minimum when  $\gamma_k$  are constant-valued for all  $k$  such that  $T - t_k > 1$ . Similarly, let  $\beta_k = \ln\left(\frac{T-t_k}{T-t_{k+1}}\right) \in (0, \infty)$ . Then  $\frac{\gamma_k}{T-t_k} = 1 - e^{-\beta_k}$  and  $\sum_{\delta < T-t_k < 1} \beta_k = \ln(1/\delta)$  is fixed. Since  $x \mapsto (1 - e^{-x})^2$  and  $x \mapsto (1 - e^{-x})^3$  are both convex functions on the domain  $x \in (0, \infty)$ , according to the Jensen's inequality,  $\sum_{\delta < T-t_k < 1} \frac{\gamma_k^2}{(T-t_k)^2}$  and  $\sum_{\delta < T-t_k < 1} \frac{\gamma_k^3}{(T-t_k)^3}$  reach their minimum when  $\beta_k$  are constant-valued for all  $k$  such that  $T - t_k < 1$ .

**Comparison to convergence results in denoising diffusion models.** (9) in Theorem 1 bounds the error of DDMC by I, II, III, which reflect the initialization error, the discretization error and the score estimation error, respectively. Assuming the score estimation error is small, (9) reduces to the same type of results that study the error bound for the denoising diffusion models (Algorithm 1). In [5], the discretization error is proved to be of order  $\mathcal{O}(d)$  assuming the score function is smooth along the trajectory and a constant step-size. In [4], they get rid of the trajectory smoothness assumption and prove a  $\mathcal{O}(d^2)$  discretization error bound with early-stopping. In [3], the discretization error bound is improved to  $\mathcal{O}(d)$  with early-stopping and exponential-decay step-size, and without the trajectory smoothness assumption. Compared to these works, our result in Theorem 1 also implies a  $\mathcal{O}(d)$  discretization error without the trajectory smoothness assumption and it applies to any choice of step-size with early stopping, as we discussed above. As shown in [3], the  $\mathcal{O}(d)$  is the optimal for the discretization error. Therefore, our results indicates that with early-stopping, the denoising diffusion model achieves the optimal linear dimension dependent error bound.

### C.6 Proofs of Section 3.3

To prove the query complexity of ZOD-MC, we first look at query complexity of Algorithm 3.

**Query complexity of Algorithm 3.** The query complexity of Algorithm 3 is essentially the number of proposals we need so that  $n(t)$  of them can be accepted. Intuitively, to get one sample accepted, the number of proposals we need is geometrically distributed with certain acceptance probability [9]. We state this formally in the following proposition, for which it suffices to assume a relaxation of the commonly used gradient-Lipschitz condition on the potential.

**Proposition C.3.** *Under Assumption 3.2, the expected number of proposals for obtaining  $n(t)$  many exact samples from  $p_{0|t}(\cdot|x)$  defined in Lemma 1 via Algorithm 3, is*

$$N(t) = n(t) \left( (L(e^{2t} - 1) + 1)^{\frac{d}{2}} \exp\left(\frac{1}{2} \frac{\|Lx^* - e^t x\|^2}{L(e^{2t} - 1) + 1}\right) \right).$$

**Remark 5.** *Our complexity bound in Proposition C.3 exponentially depends on the dimension. This is due the curse of dimensionality phenomenon in the rejection sampling: the acceptance rate and algorithm efficiency decreases significantly when the dimension increases.*

*Proof of Proposition C.3.* For each  $t \in [0, T]$ , the expected number of iterations in the rejection sampling to get one accepted sample is

$$\begin{aligned} M(t) &= \left( \int_{\mathbb{R}^d} e^{-V(z)+V(x^*)} \mathcal{N}(z; e^t x, (e^{2t} - 1)I_d) dz \right)^{-1} \\ &\leq \left( \int_{\mathbb{R}^d} \exp\left(-\frac{L}{2} \|z - x^*\|^2\right) \mathcal{N}(z; e^t x, (e^{2t} - 1)I_d) dz \right)^{-1} \\ &= (L(e^{2t} - 1) + 1)^{\frac{d}{2}} \exp\left(\frac{1}{2} \frac{\|Lx^* - e^t x\|^2}{L(e^{2t} - 1) + 1}\right). \end{aligned}$$

To get  $n(t)$  many samples, the expected number of iterations we need is  $N(t) = n(t)M(t)$ .  $\square$

**Query complexity of ZOD-MC.** The query complexity of ZOD-MC, denoted as  $\tilde{N}$ , is essentially the sum of query complexities in Proposition C.3 over the discretized time points, i.e.  $\tilde{N} = \sum_{k=0}^{N-1} N(T - t_k)$ .

*Proof of Corollary 3.1.* First, for  $\delta = \Theta\left(\min\left(\frac{\varepsilon_{W_2}^2}{d}, \frac{\varepsilon_{W_2}}{m_2}\right)\right)$ , it follows from Proposition C.1 that  $W_2(p, p_\delta) \leq \varepsilon_{W_2}$ .

Next, under the exponential-decay step size, according to Theorem 1, if we set  $n(T - t_k) = \gamma_k^{-1} e^{-2(T-t_k)}$ , then

$$\text{KL}(p_\delta | q_{t_N}) \lesssim (d + m_2^2) e^{-2T} + \frac{(d + m_2^2)}{N^2} \left(T + \ln\left(\frac{1}{\delta}\right)\right)^3 + \frac{d}{N} \left(T + \ln\left(\frac{1}{\delta}\right)\right)^2.$$

By choosing  $T = \frac{1}{2} \ln\left(\frac{d + m_2^2}{\varepsilon_{\text{KL}}}\right)$ ,  $N = \Theta\left(\max\left(\frac{(d + m_2^2)^{\frac{1}{2}} (T + \ln(\delta^{-1}))^{\frac{3}{2}}}{\varepsilon_{\text{KL}}^{\frac{1}{2}}}, \frac{d(T + \ln(\delta^{-1}))^2}{\varepsilon_{\text{KL}}}\right)\right)$  and  $\kappa = \Theta\left(\frac{T + \ln(\delta^{-1})}{N}\right)$ , we have  $\text{KL}(p_\delta | q_{t_N}) \lesssim \varepsilon_{\text{KL}}$ . Last, it follows from Proposition C.3 that

$$\tilde{N} \leq \sum_{k=0}^{N-1} \gamma_k^{-1} e^{-2(T-t_k)} \left( (L(e^{2(T-t_k)} - 1) + 1)^{\frac{d}{2}} \exp\left(\frac{1}{2} \frac{\|Lx^* - e^{T-t_k} x_k\|^2}{L(e^{2(T-t_k)} - 1) + 1}\right) \right).$$

By plugging in  $\delta, T, t_k$  and  $N$ , (10) is proved.  $\square$

## C.7 Side Lemmas

**Lemma 2.** Let  $\{\bar{X}_t\}_{0 \leq t \leq T}$  be the solution to (2). Then for any  $0 \leq k \leq N - 1$ , we have

$$\begin{aligned} & \int_{t_k}^{t_{k+1}} \mathbb{E}[\|\nabla \ln p_{T-t}(\bar{X}_t) - \nabla \ln p_{T-t_k}(\bar{X}_{t_k})\|^2] dt \\ & \lesssim \frac{d\gamma_k^2}{(1 - e^{-2(T-t_k)})^2} + \frac{e^{-2(T-t_k)} \gamma_k}{(1 - e^{-2(T-t_k)})^2} \left( \mathbb{E}[\text{trace}(\Sigma_{T-t_k}(X_{T-t_k}))] - \mathbb{E}[\text{trace}(\Sigma_{T-t_{k+1}}(X_{T-t_{k+1}}))] \right), \end{aligned}$$

where  $\{\Sigma_t(X_t)\}_{0 \leq t \leq T}$  is defined in Proposition C.2.

*Proof of Lemma 2.* For fixed  $s$ , consider the process  $\{\nabla \ln p_{T-t}(\bar{X}_t)\}_{0 \leq t \leq T}$ , denoted as  $\{L_t\}_{0 \leq t \leq T}$ , and a function  $E_{s,t} := \mathbb{E}[\|L_t - L_s\|^2]$ . It is shown by Itô's formula in [3, Lemma 3] that

$$dL_t = -L_t dt + \sqrt{2} \nabla^2 \ln q_{T-t}(\bar{X}_t) d\bar{B}_t, \quad (32)$$

and as a result, (32) implies that

$$dE_{s,t} = -2E_{s,t} dt - 2\mathbb{E}[\langle L_t - L_s, L_s \rangle] dt + 2\mathbb{E}[\|\nabla^2 \ln p_{T-t}(\bar{X}_t)\|_F^2] dt, \quad (33)$$

Apply (32) and Itô's formula again, we have

$$d\mathbb{E}[\langle L_t, L_s \rangle] = -\mathbb{E}[\langle L_t, L_s \rangle] dt \quad \implies \quad \mathbb{E}[\langle L_t, L_s \rangle] = e^{-(t-s)} \mathbb{E}[\|L_s\|^2].$$

Therefore (33) can be rewritten as

$$\frac{d}{dt} E_{s,t} = -2E_{s,t} + 2(1 - e^{-(t-s)}) \mathbb{E}[\|L_s\|^2] + 2\mathbb{E}[\|\nabla^2 \ln p_{T-t}(\bar{X}_t)\|_F^2]. \quad (34)$$

Let  $\{X_t\}_{0 \leq t \leq T}$  be the solution of (1). Since  $\bar{X}_t = X_{T-t}$  in distribution for all  $t \in [0, T]$ ,  $\mathbb{E}[\|L_s\|^2]$  and  $\mathbb{E}[\|\nabla^2 \ln p_{T-t}(\bar{X}_t)\|_F^2]$  can both be represented by the covariance matrix defined in Proposition C.2. It is proved in [3, Lemma 6] that

$$\begin{aligned} \mathbb{E}[\|L_s\|^2] &= \mathbb{E}[\|\nabla \ln p_{T-s}(\bar{X}_s)\|^2] \\ &= \frac{d}{1 - e^{-2(T-s)}} - \frac{e^{-2(T-s)}}{(1 - e^{-2(T-s)})^2} \mathbb{E}[\text{trace}(\Sigma_{T-s}(X_{T-s}))] \end{aligned} \quad (35)$$



and

$$\begin{aligned} \mathbb{E} \left[ \|\nabla^2 \ln p_{T-t}(\bar{X}_t)\|_F^2 \right] &= \frac{d}{(1 - e^{-2(T-t)})^2} - \frac{e^{-2(T-t)}}{2(1 - e^{-2(T-t)})^2} \frac{d}{dt} \left( \mathbb{E}[\text{trace}(\Sigma_{T-t}(X_{T-t}))] \right) \\ &\quad - \frac{2e^{-2(T-t)}}{(1 - e^{-2(T-t)})^3} \mathbb{E}[\text{trace}(\Sigma_{T-t}(X_{T-t}))]. \end{aligned} \quad (36)$$

Now we choose  $s = t_k$  in (34) and integrate from  $t_k$  to  $t$ . According to (35) and (36), we have

$$\begin{aligned} \frac{1}{2} e^{2t} E_{t_k, t_{k+1}} &= d \int_{t_k}^t \frac{e^{2u} - e^{u+t_k}}{1 - e^{-2(T-t_k)}} + \frac{e^{2u}}{(1 - e^{-2(T-u)})^2} du \\ &\quad - \frac{e^{-2(T-t_k)}}{(1 - e^{-2(T-t_k)})^2} \mathbb{E}[\text{trace}(\Sigma_{T-t_k}(X_{T-t_k}))] \int_{t_k}^t e^{2u} - e^{u+t_k} du \\ &\quad - \int_{t_k}^t \frac{e^{-2(T-u)+2u}}{2(1 - e^{-2(T-u)})^2} d\mathbb{E}[\text{trace}(\Sigma_{T-u}(X_{T-u}))] \\ &\quad - 2 \int_{t_k}^t \frac{e^{-2(T-u)+2u}}{(1 - e^{-2(T-u)})^3} \mathbb{E}[\text{trace}(\Sigma_{T-u}(X_{T-u}))] du \\ &= \frac{d}{2} \left( \frac{e^{2t} + e^{2t_k} - 2e^{t_k+t}}{1 - e^{-2(T-t_k)}} + \frac{e^{2t} - e^{2t_k}}{(1 - e^{-2(T-t_k)})(1 - e^{-2(T-t)})} \right) \\ &\quad - \frac{e^{-2(T-t_k)}}{(1 - e^{-2(T-t_k)})^2} \mathbb{E}[\text{trace}(\Sigma_{T-t_k}(X_{T-t_k}))] (e^{2t} + e^{2t_k} - 2e^{t_k+t}), \\ &\quad + \frac{e^{-2(T-t_k)+2t_k}}{(1 - e^{-2(T-t_k)})^2} \mathbb{E}[\text{trace}(\Sigma_{T-t_k}(X_{T-t_k}))] - \frac{e^{-2(T-t)+2t}}{(1 - e^{-2(T-t)})^2} \mathbb{E}[\text{trace}(\Sigma_{T-t}(X_{T-t}))] \\ &\quad - \int_{t_k}^t \frac{e^{-2(T-u)+2u}}{(1 - e^{-2(T-u)})^2} \mathbb{E}[\text{trace}(\Sigma_{T-u}(X_{T-u}))] du, \end{aligned}$$

where the last identity follows from integration by parts. According to Proposition C.2,  $t \mapsto \mathbb{E}[\text{trace}(\Sigma_{T-t}(X_{T-t}))]$  is positive and decreasing. Therefore, we have for all  $t \in [t_k, t_{k+1}]$ ,

$$\begin{aligned} E_{t_k, t} &\leq d \left( \frac{1 + e^{-2(t-t_k)} - 2e^{-(t-t_k)}}{1 - e^{-2(T-t_k)}} + \frac{1 - e^{-2(t-t_k)}}{(1 - e^{-2(T-t_k)})(1 - e^{-2(T-t)})} \right) \\ &\quad + 2 \frac{e^{-2(T-t_k)}}{(1 - e^{-2(T-t_k)})^2} (2e^{-(t-t_k)} - 1) \mathbb{E}[\text{trace}(\Sigma_{T-t_k}(X_{T-t_k}))] \\ &\quad - \left( \frac{2e^{-2(T-t)}}{(1 - e^{-2(T-t)})^2} + 2 \int_{t_k}^t \frac{e^{-2(T-t_k)-2(t-u)}}{(1 - e^{-2(T-t_k)})^2} du \right) \mathbb{E}[\text{trace}(\Sigma_{T-t}(X_{T-t}))]. \end{aligned}$$

Integrate again from  $t = t_k$  to  $t = t_{k+1}$ , we get

$$\begin{aligned} &\int_{t_k}^{t_{k+1}} E_{t_k, t} dt \\ &\leq \frac{d\gamma_k^3}{1 - e^{-2(T-t_k)}} + \frac{2d\gamma_k^2}{(1 - e^{-2(T-t_k)})^2} + \frac{2e^{-2(T-t_k)}}{(1 - e^{-2(T-t_k)})^2} (2 - 2e^{-\gamma_k} - \gamma_k) \mathbb{E}[\text{trace}(\Sigma_{T-t_k}(X_{T-t_k}))] \\ &\quad - \frac{e^{-2(T-t_k)}}{(1 - e^{-2(T-t_k)})^2} \left( 3\gamma_k + \frac{1}{2} e^{-\gamma_k} - \frac{1}{2} \right) \mathbb{E}[\text{trace}(\Sigma_{T-t_{k+1}}(X_{T-t_{k+1}}))] \\ &\leq \frac{3d\gamma_k^2}{(1 - e^{-2(T-t_k)})^2} + \frac{2e^{-2(T-t_k)}\gamma_k}{(1 - e^{-2(T-t_k)})^2} (\mathbb{E}[\text{trace}(\Sigma_{T-t_k}(X_{T-t_k}))] - \mathbb{E}[\text{trace}(\Sigma_{T-t_{k+1}}(X_{T-t_{k+1}}))]). \end{aligned}$$

□

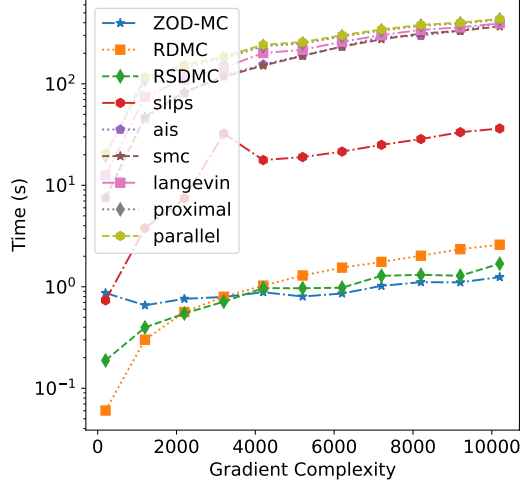


Figure 7: Wall clock of different methods as a function of oracle complexity for the 2D GMM in the main paper

## D More Experiments

### D.1 Samples from 2D GMM at different Oracle Complexities

We sample from a Gaussian Mixture model with 4 modes, the following summarizes the parameters of the GMM.

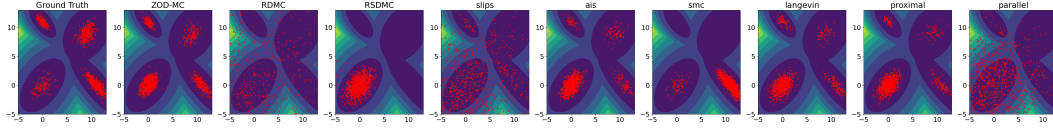
$$\begin{aligned}
 \text{Weights, } \mathbf{w} &: [0.1 \quad 0.2 \quad 0.3 \quad 0.4], \\
 \text{Means, } \boldsymbol{\mu}_k &: \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 11 \end{bmatrix}, \begin{bmatrix} 9 \\ 9 \end{bmatrix}, \begin{bmatrix} 11 \\ 0 \end{bmatrix}, \\
 \text{Covariances, } \boldsymbol{\Sigma}_k &: \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}, \begin{bmatrix} 0.3 & -0.2 \\ -0.2 & 0.3 \end{bmatrix}, \begin{bmatrix} 1 & 0.3 \\ 0.3 & 1 \end{bmatrix}, \begin{bmatrix} 1.2 & -1 \\ -1 & 1.2 \end{bmatrix}.
 \end{aligned}$$

We display the generated samples at different oracle complexities in Figures 8, 9, 10. Notice that the mode located at the origin holds less weight, and as the oracle complexity increases our method becomes better at sampling from other modes, as opposed to the corresponding baselines.

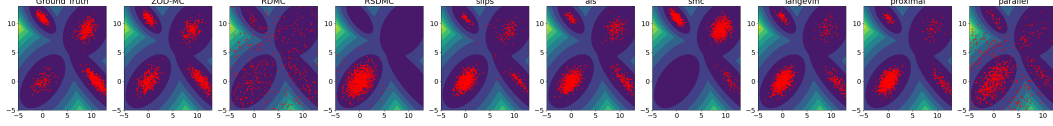
We detail the hyperparameters in Table 2 and the wall clock time of different methods in Figure 15.

Method	T	N	$\delta$	Step Size	N-MCMC	Num Steps	N-Chains
ZOD-MC	2	25	$5e-3$	-	$K$	-	-
RDMC	2	25	$5e-2$	0.01	1000	$K/1000$	-
RSDMC	2	25	$5e-2$	0.01	$K^{1/4}$	$K^{1/4}$	-
SLIPS	1	25	$6.62e-3$	Adaptive	1000	$K/1000$	-
AIS	-	-	-	Adaptive	-	M	512
SMC	-	-	-	Adaptive	-	M	512
Langevin	-	-	-	0.01	-	M	-
Proximal	-	-	-	1/5	-	M	-
Parallel	-	-	-	0.01	-	M	512

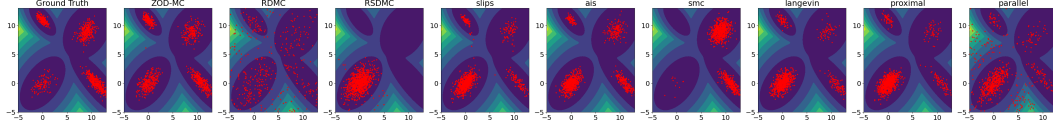
Table 2: Hyperparameters for Various Methods for the 2D GMM experiment.  $K$  means the current oracle complexity and  $M$  refers to a matched oracle complexity. For RSDMC we used 2 recursions per score evaluation.



(a) Generated samples at 200 oracle complexity per score evaluation

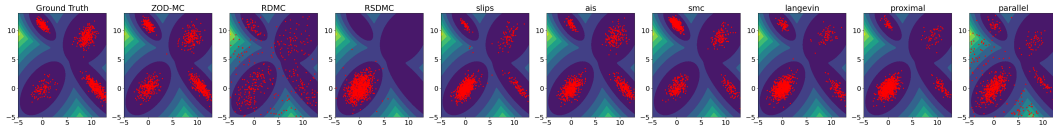


(b) Generated samples at 1200 oracle complexity per score evaluation

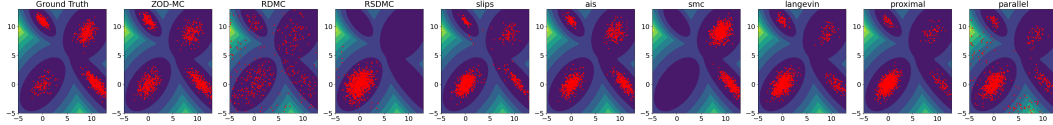


(c) Generated samples at 2200 oracle complexity per score evaluation

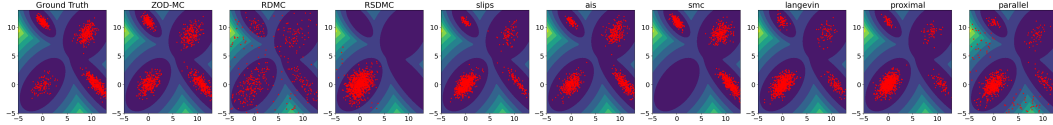
Figure 8: Generated Samples for GMM at different oracle complexity



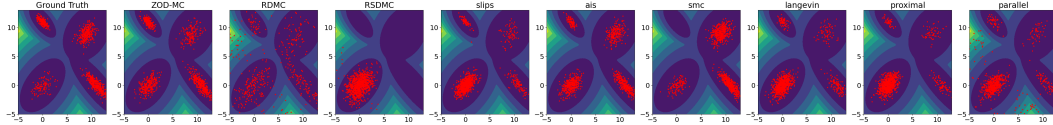
(a) Generated samples at 3200 oracle complexity per score evaluation



(b) Generated samples at 4200 oracle complexity per score evaluation

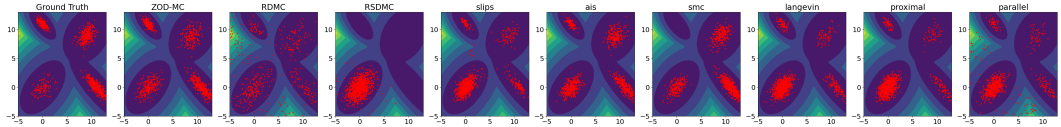


(c) Generated samples at 5200 oracle complexity per score evaluation

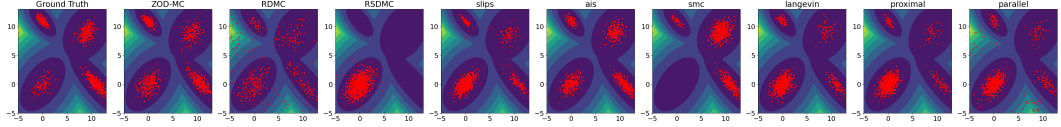


(d) Generated samples at 6200 oracle complexity per score evaluation

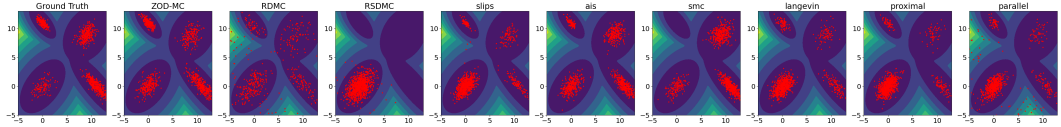
Figure 9: Generated Samples for GMM at different oracle complexity



(a) Generated samples at 7200 oracle complexity per score evaluation



(b) Generated samples at 8200 oracle complexity per score evaluation



(c) Generated samples at 9200 oracle complexity per score evaluation

Figure 10: Generated Samples for GMM at different oracle complexity

## D.2 Samples from Discontinuous 2D GMM at different Oracle Complexities

We display the generated samples at different oracle complexities in Figures 11, 12, 13. At the end we show the  $W_2$  and MMD for this example in Figure 14.

We detail the hyperparameters in Table 3.

Method	T	N	$\delta$	Step Size	N-MCMC	Num Steps	N-Chains
ZOD-MC	2	25	$5e-3$	-	$K$	-	-
RDMC	2	25	$5e-2$	0.01	1000	$K/1000$	-
RSDMC	2	25	$5e-2$	0.01	$K^{1/4}$	$K^{1/4}$	-
SLIPS	1	25	$6.62e-3$	Adaptive	1000	$K/1000$	-
AIS	-	-	-	Adaptive	-	M	512
SMC	-	-	-	Adaptive	-	M	512
Langevin	-	-	-	0.01	-	M	-
Proximal	-	-	-	1/5	-	M	-
Parallel	-	-	-	0.01	-	M	512

Table 3: Hyperparameters for Various Methods for the 2D Discontinuous GMM experiment.  $K$  means the current oracle complexity and  $M$  refers to a matched oracle complexity. For RSDMC we used 2 recursions per score evaluation.

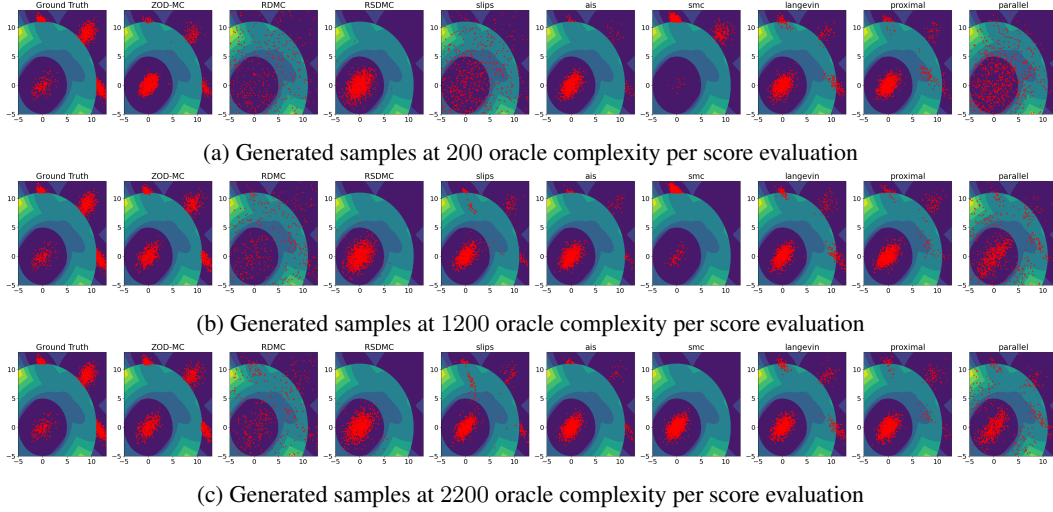
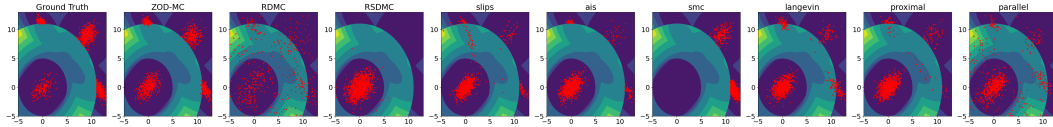
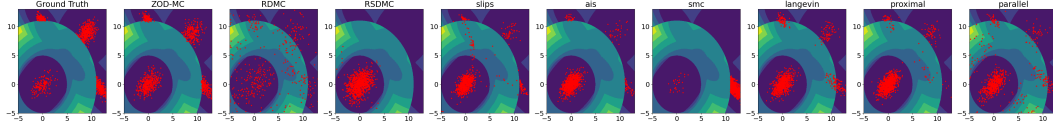


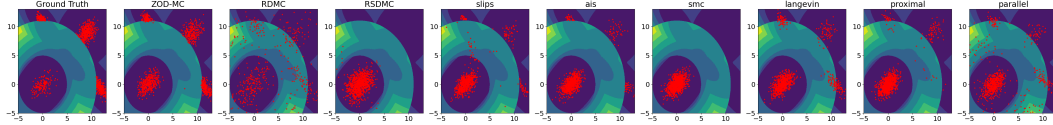
Figure 11: Generated Samples for discontinuous GMM at different oracle complexity



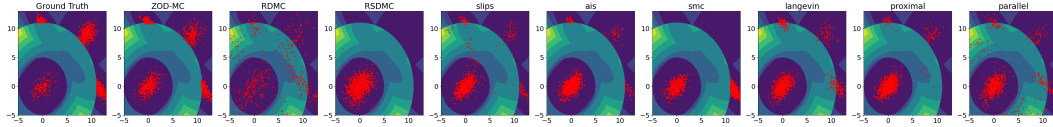
(a) Generated samples at 3200 oracle complexity per score evaluation



(b) Generated samples at 4200 oracle complexity per score evaluation

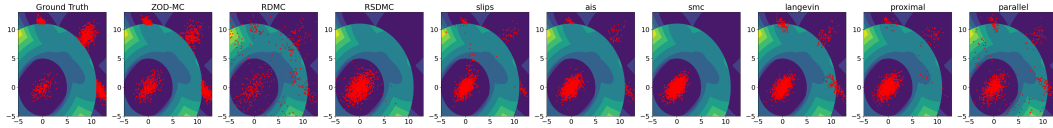


(c) Generated samples at 5200 oracle complexity per score evaluation

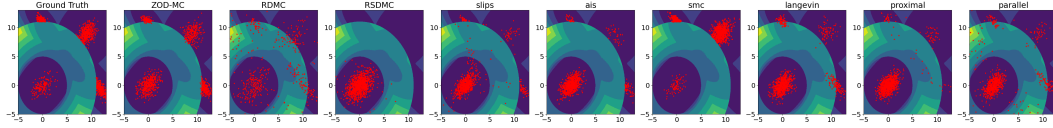


(d) Generated samples at 6200 oracle complexity per score evaluation

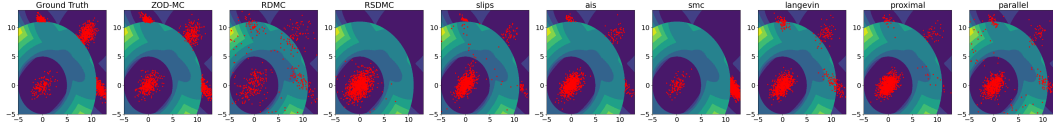
Figure 12: Generated Samples for discontinuous GMM at different oracle complexity



(a) Generated samples at 7200 oracle complexity per score evaluation



(b) Generated samples at 8200 oracle complexity per score evaluation



(c) Generated samples at 9200 oracle complexity per score evaluation

Figure 13: Generated Samples for discontinuous GMM at different oracle complexity

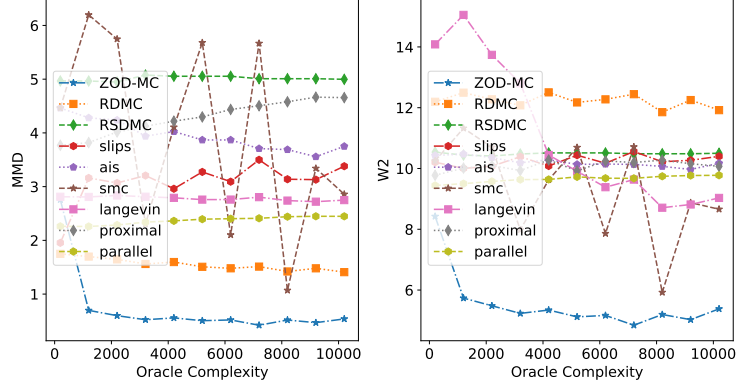


Figure 14:  $W_2$  and MMD at different oracle complexities for discontinuous potential

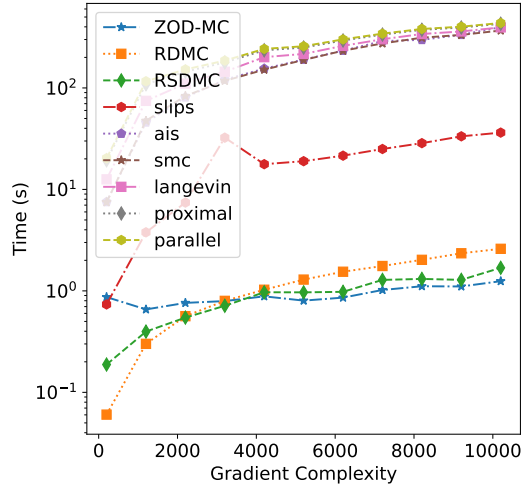


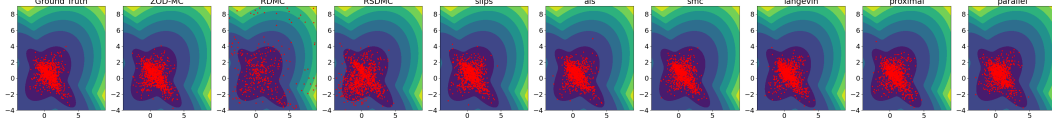
Figure 15: Wall clock of different methods as a function of oracle complexity

### D.3 Samples from different radius

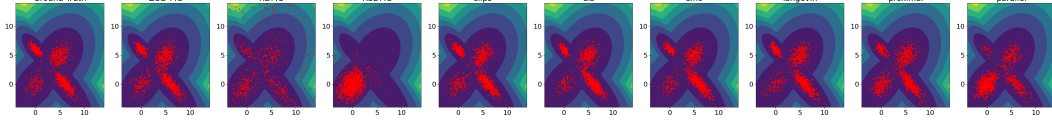
We display the generated samples at different radius in Figures 16, 18 and the hyperparameters in Table 4.

Method	T	N	$\delta$	Step Size	N-MCMC	Num Steps	N-Chains
ZOD-MC	10	50	$5e-3$	-	$K$	-	-
RDMC	2	50	$5e-2$	0.01	1000	$K/1000$	-
RSDMC	2	50	$5e-2$	0.01	$K^{1/4}$	$K^{1/4}$	-
SLIPS	1	50	$6.62e-3$	Adaptive	1000	$K/1000$	-
AIS	-	-	-	Adaptive	-	M	512
SMC	-	-	-	Adaptive	-	M	512
Langevin	-	-	-	0.01	-	M	-
Proximal	-	-	-	1/40	-	M	-
Parallel	-	-	-	0.01	-	M	512

Table 4: Hyperparameters for Various Methods for the 2D GMM experiment.  $K$  means the current oracle complexity and  $M$  refers to a matched oracle complexity.

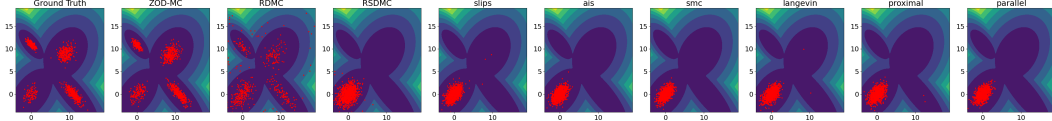


(a) Generated samples at  $R = 1$

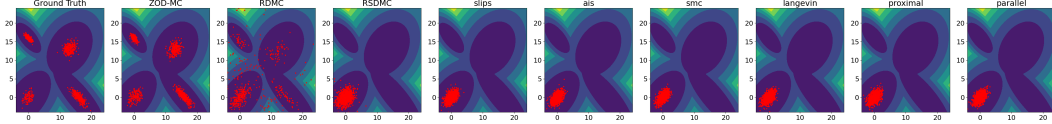


(b) Generated samples at  $R = 6$

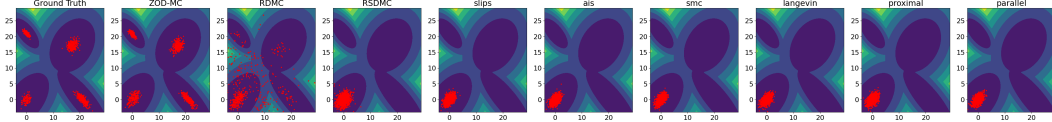
Figure 16: Generated samples at different radius



(a) Generated samples at  $R = 11$

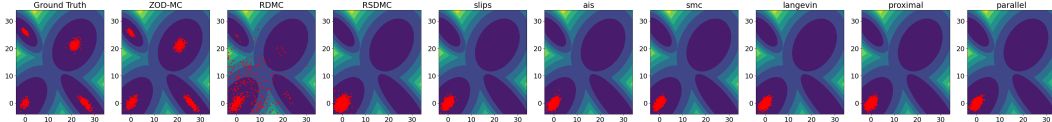


(b) Generated samples at  $R = 16$



(c) Generated samples at  $R = 21$

Figure 17: Generated samples at different radius



(a) Generated samples at  $R = 26$

Figure 18: Generated samples at different radius

#### D.4 Higher Dimensional Examples

**Score Error Approximation Details.** We use the following 5d Gaussian mixture to measure the error of the score approximation:

Coefficients,  $w : [0.25 \ 0.5 \ 0.25]$ ,

Means,  $\mu_k : \begin{bmatrix} -4 \\ -4 \\ -3 \\ -4 \\ -4 \end{bmatrix}, \begin{bmatrix} 4 \\ 4 \\ 2 \\ 4 \end{bmatrix}, \begin{bmatrix} -4 \\ -2 \\ 4 \\ -1 \end{bmatrix},$

Variances,  $\Sigma_k : \begin{bmatrix} 3 & 2 & 0 & 0 & 0 \\ 2 & 3 & 0 & 0 & 0 \\ 0 & 0 & 4 & 2 & 0 \\ 0 & 0 & 2 & 4 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 9 & 0 & 7 & 0 & 0 \\ 0 & 1 & 0 & 0.4 & 0 \\ 7 & 0 & 9 & 0 & 0 \\ 0 & 0.4 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0.4 & 0 & 0 & 0 \\ 0.4 & 1 & 0 & 0 & 0 \\ 0 & 0 & 4 & 3 & 0 \\ 0 & 0 & 3 & 4 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$



**Randomized Gaussian Mixtures** To generate the results in Figure 1b we proceed as follows. For a given dimension we take:

$$\mu = \frac{z}{\|z\|} \cdot 12$$

Where  $z \sim U[0, 1]^d$ , additionally we sample  $\sigma^2 \sim U[.3, 1.3]$ . We then consider the Gaussian target  $\mathcal{N}(\mu, \sigma^2 I)$ . We repeat this 5 times and create a Gaussian mixture with equally weighted modes. We plot the the 2d marginals of the target distribution as long as the generated samples.

### D.5 Muller Brown Potential Details

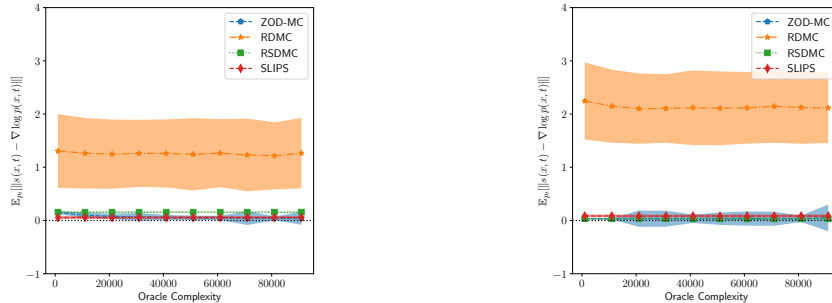
The potential is given by  $V(x, y) = \beta \cdot (V_m(x, y) + V_q(x, y))$

$$\begin{aligned} V_m(x, y) = & -170 \exp(-6.5(x+0.5)^2 + 11(x+0.5)(y-1.5) - 6.5(y-1.5)^2) \\ & - 100 \exp(-x^2 - 10(y-0.5)^2) + 15 \exp(0.7(x+1)^2 + 0.6(x+1)(y-1) + 0.7(y-1)^2) \\ & - 200 \exp(-(x-1)^2 - 10y^2), \end{aligned}$$

where  $V_m$  corresponds to the original Müller Brown and  $V_q(x, y) = 35.0136(x-x_c^*)^2 + 59.8399(y-y_c^*)^2$ , with  $(x_c^*, y_c^*)$  is approximately the minimizer at the center of the middle potential well, and  $V_q$  is a correction introduced so that the depths of all three wells are.

### D.6 Score error at $t = T$

One natural concern is that the sampling problem at  $t = T$  could be nearly as hard as sampling from the target distribution. Therefore, only a small number of samples could be accepted and the score error would be high. We display the score error at  $t = T$  to show that this is not necessarily the case.



(a) Score error at  $t = T = 4$ , for different diffusion samplers, we used  $t \approx 1$  for SLIPS. Results are for the 2d GMM in the main paper (b) Score error at  $t = T = 4$ , for different diffusion samplers, we used  $t \approx 1$  for SLIPS. Results are for the 5d GMM in the main paper

Figure 19: Score error at  $t = T$  for different target distributions

### D.7 Further details on number of accepted samples

We present the number of accepted samples from our rejection sampler as a function of time. Specifically we consider 1000 trajectories of the diffusion and for every intermediate step we sample 10K samples. We present the number of accepted samples in Table 5.

$t_0$	5.00	4.28	3.56	2.84	2.13	1.41	0.69	0.30	0.13	0.01
GMM	1.58	1.39	3.70	14.03	48.27	129.05	308.23	786.74	1299.30	2251.65
Mueller	1.01	1.20	2.62	9.79	37.25	145.42	447.96	945.72	1598.99	3129.97

Table 5: Comparison of GMM and Mueller values across different  $t$  values

## NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [NA].
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

**The checklist answers are an integral part of your paper submission.** They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes]" is generally preferable to "[No]", it is perfectly acceptable to answer "[No]" provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No]" or "[NA]" is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS paper checklist".**
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers.**

### 1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [Yes]

Justification: Section 1, Table 1.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

### 2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: Section 3.3, Remark 5 and the discussion after

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

### 3. Theory Assumptions and Proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: Main results: Proposition 3.1, Theorem 1, Corollary 3.1. Proofs are included in Appendix B.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

### 4. Experimental Result Reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Code is provided with instructions to run it. Scripts are provided so that a one line script can reproduce all the plots found in the paper.

Guidelines:

- The answer NA means that the paper does not include experiments.

- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

## 5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Code is provided with instructions to run it. Scripts are provided so that a one line script can reproduce all the plots found in the paper.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).

- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

## 6. Experimental Setting/Details

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: Yes, the target distributions are well described, furthermore every hyperparameter can be easily checked in the configuration files or the corresponding script for each experiment.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

## 7. Experiment Statistical Significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: In Figure 4b we present the standard deviation of the errors in the score approximation. In other experiments we didn't find it insightful to add these statistics.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

## 8. Experiments Compute Resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: Yes, we give this information at the beginning of section 4

Guidelines:

- The answer NA means that the paper does not include experiments.

- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

## 9. Code Of Ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: The research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

## 10. Broader Impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [NA]

Justification: Our paper studies the fundamental theory of sampling, with no specified applications.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

## 11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: Our paper poses no such risks.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

## 12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [NA]

Justification: We didn't use any dataset or asset from another party during these experiments.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, [paperswithcode.com/datasets](https://paperswithcode.com/datasets) has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

## 13. New Assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: Our paper does not release new assets.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

## 14. Crowdsourcing and Research with Human Subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: Our paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

**15. Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: Our paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.