Distillation of Discrete Diffusion through Dimensional Correlations

Satoshi Hayakawa[†][*] **Yuhta Takida**[‡] **Masaaki Imaizumi**[§]
Hiromi Wakaki[†] **Yuki Mitsufuji**[†][‡]
[†]Sony Group Corporation [‡]Sony AI [§]The University of Tokyo
[*]satoshi.a.hayakawa@sony.com

Abstract

Diffusion models have demonstrated exceptional performances in various fields of generative modeling. While they often outperform competitors including VAEs and GANs in sample quality and diversity, they suffer from slow sampling speed due to their iterative nature. Recently, distillation techniques and consistency models are mitigating this issue in continuous domains, but discrete diffusion models have some specific challenges towards faster generation. Most notably, in the current literature, correlations between different dimensions (pixels, locations) are ignored, both by its modeling and loss functions, due to computational limitations. In this paper, we propose "mixture" models in discrete diffusion that are capable of treating dimensional correlations while remaining scalable, and we provide a set of loss functions for distilling the iterations of existing models. Two primary theoretical insights underpin our approach: first, that dimensionally independent models can well approximate the data distribution if they are allowed to conduct many sampling steps, and second, that our loss functions enables mixture models to distill such many-step conventional models into just a few steps by learning the dimensional correlations. We empirically demonstrate that our proposed method for discrete diffusions work in practice, by distilling a continuous-time discrete diffusion model pretrained on the CIFAR-10 dataset.

1 Introduction

Diffusion models [40, 20, 44] have demonstrated excellent performance in generative modeling, particularly for continuous data such as images [34, 37, 38], audio [25, 8, 13], and video [18, 21, 5]. Recent advancements in diffusion models often outperform traditional generative models, such as variational autoencoders (VAEs) [24, 19, 53] and generative adversarial networks (GANs) [16], in terms of sample quality and the controllability of the generated results. Furthermore, diffusion models are not limited to learning continuous data; they can also be applied to discrete or categorical data with some straightforward modifications [22, 2] and offer a promising approach for discrete generative modeling [17, 29]. Such discrete diffusion models are the main topic of this paper.

A notable drawback of diffusion models, whether continuous or discrete, is that they suffer from slow sampling speed [50, 52], coming from the iterative nature of their sampling procedure. Although this feature allows many variants of conditional generations [9, 42, 3, 51], naive sampling schemes for diffusion models typically require a few thousands of sampling steps. In the continuous case, there have been various approaches to reduce the number of sampling steps. Earlier attempts include well-designed forward diffusion processes [41] and the use of fast solvers for stochastic/ordinary differential equations (SDEs/ODEs) [30, 31, 55]. Another notable approach is knowledge distillation, which compresses pretrained diffusion models into single- or few-step generative models [32, 39, 33, 54]. An emerging sub-family of distillation is the consistency-type models [45, 43, 23], which exploit the fact that generated samples via different paths from the same initial noise should coincide.

Machine Learning and Compression Workshop at NeurIPS 2024.

Applying such distillation methods to discrete diffusion models for reducing the number of sampling steps, however, is not straightforward. We claim that this is mainly because current methods are not designed to capture dimensional correlations in the data distributions, both in terms of modeling and loss functions. In this paper, we provide evidence for the claim and propose a method called **Di4C** (Distilling Discrete Diffusion through Dimensional Correlations) that captures dimensional correlations to reduce the number of sampling steps. Our contribution is summarized as follows:

- We show that the N-step denoising with the existing dimensionally independent discrete diffusion models can approximate data distribution in $\mathcal{O}(1/N)$ total variation error, together with the fact that there is a simple two-dimensional example where this bound cannot be improved (Theorem 1). This underpins the empirical effectiveness of existing discrete diffusion models *with many steps* and, at the same time, shows the importance of modeling dimensional correlations to reduce the number of sampling steps.
- To capture the aforementioned dimensional correlations, we propose Di4C, which distills a many-step discrete diffusion model (teacher) into a few-step model (student), by introducing a new set of loss functions compressing the iterative process of the teacher (Section 3.2) and a "mixture" modeling that can represent dimensional correlations (Section 3.3). In theory, we prove that the loss functions in Di4C can upper-bound the distance between the output distributions of the N-step teacher and the student with just one step (Theorem 2). This result, in combination with Theorem 1, provides an overall theoretical guarantee for Di4C.
- In numerical experiments with the CIFAR-10 dataset [26], we verify that Di4C can actually substantially improve quantitative evaluations in 10- and 20-step sampling compared to the pretrained teacher model of Campbell et al. [6].

Outline. Section 2 gives some preliminaries on discrete diffusion models and explains the dimensionality issue in discrete diffusion. We then explain the central idea of Di4C in Section 3 and show theoretical results in Section 4, which are partially described above as our contribution. We also provide experimental results with CIFAR-10 in Section 5. In Section 6, we conclude the paper with some discussions on its limitations and future work.

2 Preliminaries

Discrete diffusion models. Suppose we have a data distribution $q_0 := q_{\mathrm{data}}$ on the space $\mathcal{X}$. In diffusion models [40, 20], we consider a Markov process $(\boldsymbol{x}_t)_{0\le t\le T}$ with $\boldsymbol{x}_0 \sim q_0$ and $\boldsymbol{x}_T \sim q_T$, where the time t can be either discrete or continuous. In this paper, we follow the notational convention that $q_{t|s}$ and $q_{s,t}$ represent the true conditional and joint distributions defined by this Markov process, respectively. This process is designed so that the terminal distribution q_T is a tractable distribution. Our aim is to generate samples approximately from the conditional distribution $q_{0|T}(\cdot|\boldsymbol{x}_T)$ with $\boldsymbol{x}_T \sim q_T$, which is a generative model for q_{data}. To this end, we introduce a *model* or *denoiser*, which is represented as $p_{s|t}$ (for $s < t$), to approximate $q_{0|T}(\cdot|\boldsymbol{x}_T)$.

Our primary interest is in the discrete diffusion models [2, 6], where the space $\mathcal{X}$ is a finite set. In this case, a probability distribution p on $\mathcal{X}$ can be regarded as a function $p : \mathcal{X} \to \mathbb{R}$, and we will sometimes abuse the notation by treating p as just an ordinary function. We are given a finite set $\mathcal{S}$ and consider a diffusion process over the product space $\mathcal{X} = \mathcal{S}^D$ for a large D. Each state $\boldsymbol{x} \in \mathcal{X}$ can thus be written as $\boldsymbol{x} = (x^d)_{d=1}^D$, where x^d indicates the entry of $\boldsymbol{x}$ in the d-th dimension. Given a probability distribution $p = p(\boldsymbol{x})$ on $\mathcal{X}$, let $p^d = p^d(x^d)$ be its d-th marginal distribution, i.e., the distribution of x^d given $\boldsymbol{x} \sim p$. In order to enjoy scalability, the forward process is usually set to be factorized over dimensions, i.e., $q_{t|s}(\boldsymbol{x}_t|\boldsymbol{x}_s) = \prod_{d=1}^D q_{t|s}^d(x_t^d|x_s^d)$ holds for $s < t$ [17, 6].

Ignorance of dimensional correlations in discrete diffusion models. The common practices in modeling and training discrete diffusion models lead them to ignore the dimensional correlations within the data distribution. First, under the aforementioned problem setting, for the sake of scalability, the denoiser model is usually defined as a *product model* that satsifies

$$p_{s|t}(\boldsymbol{x}_s|\boldsymbol{x}_t) = \prod_{d=1}^D p_{s|t}^d(x_s^d|\boldsymbol{x}_t), \qquad s < t. \tag{1}$$

Namely, the distribution $p_{s|t}(\cdot|\boldsymbol{x}_t)$ is dimensionally independent. Second, the commonly used loss function $\mathbb{E}_{s<t,\boldsymbol{x}_0,\boldsymbol{x}_s,\boldsymbol{x}_t\sim q_{0,s,t}}\left[D_{\mathrm{KL}}(q_{s|0,t}(\cdot|\boldsymbol{x}_0,\boldsymbol{x}_t)\,\|\,p_{s|t}(\cdot|\boldsymbol{x}_t))\right]$ does not enforce dimensional correlations, since $q_{s|0,t}(\cdot|\boldsymbol{x}_0,\boldsymbol{x}_t)$ is dimensionally independent (see Section A for more details). The capability and limitation of the product modeling is mathematically shown in Theorem 1.

3 Di4C for distilling discrete diffusion models

This section describes our proposed method, Di4C. We first show that the *composition* of well-trained discrete diffusion models can represent the dimensional correlation in Section 3.1, and in the later sections we discuss how to distill the multi-step denoising of a teacher model into a student model that requires fewer steps. See Section B for more technical details of Di4C.

3.1 Composition of diffusion denoisers for inducing dimensional correlation

We introduce the notion of composition, which plays a significant role in representing the dimensional correlations to be learned. Consider two general conditional distributions $p(\boldsymbol{x}|\boldsymbol{y})$ and $\tilde{p}(\boldsymbol{y}|\boldsymbol{z})$ over finite sets. We define their composition as

$$p\circ\tilde{p}(\boldsymbol{x}|\boldsymbol{z}):=\mathbb{E}_{\boldsymbol{y}\sim\tilde{p}(\cdot|\boldsymbol{z})}[p(\boldsymbol{x}|\boldsymbol{y})]=\sum_{\boldsymbol{y}}p(\boldsymbol{x}|\boldsymbol{y})\tilde{p}(\boldsymbol{y}|\boldsymbol{z}),$$

where this definition can be extended to the continuous case in a straightforward way. Although this is just a convolution of two functions, it can be viewed as a composition of denoising operators in the context of diffusion models. Specifically, given a single-step denoiser $p_{s|t}$ and the finite timesteps $0=t_0<t_1<\cdots<t_N=T$, we typically use $p_{t_0|t_1}\circ\cdots\circ p_{t_{N-1}|t_N}(\cdot|\boldsymbol{x}_T)$ with the terminal noise $\boldsymbol{x}_T\sim q_T$ as a generative sampler.

Notably, the composition can serve as the source of dimensional correlation in discrete diffusion models. Even if one-step denoisers, $p_{s|u}$ and $p_{u|t}$ $(s<u<t)$, are dimensionally independent, their composition is generally not. Furthermore, Theorem 1 suggests that this composition applied to the conventional product model has enough capacity to capture the data distribution including dimensional correlation. Therefore, compressing the composition of well-trained denoisers into few-step sampling is a feasible way of learning dimensional correlation.

Let p^ψ be a pretrained teacher model with product structure and $p^\psi_{0|t_1}\circ\cdots\circ p^\psi_{t_{N-1}|t_N}$ be a sufficiently good approximation of $q_{0|T}$, where $0<t_1<\cdots<t_N=T$ are timesteps. Our aim is to train a student model p^θ that compresses the dimensional correlation learned by the teacher as

$$p^\theta_{0|t_n}\approx p^\psi_{0|t_1}\circ\cdots\circ p^\psi_{t_{n-1}|t_n},\qquad n=1,\dots,N. \tag{2}$$

To achieve this, we propose a set of loss functions to distill dimensional correlation represented by the compositions of a teacher model in Section 3.2, and we provide a way of modeling p^θ that is capable of representing dimensional correlations in Section 3.3.

3.2 Consistency for distilling dimensional correlation

We present a set of (two) loss functions that take dimensional correlation into account. Consider we are given a product teacher model, which is denoted as p^ψ. Let p^θ be a general student model (with enough expressive power; an example is given in Section 3.3) that we want to train based on p^ψ.

Distillation loss. We first introduce a *distillation loss*, which forces the student model to be consistent with the teacher model at time δ $(\ll T)$:

$$\mathcal{L}_{\mathrm{distil}}(\theta;\psi,r_\delta,\delta):=\mathbb{E}_{\boldsymbol{x}_\delta\sim r_\delta}\left[D_{\mathrm{KL}}(p^\psi_{0|\delta}(\cdot|\boldsymbol{x}_\delta)\,\|\,p^\theta_{0|\delta}(\cdot|\boldsymbol{x}_\delta))\right], \tag{3}$$

where r_δ $(\approx q_\delta)$ is a reference distribution over $\mathcal{X}$ at time δ and D_{KL} is the Kullback–Leibler (KL) divergence. We expect that a single teacher denoising step is enough to estimate $\boldsymbol{x}_0$ from $\boldsymbol{x}_\delta$; the dimensional correlation is mainly incorporated in the following consistency loss (see also Section B.1).

Consistency loss. We then propose a *consistency loss*, which allows the student model to learn the dimensional correlation represented by the composition of teacher denoisers:

$$\mathcal{L}_{\mathrm{consis}}(\theta; \psi, r_t, s, u, t) := \mathbb{E}_{\boldsymbol{x}_t \sim r_t}\Big[D_{\mathrm{KL}}(p^{\theta}_{s|u} \circ p^{\psi}_{u|t}(\cdot|\boldsymbol{x}_t) \,\|\, p^{\theta}_{s|t}(\cdot|\boldsymbol{x}_t))\Big], \tag{4}$$

where r_t is a reference distribution over $\mathcal{X}$ at time t approximating q_t. While this loss is not straightforward to compute, we discuss how to approximate it in practice with Monte Carlo or control variates in Section B.2. Note that the idea of mixing the teacher denoiser and student denoiser in $\mathcal{L}_{\mathrm{consis}}$ can also be found in the continuous-state setting regarding ODE trajectories [23, Fig. 3], but our loss is different in that we work on the compositions of conditional probabilities.

As reference distributions r_δ and r_t, we can either use q_t generated from data or the distribution obtained by applying multiple teacher denoising steps. See Section 4 for their roles and further theoretical guarantees on $\mathcal{L}_{\mathrm{distil}}$ and $\mathcal{L}_{\mathrm{consis}}$.

3.3 Mixture models for representing dimensional correlation

As an effective instance to represent correlated multivariate categorical distributions, we propose a *mixture model*. We define it as a family of conditional probability distributions that have the following representation for $s < t$:

$$p^{\theta}_{s|t}(\boldsymbol{x}_s|\boldsymbol{x}_t) = \mathbb{E}_{\lambda}\Big[p^{\theta}_{s|t}(\boldsymbol{x}_s|\boldsymbol{x}_t;\lambda)\Big], \qquad p^{\theta}_{s|t}(\boldsymbol{x}_s|\boldsymbol{x}_t;\lambda) = \prod_{d=1}^{D} p^{\theta,d}_{s|t}(x^d_s|\boldsymbol{x}_t;\lambda), \tag{5}$$

where λ is a random variable with an arbitrary distribution. This distribution can be viewed as a convex mixture of product model indexed by λ. Despite the fact that $p^{\theta}_{0|t}(\boldsymbol{x}_0|\boldsymbol{x}_t;\lambda)$ is dimensionally independent for each given point λ, this mixture representation is universal in the following sense:

Proposition 1. *For any probability distribution* p *over* $\mathcal{S}^D$*, there exist a probability distribution* π *and a family of product distributions* $p^{\lambda}(\boldsymbol{x}) = \prod_{d=1}^{D} p^{\lambda,d}(x^d)$ *indexed by* λ *satisfying* $\mathrm{p} = \mathbb{E}_{\lambda\sim\pi}\left[p^{\lambda}\right]$.

Indeed, we have $p = \mathbb{E}_{\boldsymbol{x}\sim p}[\delta_{\boldsymbol{x}}]$, where $\delta_{\boldsymbol{x}}$ is the delta distribution at $\boldsymbol{x}$, which is certainly a product distribution. Although the proof is not very informative, the assertion itself implies that the mixture model has sufficient expressive power to capture dimensional correlation. It should also be noted that sampling from this mixture model during the inference has almost no extra computational overhead compared to the conventional product model, since it just requires sampling of λ.

4 Theoretical analysis

In this section, we present an overall theoretical analysis on our distillation method. In Section 4.1, we show that the conventional product model (1) can approximate the data distribution if the model's marginal is perfectly trained and *given many steps*, which supports the empirical evidences of existing works. In Section 4.2, we prove that the proposed objective functions enable the many-step denoising with a teacher model to be distilled into a few-step student model, provided that the student model has enough expressive power. The former (Theorem 1) bounds the discrepancy between the data distribution and many-step teacher denoiser, and the latter (Theorem 2) provides a bound between the many-step teacher and (few-step) student denoiser. Thus, by combining these two, we can conclude that a few-step high-quality discrete diffusion model should be obtained if we apply our method to a well-trained teacher product model and an expressive student model (e.g., mixture model).

4.1 Product models with multi-step sampling can approximate data distribution

We first show that dimensionally independent denoisers with many steps are capable of approximately recovering the data distribution, which has already been empirically observed in existing studies. To consider varying the number of denoising steps, let us work on the continuous-time setting. Let $(\boldsymbol{x}_t)_{0\le t\le T}$ follow a continuous-time Markov chain over $[0, T]$ and the space $\mathcal{X} = \mathcal{S}^D$ with factorized forward process, i.e., $q_{t|s}(\boldsymbol{x}_t|\boldsymbol{x}_s) = \prod_{d=1}^{D} q^d_{t|s}(x^d_t|x^d_s)$ for $s < t$. See Section D for more details.

Theorem 1 shows the capability and limitation of a dimensionally independent sampling scheme called *analytical sampling* [46] (a.k.a. Tweedie τ-leaping [29, 36]), where we use a product-model denoiser

$p_{s|t}(\boldsymbol{x}_s|\boldsymbol{x}_t) = \prod_{d=1}^{D} p_{s|t}^d(x_s^d|\boldsymbol{x}_t)$ approximating the true marginal as $p_{s|t}^d(x_s^d|\boldsymbol{x}_t) \approx q_{s|t}^d(x_s^d|\boldsymbol{x}_t)$. Although commonly used, there has been only empirical evidences for the overall efficiency of this dimensionally independent method. Note that Campbell et al. [6] provides a guarantee for another dimensionally independent method called τ-leaping[1] [6].

Theorem 1 (N-step analytical sampling approximates data, informal)**.** *Let $q_{t|s}$ be forward transition probabilities that factorize as above and $p_{s|t}$ be a product model with the correct marginals, i.e., $p_{s|t}(\boldsymbol{x}_s|\boldsymbol{x}_t) = \prod_{d=1}^{D} q_{s|t}^d(x_s^d|\boldsymbol{x}_t)$ for $s < t$. Then, under some regularity conditions, given timesteps $t_i = iT/N$ for $i = 0, \ldots, N$, we have*

$$d_{\mathrm{TV}}\big(q_0, \mathbb{E}_{\boldsymbol{x}_t \sim q_T}\left[p_{t_0|t_1} \circ p_{t_1|t_2} \circ \cdots \circ p_{t_{N-1}|t_N}(\cdot|\boldsymbol{x}_T)\right]\big) = \mathcal{O}(1/N)\,, \tag{6}$$

where d_{TV} denotes the total variation distance.

Moreover, there is an example with $|\mathcal{S}| = D = 2$ such that the left-hand side of (6) *is lower-bounded by c/N with some constant $c > 0$ for sufficienty large N.*

See Theorem 3 for a formal version. Theorem 1 is important as it underpins the use of dimensionally parallel denoising given sufficient steps, which has been claimed as an advantage of discrete diffusion over autoregressive models whose sampling is sequential [29]. However, it still requires $\Omega(1/\epsilon)$ steps for having a uniform error bound ϵ, according to the latter half of the assertion. We show next that we can further reduce the number of steps with our loss functions, by distilling the distribution of an N-step teacher model into a few-step student model by learning dimensional correlations.

4.2 Our loss functions can distill multi-step denoising models

To consider our loss functions, let p^ψ and p^θ respectively be the teacher and student models given in Sections 3.1 & 3.2. The following statement gives a theoretical guarantee for using the proposed loss functions at appropriate time and distribution settings.

Theorem 2 (Di4C student approximates N-step teacher)**.** *Let $0 = t_0 < \cdots < t_N = T$ be timesteps and r_T be a probability distribution on $\mathcal{X}$. If we let $r_{t_n} = \mathbb{E}_{\boldsymbol{x}_T \sim r_T}\left[p^\psi_{t_n|t_{n+1}} \circ \cdots \circ p^\psi_{t_{N-1}|t_N}(\cdot|\boldsymbol{x}_T)\right]$ for each n, we have*

$$\begin{aligned} &d_{\mathrm{TV}}\left(r_0, \mathbb{E}_{\boldsymbol{x}_T \sim r_T}\left[p^\theta_{0|T}(\cdot|\boldsymbol{x}_T)\right]\right) \\ &\le \frac{1}{\sqrt{2}}\left(\mathcal{L}_{\mathrm{distil}}(\theta;\psi, r_{t_1}, t_1)^{1/2} + \sum_{n=1}^{N-1} \mathcal{L}_{\mathrm{consis}}(\theta;\psi, r_{t_{n+1}}, 0, t_n, t_{n+1})^{1/2}\right). \end{aligned} \tag{7}$$

See Section E.1 for the full proof. Note that the right-hand side of inequality (7) becomes zero (so does the left-hand side) if the student model perfectly learns the composition of the teacher as in (2), and so learning with these loss functions is feasible in theory if the student model has enough expressive power. Existing theoretical guarantees in consistency-based distillation of continuous-state diffusions typically discuss the case when consistency losses are exactly zero [45, 11, 27], and so our guarantee would be interesting in that it explicitly shows the relationships between the magnitude of loss functions and the upper bound of the total variation distance between the teacher and student.

Regarding the choice of r_t, we should take $r_T = q_T$ if we would like to combine Theorem 2 with Theorem 1 to evaluate Di4C's overall performance against the data distribution. For r_t with $t < T$, though we can generate samples $\boldsymbol{x}_t \sim r_t$ by using the teacher model, it might be expensive due to the multi-step inference required. Instead, we can use q_t if we have access to data, which is given by just one-step forward sampling from $q_{t|0}(\cdot|\boldsymbol{x}_0)$ with the data $\boldsymbol{x}_0 \sim q_0$. Since r_t is an approximation of q_t (Theorem 1), it would not harm the training quality as long as the teacher model is well-trained.

5 Experimental results

In numerical experiments, we adopted the same setting as Campbell et al. [6]: a continuous-time discrete-state Markov process with the CIFAR-10 image dataset, where the authors share a well-trained model checkpoint (which we use as a product teacher model p^ψ) that outperforms previous

[1] Although sharing the same name, τ-leaping and Tweedie τ-leaping are essentially different.

discrete-time discrete-state models such as Austin et al. [2]. As in the original paper, we worked directly with the discrete pixel channel values (0 to 255) on $32 \times 32 \times 3$ entries ($|\mathcal{S}| = 256$, $D = 3072$).

The teacher model p^ψ has the U-net architecture [20] tailored for discrete input-output, which is fed a time feature at each up-/down-sampling stage. We combined the teacher model with two sampling strategies (τ-leaping [6] and analytical sampling [46, 29]; see Sections 4.1 & G.1) and report their evaluation results as baselines in Table 1. We found the performance of the two sampling schemes to be very different: 40-step analytical sampling outperforms 1000-step τ-leaping in FID.

To obtain an architecture for our student mixture model (5), we slightly extended the teacher's architecture so that it accepts a conditioning with λ (following the uniform distribution over $[0, 1]$ in this experiment), by imitating the original implementation of time conditioning. In training, we fine-tuned from the teacher network parameters with additional zero-initialized subnetworks concerning λ. Note that the inference time of our student model is almost the same as that of the teacher model thanks to the architecture, and so the NFE is the dominant factor of the sampling speed among all the methods. See Section G.2 for details of the implementation and training.

Table 1: Comparison of models on CIFAR-10 dataset. NFE corresponds to the number of sampling steps. The Fréchet inception distance (FID $\downarrow$) against the training dataset and the inception score (IS $\uparrow$) are calculated using 50000 generated samples. *: reported values from Campbell et al. [6].

Method	NFE 10		NFE 20		NFE 40		NFE 1000	
	FID	IS	FID	IS	FID	IS	FID	IS
p^ψ + τ-leaping	-	-	-	-	315.75	1.66±0.01	8.10*	8.74*
p^ψ + analytical	32.61	7.59±0.10	12.36	8.55±0.13	**8.01**	**8.77±0.09**	-	-
p^θ (ours)	**20.64**	**8.29±0.13**	9.77	8.52±0.08	9.66	8.28±0.10	-	-
p^θ&p^ψcombined	25.54	8.00±0.11	**9.47**	**8.56±0.14**	8.02	8.43 ±0.11	-	-

The results are shown in Table 1. The "p^θ&p^ψ combined" model uses the same p^θ for just the first half (from noise to an intermediate state) of the denoising process and uses p^ψ with analytical sampling for the rest. We can see that p^θ substantially improves the metrics upon the teacher in 10-step sampling, while the gain of using our method gets smaller as NFE grows. The hybrid model interestingly beats other models in 20-step FID and shows almost the same 40-step FID with the teacher, while using the student solely gets worse. We hypothesize that this is because the true denoiser $q_{s|t}$ ($s < t$) becomes more "dimensionally independent" as $t - s$ or t is small. The former condition (small $t - s$) explains the worse performance gain of the mixture model as NFE grows, and the latter partially explains the effectiveness of using the combined model. However, we should further consider different forward diffusion and/or noise schedule to investigate it.

6 Conclusion

In this paper, as the current discrete diffusion models ignore the dimensional correlations that need to be incorporated for realizing few-step models, we proposed Di4C, a method for distilling pretrained discrete diffusion models. Di4C provides a set of loss functions for models that can capture dimensional correlations, an example of which is the mixture model. As a theoretical contribution, we proved that the existing discrete diffusion models with many steps can indeed recover the data distribution, even without modeling dimensional correlations. We also proved that such many-step models can be distilled into few-step ones, if we use the Di4C loss functions with a model that has enough expressive power, such as a mixture model. In numerical experiments with the CIFAR-10 dataset, we confirmed the efficiency of our framework in 10-step sampling.

However, there are still some problems to be solved. For example, although we can distill many-step models into one-step ones in theory (Theorem 2), our empirical results only show the improvements over the same few-step sampling. To address this point, we need to further optimize the architecture (mainly concerning λ) and training hyperparameters. It is also important to investigate how "dimensionally independent" $q_{s|t}$ is, as mentioned at the end of Section 5, and to clarify the situations in which dimensional correlations should be considered, rather than just using product models.

Acknowledgments

The authors would like to thank Chieh-Hsin Lai, Naoki Murata, Bac Nguyen, and Toshimitsu Uesaka for their constructive feedback.

References

[1] W. J. Anderson. *Continuous-time Markov chains: An applications-oriented approach*. Springer Science & Business Media, 2012.

[2] J. Austin, D. D. Johnson, J. Ho, D. Tarlow, and R. Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in Neural Information Processing Systems*, 34: 17981–17993, 2021.

[3] A. Bansal, H.-M. Chu, A. Schwarzschild, S. Sengupta, M. Goldblum, J. Geiping, and T. Goldstein. Universal guidance for diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 843–852, 2023.

[4] P. Blanchard, D. J. Higham, and N. J. Higham. Accurately computing the log-sum-exp and softmax functions. *IMA Journal of Numerical Analysis*, 41(4):2311–2330, 2021.

[5] A. Blattmann, T. Dockhorn, S. Kulal, D. Mendelevitch, M. Kilian, D. Lorenz, Y. Levi, Z. English, V. Voleti, A. Letts, V. Jampani, and R. Rombach. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023.

[6] A. Campbell, J. Benton, V. De Bortoli, T. Rainforth, G. Deligiannidis, and A. Doucet. A continuous time framework for discrete denoising models. *Advances in Neural Information Processing Systems*, 35:28266–28279, 2022.

[7] C. L. Canonne. A short note on an inequality between kl and tv. *arXiv preprint arXiv:2202.07198*, 2022.

[8] N. Chen, Y. Zhang, H. Zen, R. J. Weiss, M. Norouzi, and W. Chan. Wavegrad: Estimating gradients for waveform generation. In *International Conference on Learning Representations*, 2021.

[9] H. Chung, J. Kim, M. T. Mccann, M. L. Klasky, and J. C. Ye. Diffusion posterior sampling for general noisy inverse problems. In *International Conference on Learning Representations*, 2023.

[10] T. M. Cover and J. A. Thomas. *Elements of information theory*. John Wiley & Sons, second edition, 2006.

[11] G. Daras, Y. Dagan, A. Dimakis, and C. Daskalakis. Consistent diffusion models: Mitigating sampling drift by learning to be consistent. *Advances in Neural Information Processing Systems*, 36, 2024.

[12] S. Elfwing, E. Uchibe, and K. Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural networks*, 107:3–11, 2018.

[13] Z. Evans, C. Carr, J. Taylor, S. H. Hawley, and J. Pons. Fast timing-conditioned latent audio diffusion. In *International Conference on Machine Learning*, 2024.

[14] D. T. Gillespie. Approximate accelerated stochastic simulation of chemically reacting systems. *The Journal of chemical physics*, 115(4):1716–1733, 2001.

[15] P. Glasserman. *Monte Carlo methods in financial engineering*, volume 53. Springer, 2004.

[16] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014.

[17] S. Gu, D. Chen, J. Bao, F. Wen, B. Zhang, D. Chen, L. Yuan, and B. Guo. Vector quantized diffusion model for text-to-image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10696–10706, 2022.

[18] W. Harvey, S. Naderiparizi, V. Masrani, C. Weilbach, and F. Wood. Flexible diffusion modeling of long videos. In *Advances in Neural Information Processing Systems*, volume 35, pages 27953–27965, 2022.

[19] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner. beta-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2017.

[20] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[21] J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet. Video diffusion models. In *Advances in Neural Information Processing Systems*, volume 35, pages 8633–8646, 2022.

[22] E. Hoogeboom, D. Nielsen, P. Jaini, P. Forré, and M. Welling. Argmax flows and multinomial diffusion: Learning categorical distributions. In *Advances in Neural Information Processing Systems*, volume 34, pages 12454–12465, 2021.

[23] D. Kim, C.-H. Lai, W.-H. Liao, N. Murata, Y. Takida, T. Uesaka, Y. He, Y. Mitsufuji, and S. Ermon. Consistency trajectory models: Learning probability flow ode trajectory of diffusion. In *The Twelfth International Conference on Learning Representations*, 2024.

[24] D. P. Kingma and M. Welling. Auto-encoding variational Bayes. In *International Conference on Learning Representations*, 2014.

[25] Z. Kong, W. Ping, J. Huang, K. Zhao, and B. Catanzaro. Diffwave: A versatile diffusion model for audio synthesis. In *International Conference on Learning Representations*, 2020.

[26] A. Krizhevsky. Learning multiple layers of features from tiny images. Master's thesis, University of Toronto, 2009.

[27] C.-H. Lai, Y. Takida, T. Uesaka, N. Murata, Y. Mitsufuji, and S. Ermon. On the equivalence of consistency-type models: Consistency models, consistent diffusion models, and fokker-planck regularization. In *ICML 2023 Workshop on Structured Probabilistic Inference & Generative Modeling*, 2023.

[28] Y. Li, B. van Breugel, and M. van der Schaar. Soft mixture denoising: Beyond the expressive bottleneck of diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024.

[29] A. Lou, C. Meng, and S. Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution. In *Forty-first International Conference on Machine Learning*, 2024.

[30] C. Lu, Y. Zhou, F. Bao, J. Chen, C. Li, and J. Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. In *Advances in Neural Information Processing Systems*, volume 35, pages 5775–5787, 2022.

[31] C. Lu, Y. Zhou, F. Bao, J. Chen, C. Li, and J. Zhu. DPM-solver++: Fast solver for guided sampling of diffusion probabilistic models. *arXiv preprint arXiv:2211.01095*, 2022.

[32] E. Luhman and T. Luhman. Knowledge distillation in iterative generative models for improved sampling speed. *arXiv preprint arXiv:2101.02388*, 2021.

[33] C. Meng, R. Rombach, R. Gao, D. Kingma, S. Ermon, J. Ho, and T. Salimans. On distillation of guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14297–14306, 2023.

[34] A. Nichol, P. Dhariwal, A. Ramesh, P. Shyam, P. Mishkin, B. McGrew, I. Sutskever, and M. Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In *International Conference on Machine Learning*, 2021.

[35] C. J. Oates, M. Girolami, and N. Chopin. Control functionals for monte carlo integration. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 79(3):695–718, 2017.

[36] J. Ou, S. Nie, K. Xue, F. Zhu, J. Sun, Z. Li, and C. Li. Your absorbing discrete diffusion secretly models the conditional distributions of clean data. *arXiv preprint arXiv:2406.03736*, 2024.

[37] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[38] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. L. Denton, K. Ghasemipour, R. Gontijo Lopes, B. Karagol Ayan, T. Salimans, J. Ho, D. J. Fleet, and M. Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. In *Advances in neural information processing systems*, volume 35, pages 36479–36494, 2022.

[39] T. Salimans and J. Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2022.

[40] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015.

[41] J. Song, C. Meng, and S. Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021.

[42] J. Song, Q. Zhang, H. Yin, M. Mardani, M.-Y. Liu, J. Kautz, Y. Chen, and A. Vahdat. Loss-guided diffusion models for plug-and-play controllable generation. In *International Conference on Machine Learning*, pages 32483–32498. PMLR, 2023.

[43] Y. Song and P. Dhariwal. Improved techniques for training consistency models. In *International Conference on Learning Representations*, 2023.

[44] 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.

[45] Y. Song, P. Dhariwal, M. Chen, and I. Sutskever. Consistency models. In *International Conference on Machine Learning*, pages 32211–32252. PMLR, 2023.

[46] H. Sun, L. Yu, B. Dai, D. Schuurmans, and H. Dai. Score-based continuous-time discrete diffusion models. In *The Eleventh International Conference on Learning Representations*, 2023.

[47] A. van den Oord, O. Vinyals, and K. Kavukcuoglu. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

[48] H. von Eitzen. Continuous function with continuous one-sided derivative. Mathematics Stack Exchange, 2014. URL `https://math.stackexchange.com/q/975094`. (version: 2014-10-16).

[49] R. J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8:229–256, 1992.

[50] Z. Xiao, K. Kreis, and A. Vahdat. Tackling the generative learning trilemma with denoising diffusion gans. In *International Conference on Learning Representations*, 2022.

[51] J. Yu, Y. Wang, C. Zhao, B. Ghanem, and J. Zhang. Freedom: Training-free energy-guided conditional diffusion model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 23174–23184, 2023.

[52] Q. Zhang and Y. Chen. Fast sampling of diffusion models with exponential integrator. In *International Conference on Learning Representations*, 2023.

[53] S. Zhao, J. Song, and S. Ermon. InfoVAE: Balancing learning and inference in variational autoencoders. In *AAAI Conference on Artificial Intelligence*, pages 5885–5892, 2019.

[54] H. Zheng, W. Nie, A. Vahdat, K. Azizzadenesheli, and A. Anandkumar. Fast sampling of diffusion models via operator learning. In *International conference on machine learning*, pages 42390–42402. PMLR, 2023.

[55] K. Zheng, C. Lu, J. Chen, and J. Zhu. DPM-solver-v3: Improved diffusion ode solver with empirical model statistics. In *Advances in Neural Information Processing Systems*, volume 36, pages 55502–55542, 2023.

A Ignorance of dimensional correlations in discrete diffusion models

In this section, we discuss the limitations introduced in the latter half of Section 2 in more detail, namely, the fact that the existing discrete diffusion models as well as the loss functions ignore dimensional correlations appearing in the data distributions.

Model's ignorance of dimensional correlation. The product model

$$p_{s|t}(\boldsymbol{x}_s|\boldsymbol{x}_t) = \prod_{d=1}^{D} p_{s|t}^d(x_s^d|\boldsymbol{x}_t), \qquad s < t,$$

is common if not particularly highlighted [6, Section G], due to the combinatorial explosion of the product discrete state. Indeed, adopting a product model significantly reduces the output length from $\mathcal{O}(D^S)$ to $\mathcal{O}(DS)$ at the cost of representational capacity. This limited expressive power can be crucial for considering few-step discrete diffusion models. As an extreme example, consider doing one-step denoising in the case of absorbing-state diffusion [2]; there is no chance we can approximate a complex distribution by one step when $\boldsymbol{x}_T$ is a completely masked sentence (i.e., following a delta distribution) and $p_{0|T}(\cdot|\boldsymbol{x}_T)$ is dimensionally independent. See Section G.1 for more examples. To mitigate this issue, we propose a class of model that is capable of treating dimensional correlation in Section 3.3.

The issue caused by the dimensionally independent modeling has also been pointed out in the context of continuous-state diffusion models [28]. However, such a modeling in the continuous case (i.e., modeling the added noise as a unimodal Gaussian) is empirically less problematic, partially due to the use of ℓ^2 loss and the existence of probability flow ODEs.

Loss function's ignorance of dimensional correlation. Another potential factor making the learning of dimensional correlation infeasible in discrete diffusion models is that the existing loss function is not well prepared for learning dimensional correlation. This common loss is derived as variational lower bound (VLB) of log-likelihood, which is given by

$$\alpha\mathbb{E}_{\boldsymbol{x}_0,\boldsymbol{x}_\delta\sim q_{0,\delta}}\left[-\log p_{0|\delta}(\boldsymbol{x}_0|\boldsymbol{x}_\delta)\right] + \beta\mathbb{E}_{s<t,\boldsymbol{x}_0,\boldsymbol{x}_s,\boldsymbol{x}_t\sim q_{0,s,t}}\left[D_{\mathrm{KL}}(q_{s|0,t}(\cdot|\boldsymbol{x}_0,\boldsymbol{x}_t) \,\|\, p_{s|t}(\cdot|\boldsymbol{x}_t))\right]$$

with $\alpha, \beta > 0$ and $0 < \delta \ll 1$, where D_{KL} denotes the Kullback–Leibler (KL) divergence (see Section C). It usually does not force $p_{s|t}$ for $t > \delta$ to be dimensionally correlated, due to the product structure of $q_{s|0,t}$ for $s < t$ given by

$$q_{s|0,t}(\boldsymbol{x}_s|\boldsymbol{x}_0,\boldsymbol{x}_t) = \frac{q_{s|0}(\boldsymbol{x}_s|\boldsymbol{x}_0)q_{t|0,s}(\boldsymbol{x}_t|\boldsymbol{x}_0,\boldsymbol{x}_s)}{q_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)} = \prod_{d=1}^{D}\frac{q_{s|0}(x_s^d|x_0^d)q_{t|s}(x_t^d|x_s^d)}{q_{t|0}(x_t^d|x_0^d)}.$$

An exeption is the auxiliary loss $\mathbb{E}_{\boldsymbol{x}_0,\boldsymbol{x}_t\sim q_{0,t}}[-\log p_{0|t}(\boldsymbol{x}_0|\boldsymbol{x}_t)]$ sometimes added with a very small coefficient [2, 17], which is still not enough for learning the correlation in practice.

In the continuous-time score-based discrete diffusion, we only need the marginal $p_{s|t}^d(\cdot|\boldsymbol{x}_t)$ or its equivalent for computing the infinitesimal transition rate [6, 46]. Therefore, the existing training pipelines cannot learn the dimensional correlation. Note that using a product model with these loss functions is "scalable" in the sense that $D_{\mathrm{KL}}(q_{s|0,t}(\cdot|\boldsymbol{x}_0,\boldsymbol{x}_t) \,\|\, p_{s|t}(\cdot|\boldsymbol{x}_t))$ becomes just the sum of KL divergence over the dimensions.

We addressed this challenge by introducing $\mathcal{L}_{\mathrm{consis}}$ in Section 3.2, which allows a general model to learn the dimensional correlation produced by the "composition" discussed Section 3.1.

B Training techniques for Di4C

In this section, we review the novel loss functions of Di4C and the mixture model given in Section 3.2 from an algorithmic perspective, and provide a set of techniques to stably train it. Specifically, we introduce techniques to make the computation of the loss functions scalable through Monte Carlo integration and control variate methods.

Before going into the details of the training techniques, we introduce two auxiliary loss functions, which we can use in addition to $\mathcal{L}_{\mathrm{distil}}$ and $\mathcal{L}_{\mathrm{consis}}$ for practical improvements. One is the *datapoint* *loss* that directly computes the negative log-likelihood with respect to the data distribution [e.g., 2, Eq. 5], which we can use when we have access to data q_0:

$$\mathcal{L}_{\mathrm{data}}(\theta; t) := \mathbb{E}_{(\boldsymbol{x}_0, \boldsymbol{x}_t)\sim q_{0,t}}\left[-\log p^{\theta}_{0|t}(\boldsymbol{x}_0|\boldsymbol{x}_t)\right]. \tag{8}$$

The other is the following *marginal loss*, which is easier to compute, under the assumption that the teacher model sufficiently learns the true marginal, i.e., $p^{\psi,d}_{0|t} \approx q^{d}_{0|t}$:

$$\mathcal{L}_{\mathrm{marginal}}(\theta; \psi, r_t, t) := \mathbb{E}_{\boldsymbol{x}_t\sim r_t}\left[\sum_{d=1}^{D} D_{\mathrm{KL}}(p^{\psi,d}_{0|t}(\cdot|\boldsymbol{x}_t) \,\|\, p^{\theta,d}_{0|t}(\cdot|\boldsymbol{x}_t))\right]. \tag{9}$$

B.1 Surrogate of distillation loss

Since the exact evaluation of $\mathcal{L}_{\mathrm{distil}}$ with a mixture model seems intractable, we consider an upper bound of $\tilde{\mathcal{L}}_{\mathrm{distil}}$ as a practical alternative:

$$\begin{aligned}\mathcal{L}_{\mathrm{distil}}(\theta; \psi, r_\delta, \delta) &= \mathbb{E}_{\boldsymbol{x}_\delta\sim r_\delta}\left[D_{\mathrm{KL}}(p^{\psi}_{0|\delta}(\cdot|\boldsymbol{x}_\delta) \,\|\, \mathbb{E}_\lambda[p^{\theta}_{0|\delta}(\cdot|\boldsymbol{x}_\delta; \lambda)])\right] \\ &\le \mathbb{E}_{\boldsymbol{x}_\delta\sim r_\delta}\mathbb{E}_\lambda\left[D_{\mathrm{KL}}(p^{\psi}_{0|\delta}(\cdot|\boldsymbol{x}_\delta)) \,\|\, p^{\theta}_{0|\delta}(\cdot|\boldsymbol{x}_\delta; \lambda)\right] \\ &\le \mathbb{E}_{\lambda, \boldsymbol{x}_\delta\sim r_\delta}\left[\sum_{d=1}^{D} D_{\mathrm{KL}}(p^{\psi,d}_{0|\delta}(\cdot|\boldsymbol{x}_\delta) \,\|\, p^{\theta,d}_{0|\delta}(\cdot|\boldsymbol{x}_\delta; \lambda))\right] =: \tilde{\mathcal{L}}_{\mathrm{distil}}(\theta; \psi, r_\delta, \delta).\end{aligned}$$

Here, the inequality is given by the convexity of KL divergence (see Proposition 3). The upper bound $\tilde{\mathcal{L}}_{\mathrm{distil}}$ (and then $\mathcal{L}_{\mathrm{distil}}$) becomes zero if the student denoiser coincides with the teacher for the time interval $[0, \delta]$, regardless of λ. Therefore, the use of this upper bound is feasible if p^θ has enough expressive power.

B.2 Surrogate of consistency loss

We consider $\mathcal{L}_{\mathrm{consis}}$ in this section. As $p^{\theta}_{s|u}$ is more "reliable" than $p^{\theta}_{s|t}$ (since $s < u < t$), we only consider the gradient of $\mathcal{L}_{\mathrm{consis}}$ concerning $p^{\theta}_{s|t}$ and ignore the gradient coming from $p^{\theta}_{s|u}$. Therefore, we conduct stochastic gradient descent on θ with the loss

$$D_{\mathrm{KL}}(p^{\mathrm{sg}(\theta)}_{s|u} \circ p^{\psi}_{u|t}(\cdot|\boldsymbol{x}_t) \,\|\, p^{\theta}_{s|t}(\cdot|\boldsymbol{x}_t)) = H(p^{\mathrm{sg}(\theta)}_{s|u} \circ p^{\psi}_{u|t}(\cdot|\boldsymbol{x}_t), p^{\theta}_{s|t}(\cdot|\boldsymbol{x}_t)) + \mathit{const.}, \tag{10}$$

where $\mathrm{sg}(\cdot)$ is the stop-gradient operator [47] and $H(p, q) = \mathbb{E}_{\boldsymbol{x}\sim p}[-\log q(\boldsymbol{x})]$ is the cross entropy between p and q. We hereby ignore the constant term in (10) and consider how to efficiently compute the cross entropy term.

Most naively, by using finite samples $\boldsymbol{x}^{(1)}_s, \ldots, \boldsymbol{x}^{(M)}_s \sim_{\mathrm{iid}} p^{\mathrm{sg}(\theta)}_{s|u} \circ p^{\psi}_{u|t}(\cdot|\boldsymbol{x}_t)$ and $\lambda_1, \ldots, \lambda_N \sim_{\mathrm{iid}} \lambda$, we can approximate this cross entropy by two-fold Monte Carlo:

$$\begin{aligned}&H(p^{\mathrm{sg}(\theta)}_{s|u} \circ p^{\psi}_{u|t}(\cdot|\boldsymbol{x}_t), p^{\theta}_{s|t}(\cdot|\boldsymbol{x}_t)) \\ &\quad\approx -\frac{1}{M}\sum_{j=1}^{M}\log p^{\theta}_{s|t}(\boldsymbol{x}^{(j)}_s|\boldsymbol{x}_t) \approx -\frac{1}{M}\sum_{j=1}^{M}\log\left(\frac{1}{N}\sum_{i=1}^{N} p^{\theta}_{s|t}(\boldsymbol{x}^{(j)}_s|\boldsymbol{x}_t; \lambda_i)\right).\end{aligned} \tag{11}$$

Although the value of each $p^{\theta}_{s|t}(\boldsymbol{x}^{(j)}_s|\boldsymbol{x}_u; \lambda_i) = \prod_{d=1}^{D} p^{\theta,d}_{s|t}(x^{(j),d}_s|\boldsymbol{x}_u; \lambda_i)$ can be extremely small due to the D-fold product, we can exploit the log-sum-exp structure:

$$\log\left(\sum_{i=1}^{N} p^{\theta}_{s|t}(\boldsymbol{x}^{(j)}_s|\boldsymbol{x}_t; \lambda_i)\right) = \underbrace{\log\left(\sum_{i=1}^{N}\exp\right.}_{\text{log-sum-exp}}\left.\left(\sum_{d=1}^{D}\log p^{\theta,d}_{s|t}(x^{(j),d}_s|\boldsymbol{x}_t; \lambda_i)\right)\right),$$

which is implemented as a function with some additional stabilization to avoid under/overflows in some of the common numerical packages including PyTorch. See [4] for details of numerical properties associated with the log-sum-exp structure.

Dimensionally independent control variate. Although the naive Monte Carlo sampling with a sufficiently large sample size can approximate the left-hand side of Eq. (11) well, a small batch can cause high variance in the evaluation of the expected values. An established way of stabilizing Monte Carlo integration is to use so-called *control variates* [15, 35], also known as *baseline* in reinforcement learning [49]. To estimate an expectation $\mathbb{E}[f]$, we can subtract another function/random variable g, called a control variate, whose integral we know or can compute more precisely than Monte Carlo, and execute the Monte Carlo for $f-g$, by using the decomposition $\mathbb{E}[f] = \mathbb{E}[f-g] + \mathbb{E}[g]$. See Section F for a more detailed explanation. As a concrete application of this technique, we below propose the use of a dimensionally independent control variate.

We first exploit the compositional form of $p^{\mathrm{sg}(\theta)}_{s|u} \circ p^{\psi}_{u|t}(\cdot|\boldsymbol{x}_t)$, which is more informative than $\boldsymbol{x}_s^{(j)}$, the pure samples in the Monte Carlo approach. We can write it in expectation as follows:

$$p^{\mathrm{sg}(\theta)}_{s|u} \circ p^{\psi}_{u|t}(\cdot|\boldsymbol{x}_t) = \mathbb{E}_{\lambda, \boldsymbol{x}_u \sim p^{\psi}_{u|t}(\cdot|\boldsymbol{x}_t)}\left[p^{\mathrm{sg}(\theta)}_{s|u}(\cdot|\boldsymbol{x}_u;\lambda)\right]. \tag{12}$$

To simplify (12), let use denote $q^\eta := p^{\mathrm{sg}(\theta)}_{s|u}(\cdot|\boldsymbol{x}_u;\lambda)$ and $q := \mathbb{E}_\eta[q^\eta]$ with $\eta = (\boldsymbol{x}_u, \lambda)$. To construct an efficient control variate given q, we need a function g such that (i) it reasonably approximates $p^\theta_{s|t}(\cdot|\boldsymbol{x}_t)$ and (ii) $\mathbb{E}_{\boldsymbol{x}\sim q}[g(\boldsymbol{x})]$ is easy to compute/approximate. One such example is the product model defined as

$$\overline{p}^{\theta}_{s|t}(\cdot|\boldsymbol{x}_t) := \prod_{d=1}^{D} \overline{p}^{\theta,d}_{s|t}(\cdot|\boldsymbol{x}_t), \qquad \overline{p}^{\theta,d}_{s|t}(\cdot|\boldsymbol{x}_t) := p^{\theta,d}_{s|t}(\cdot|\boldsymbol{x}_t) = \mathbb{E}_\lambda\left[p^{\theta,d}_{s|t}(\cdot|\boldsymbol{x}_t;\lambda)\right]. \tag{13}$$

We defer the explanation of how (i) and (ii) are satisfied to Section F.1. Given a control variate $\overline{p}^{\theta}_{s|t}(\cdot|\boldsymbol{x}_t)$, we can decompose the loss computation:

$$H(q, p^\theta_{s|t}(\cdot|\boldsymbol{x}_t)) = \underbrace{\mathbb{E}_{\boldsymbol{x}_s\sim q}\left[-\log p^\theta_{s|t}(\boldsymbol{x}_s|\boldsymbol{x}_t) + \log \overline{p}^\theta_{s|t}(\boldsymbol{x}_s|\boldsymbol{x}_t)\right]}_{\text{Monte Carlo by sampling } \boldsymbol{x}_s} + \underbrace{\mathbb{E}_\eta\left[H(q^\eta, \overline{p}^\theta_{s|t}(\cdot|\boldsymbol{x}_t))\right]}_{\text{Monte Carlo by sampling } \eta}. \tag{14}$$

Here, the first term can be treated similarly to (11), and we approximately compute the second term by sampling η and using the identity $H(q^\eta, \overline{p}^\theta_{s|t}(\cdot|\boldsymbol{x}_t)) = \sum_{d=1}^{D} H(q^{\eta,d}, \overline{p}^{\theta,d}_{s|t}(\cdot|\boldsymbol{x}_t))$ (see (64) in Section F.1). In this decomposition, we expect that the mixture model explicitly learns the dimensional correlation with the first term, while the second term stabilizes the overall approximation, as we use more detailed information on q than just its samples. See also Section F.2 for more background on how we derive $\overline{p}^\theta$ and another possible choice of control variate.

B.3 Auxiliary losses

While we can use a similar Monte Carlo estimate for $\mathcal{L}_{\mathrm{data}}$ (with random samples of $\boldsymbol{x}_0, \boldsymbol{x}_t, \lambda$), we can regard $\mathcal{L}_{\mathrm{marginal}}$ as a possible control variate for it. Indeed, if the teacher network is well-trained, we can expect that its marginal approximates the true marginal as $p^{\psi,d} \approx q^d$. Thus, for the marginal-matching product model $\overline{p}^\theta$ given in Eq. (13), we have

$$\mathbb{E}_{\boldsymbol{x}_t\sim q_t}\left[H(q_{0|t}(\cdot|\boldsymbol{x}_t), \overline{p}^\theta_{0|t}(\cdot|\boldsymbol{x}_t))\right] \approx \mathcal{L}_{\mathrm{marginal}}(\theta;\psi,q_t,t) + \mathit{const.}, \tag{15}$$

where the constant term is independent of θ. We give the derivation of (15) in Appendix F.3. We then obtain a decomposed formulation of $\mathcal{L}_{\mathrm{data}}$ for given $\boldsymbol{x}_t \sim q_t$ as follows, by letting $q = q_{0|t}(\cdot|\boldsymbol{x}_t)$ and $s = 0$ in Eq. (14) and then using the approximation (15):

$$\begin{aligned}\mathcal{L}_{\mathrm{data}}(\theta;t) &\approx \mathcal{L}_{\mathrm{corr}}(\theta;t) + \mathcal{L}_{\mathrm{marginal}}(\theta;\psi,q_t,t) + \mathit{const.},\\ \mathcal{L}_{\mathrm{corr}}(\theta;t) &:= \mathbb{E}_{(\boldsymbol{x}_0,\boldsymbol{x}_t)\sim q_{0,t}}\left[-\log p^\theta_{0|t}(\boldsymbol{x}_0|\boldsymbol{x}_t) + \log \overline{p}^\theta_{0|t}(\boldsymbol{x}_0|\boldsymbol{x}_t)\right].\end{aligned}$$

Here, $\mathcal{L}_{\mathrm{corr}}$ measures the difference between p^θ and $\overline{p}^\theta$ and so represents the dimensional correlation learned by the model p^θ. In the actual implementation for the first term $\mathcal{L}_{\mathrm{corr}}$, we generate $\boldsymbol{x}_0 \sim q_0$ and then $\boldsymbol{x}_t \sim q_{t|0}(\cdot|\boldsymbol{x}_0)$, and regard them as samples from $(\boldsymbol{x}_0, \boldsymbol{x}_t) \sim q_{0,t}$, which are required for conducting Monte Carlo. When combining $\mathcal{L}_{\mathrm{data}}$ and $\mathcal{L}_{\mathrm{marginal}}$ (both as loss and control variate), we empirically find that mixing as $\alpha_t \mathcal{L}_{\mathrm{corr}}(\theta;t) + \mathcal{L}_{\mathrm{marginal}}(\theta;\psi,q_t,t)$ with some $\alpha_t \in [0,1]$ depending on t is more efficient than just using constant $\alpha_t = 0$ (pure marginal loss) or $\alpha_t = 1$ (pure data loss). See Section G for details in this regard.

C Kullback–Leibler divergence and total variation distance

Let p and q be probability distributions on the same finite set $\mathcal{X}$. The KL divergence D_{KL} and the total variation distance d_{TV} are defined as follows:

$$D_{\mathrm{KL}}(p \,\|\, q) := \sum_{x\in\mathcal{X}} p(x) \log \frac{p(x)}{q(x)}, \quad d_{\mathrm{TV}}(p, q) := \sup_{A\subset\mathcal{X}} |p(A) - q(A)| = \frac{1}{2} \sum_{x\in\mathcal{X}} |p(x) - q(x)|.$$

Here, in the computation of D_{KL}, we ignore the term with $p(x) = 0$ and, if there is an x with $p(x) > 0$ and $q(x) = 0$, we then define $D_{\mathrm{KL}}(p \,\|\, q) = 0$. These two error criteria between distributions are bridged by the following inequality (see, e.g., [7]).

Proposition 2 (Pinsker's inequality)**.** *For probability distributions p and q on $\mathcal{X}$, we have*

$$d_{\mathrm{TV}}(p, q) \le \sqrt{\frac{1}{2} D_{\mathrm{KL}}(p \,\|\, q)}.$$

The convexity of KL divergence in the following plays a role in the main body of the paper.

Proposition 3 ([10, Theorem 2.7.2])**.** *$D_{\mathrm{KL}}(p \,\|\, q)$ is convex with respect to the pair (p, q). Namely, for $t \in [0, 1]$ and probability distributions p_1, p_2, q_1, q_2 on the same domain, we have*

$$D_{\mathrm{KL}}(tp_1 + (1-t)p_2 \,\|\, tq_1 + (1-t)q_2) \le tD_{\mathrm{KL}}(p_1 \,\|\, q_1) + (1-t)D_{\mathrm{KL}}(p_2 \,\|\, q_2).$$

We also use the following triangle-like inequality for the total variation distance of compositions.

Proposition 4. *For probability distributions $p_1(\cdot|y), p_2(\cdot|y)$ over $\mathcal{X}$ conditioned on $y \in \mathcal{Y}$ and q_1, q_2 over $\mathcal{Y}$, we have*

$$d_{\mathrm{TV}}\left(\mathbb{E}_{y\sim q_1}[p_1(\cdot|y)], \mathbb{E}_{y\sim q_2}[p_2(\cdot|y)]\right) \le \mathbb{E}_{y\sim q_1}[d_{\mathrm{TV}}(p_1(\cdot|y), p_2(\cdot|y))] + d_{\mathrm{TV}}(q_1, q_2).$$

We give its proof in Section E.2.

D Continuous-time Markov chains and Kolmogorov equations

Let us discuss the Kolmogorov forward/backward equations associated with continuous-time Markov chains. While the arguments below are mostly a reorganization of those given in previous studies [6, 46], we explicitly track the continuity/nonzero assumptions used in their derivations.

D.1 Kolmogorov equations in the general case

Let us consider a general Markov process over the continuous time interval $[0, T]$ and a discrete (finite) state space $\mathcal{X}$, which is called a continuous-time Markov chain [1, 6]. The starting block is the forward transition rate in a short-time interval. For $t < t + \epsilon$, assume the following equation for the infinitesimal forward transition:

$$q_{t+\epsilon|t}(y|x) = \delta_{y,x} + \epsilon Q_t(y, x) + o(\epsilon), \qquad \epsilon > 0, \tag{16}$$

where $\delta_{y,x}$ is the Kronecker delta and Q_t is a function $\mathcal{X} \times \mathcal{X} \to \mathbb{R}$ called the transition rate. Here, for $s \le t < t + \epsilon$, we have

$$\begin{aligned} q_{t+\epsilon|s}(y|x) = \sum_z q_{t+\epsilon|t}(y|z) q_{t|s}(z|x) &= \sum_z (\delta_{y,z} + Q_t(y, z)\epsilon) q_{t|s}(z|x) + o(\epsilon) \\ &= q_{t|s}(y|x) + \epsilon \sum_z Q_t(y, z) q_{t|s}(z|x) + o(\epsilon). \end{aligned}$$

This means that we have $\partial_t^+ q_{t|s}(y|x) = \sum_z Q_t(y, z) q_{t|s}(z|x)$, where ∂_t^+ is the right-derivative regarding t. Under the condition that Q_t is continuous over $[0, T]$ (assume it is continuously extended to $t = T$, though it is not necessary right now) and $q_{t|s}$ is continuous over $t \in [s, T]$, $q_{t|s}$ becomes differentiable over the open interval (from a general fact in analysis [48]) and we have the Kolmogorov forward equation for $t \in (s, T)$:

$$\partial_t q_{t|s}(y|x) = \sum_z Q_t(y, z) q_{t|s}(z|x). \tag{17}$$

Now, let us derive the backward equation. For $s < s + \epsilon \leq t$, by using (16), we have

$$q_{t|s}(y|x) = \sum_z q_{t|s+\epsilon}(y|z) q_{s+\epsilon|s}(z|x) = \sum_z q_{t|s+\epsilon}(y|z)(\delta_{z,x} + \epsilon Q_s(z,x)) + o(\epsilon)$$
$$= q_{t|s+\epsilon}(y|x) + \epsilon \sum_z q_{t|s+\epsilon}(y|z) Q_s(z,x) + o(\epsilon).$$

Thus, by additionally assuming the continuity of $q_{t|s}$ for $s \in [0,T]$, we obtain the one-sided derivative $\partial_s^+ q_{t|s}(y|x) = -\sum_z q_{t|s}(y|z) Q_s(z,x)$. When combined with the continuity of Q_s similarly to the above argument on the forward equation, it leads to the backward Kolmogorov equation for $s \in (0,t)$:

$$\partial_s q_{t|s}(y|x) = -\sum_z q_{t|s}(y|z) Q_s(z,x). \tag{18}$$

To summarize so far, under the assumption that $q_{t|s}$ is continuous for s, t with $0 \leq s \leq t \leq T$ and Q_t in (16) is continuous over $[0,T]$, we have the two Kolmogorov equations given by (17) and (18). Note that all the $\sum_z$ are finite sums because of the finiteness of $\mathcal{X}$.

D.2 Kolmogorov equations for factorized forward processes

Let us now consider the case where $\mathcal{X} = \mathcal{S}^D$ and $\boldsymbol{x}_t = (x_t^d)_{d=1}^D$ follows a dimensionally independent forward process with transition rate Q_t^d. Namely, suppose

$$q^d_{t+\epsilon|t}(y^d|x^d) = \delta_{y^d,x^d} + \epsilon Q_t^d(y^d,x^d) + o(\epsilon) \tag{19}$$

for each $d = 1, \ldots, D$ and $t < t + \epsilon$. In this case, we have

$$q_{t+\epsilon|t}(\boldsymbol{y}|\boldsymbol{x}) = \prod_{d=1}^D q^d_{t+\epsilon|t}(y^d|x^d) = \delta_{\boldsymbol{y},\boldsymbol{x}} + \epsilon \sum_{d=1}^D Q_t^d(y^d,x^d)\delta_{\boldsymbol{y}^{\backslash d},\boldsymbol{x}^{\backslash d}} + o(\epsilon) \tag{20}$$

by simply expanding the product, where $\boldsymbol{x}^{\backslash d} \in \mathcal{S}^{D-1}$ is given by omitting the d-th entry of $\boldsymbol{x}$. From (20), the transition rate for $\boldsymbol{x}_t$ is given by

$$Q_t(\boldsymbol{y},\boldsymbol{x}) = \sum_{d=1}^D Q_t^d(y^d,x^d)\delta_{\boldsymbol{y}^{\backslash d},\boldsymbol{x}^{\backslash d}} \tag{21}$$

as in Campbell et al. [6, Proposition 3]. Let us assume the continuity regarding the forward process in each dimension:

Assumption A. *For each $d = 1, \ldots, D$, there exists a function $Q_t^d : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ indexed by $t \in [0,T]$ satisfying Eq. (19). Furthermore, for any fixed $x, y \in \mathcal{S}$, $q^d_{t|s}(y|x)$ is continuous over $\{(s,t) \in [0,T]^d \mid s \leq t\}$ and $Q_t^d(y,x)$ is continuous over $[0,T]$.*

Under this assumption, $q_{t|s}$ and Q_t for the original process $\boldsymbol{x}_t$ are also continuous since we have $q_{t|s}(\boldsymbol{y}|\boldsymbol{x}) = \prod_{d=1}^D q^d_{t|s}(y^d|x^d)$ and (21). Thus, we can apply the argument in Section D.1 to obtain the Kolmogorov equations (17) & (18).

To consider the time-reversal transition rate, let us further assume the following property for the forward process:

Assumption B. *For any $t \in [0,T]$ and $\boldsymbol{x} \in \mathcal{S}^D$, $q_t(\boldsymbol{x}) > 0$ holds.*

This is satisfied, for instance, when $q_{\mathrm{data}}(\boldsymbol{x}) > 0$ for all $\boldsymbol{x} \in \mathcal{S}^D$ and $q^d_{t|s}(y|x) > 0$ for all $x, y \in \mathcal{S}$ and d. The latter holds true for common forward diffusions such as uniform diffusion and discretized Gaussian [2].

Under these assumptions, we can show a favorable property of the time-reversal process. This is just a re-formalization of a well-known fact (e.g., Campbell et al. [6, Proposition 3] and Sun et al. [46, Proposition 3.2]).

Proposition 5. *Under Assumptions A & B, there exists a function $R_t : \mathcal{S}^D \times \mathcal{S}^D \to \mathbb{R}$ indexed by $t \in (0,T]$ such that* *(a) we have $q_{t-\epsilon|t}(\boldsymbol{y}|\boldsymbol{x}) = \delta_{\boldsymbol{y},\boldsymbol{x}} + \epsilon R_t(\boldsymbol{y},\boldsymbol{x}) + o(\epsilon)$ for $\epsilon > 0$ with $t - \epsilon \geq 0$, and*

(b) $R_t(\boldsymbol{y},\boldsymbol{x})$ can be nonzero only if $\boldsymbol{x},\boldsymbol{y} \in \mathcal{S}^D$ coincide in at least $D-1$ entries.

We give its proof in Section E.3. As one can see from the proof, the time-reversal transition rate R_t is given concretely by $R_t(\boldsymbol{y},\boldsymbol{x}) = Q_t(\boldsymbol{x},\boldsymbol{y})q_t(\boldsymbol{y})/q_t(\boldsymbol{x})$ when $\boldsymbol{x} \neq \boldsymbol{y}$, and the ratio $q_t(\boldsymbol{y})/q_t(\boldsymbol{x})$ is treated as a discrete counterpart of the score function [46, 29].

Let us add one more regularity assumption:

Assumption C. *For each $d = 1, \dots, D$ and $x, y \in \mathcal{S}$, $Q_t^d(y,x)$ is differentiable for $t \in (0,T)$ and the derivative $\partial_t Q_t^d(y,x)$ can be continuously extended to $[0,T]$.*

Note that usual choices of Q_t^d regarding t including the time-homogeneous case $Q_t = Q$ and the noise scheduling $Q_t = \beta(t)Q$ with a smooth β [6, 29] satisfy this assumption. Finally, under these three assumptions, we can formalize Theorem 1 as follows.

Theorem 3. *Suppose $(\boldsymbol{x}_t)_{0\leq t\leq T}$ satisfies Assumptions A, B & C. Let $p_{s|t}$ be a product model with the correct marginals, i.e., $p_{s|t}(\boldsymbol{x}_s|\boldsymbol{x}_t) = \prod_{d=1}^D q_{s|t}^d(x_s^d|\boldsymbol{x}_t)$ for $s < t$. Then, there exists a constant $C > 0$ such that, given timesteps $t_i = iT/N$ for $i = 0, \dots, N$, we have*

$$d_{\mathrm{TV}}\big(q_0, \mathbb{E}_{\boldsymbol{x}_T\sim q_T}\left[p_{t_0|t_1} \circ p_{t_1|t_2} \circ \cdots \circ p_{t_{N-1}|t_N}(\cdot|\boldsymbol{x}_T)\right]\big) \leq \frac{C}{N}. \tag{22}$$

Furthermore, there exists an example of $(\boldsymbol{x}_t)_{0\leq t\leq T}$ satisfying $D = |\mathcal{S}| = 2$ and the same assumptions such that the left-hand side of (22) is lower-bounded by c/N with some constant $c > 0$ for sufficiently large N.

This theorem basically says the min-max convergence rate of the analytical sampling is $1/N$. We give the proof of the first half, i.e., Eq. (22), in Section E.4. For the latter half, we provide the concrete version in Proposition 6 in the following section.

D.3 A lower bound of Theorem 3

We shall provide an example that yields an $\Omega(1/N)$ error between the analytical and true denoisers. Consider $\mathcal{S} = \{a, b\}$ and $D = 2$, where the state-space is given by $\mathcal{X} = \{aa, ab, ba, bb\}$ by omitting parentheses. Consider the (forward) Markov process given by the initial distribution $q_0 = (\delta_{aa} + \delta_{bb})/2$ and the dimension-wise time-homogeneours transition rate $Q_t^d(y,x) = 1/2 - \delta_{yx}$ for $d = 1, 2$ and $x, y \in \mathcal{S}$. Under this setting, the forward transition probability is continuous and satisfies $q_{t|s}^d(\cdot|0) = Q_t^d q_{t|s}^d(\cdot|0)$ as a vector-valued differential equation, and so we have, for $t > s$,

$$\partial_t q_{t|s}^d(a|a) = -\frac{1}{2}q_{t|s}^d(a|a) + \frac{1}{2}q_{t|s}^d(b|a) = \frac{1}{2} - q_{t|s}^d(a|a).$$

By solving this, we obtain $q_{t|s}^d(a|a) = \frac{1}{2}(1 + e^{-(t-s)})$ for $t \geq s$. By symmetry, we generally have

$$q_{t|s}^d(a|a) = q_{t|s}^d(b|b) = \frac{1}{2}(1 + e^{-(t-s)}), \quad q_{t|s}^d(b|a) = q_{t|s}^d(a|b) = \frac{1}{2}(1 - e^{-(t-s)}) \tag{23}$$

This is a special case of uniform diffusion and clearly satisfies Assumptions A & C. Although the singularity of q_0 violates Assumption B at time zero, we can consider the time interval $[\delta, T]$ for some $\delta > 0$ instead of $[0,T]$ to ensure $q_t > 0$. We will, however, work with the singular q_0 for simplicity of computations. The following proposition gives the lower bound discussed in Theorem 3. If necessary, we can replace T with $T + \delta$ and consider $\boldsymbol{x}_t' = \boldsymbol{x}_{t+\delta}$ to match the time intervals.

Proposition 6. *Let $(\boldsymbol{x}_t)_{\delta\leq t\leq T}$ be the Markov process defined above and $p_{s|t}$ be the product model $p_{s|t}(\boldsymbol{x}_s|\boldsymbol{x}_t) = \prod_{d=1}^D q_{s|t}^d(x_s^d|\boldsymbol{x}_t)$ for $s < t$. If we let $N \geq 2(T-\delta)/\delta$ be an integer and $t_i = \delta + i(T-\delta)/N$ for $i = 0, \dots, N$ be timesteps, then there is a constant $c > 0$ such that*

$$d_{\mathrm{TV}}\big(q_\delta, \mathbb{E}_{\boldsymbol{x}_T\sim q_T}\left[p_{t_0|t_1} \circ p_{t_1|t_2} \circ \cdots \circ p_{t_{N-1}|t_N}(\cdot|\boldsymbol{x}_T)\right]\big) \geq \frac{c}{N}. \tag{24}$$

The proof is given in Section E.5.

E Proofs

E.1 Proof of Theorem 2

Proof. For simplicity of notation, let $\tilde{p}^{\psi}_{t_n|T}$ be the denoiser given by the teacher with timesteps $t_n < t_{n+1} < \cdots < t_N$, i.e,

$$\tilde{p}^{\psi}_{t_n|T} := p^{\psi}_{t_n|t_{n+1}} \circ \cdots \circ p^{\psi}_{t_{N-1}|t_N},$$

so that we have $r_{t_n} = \mathbb{E}_{\boldsymbol{x}_T \sim r_T}\left[\tilde{p}^{\psi}_{t_n|T}(\cdot|\boldsymbol{x}_T)\right]$. Note that we can just set $\tilde{p}^{\psi}_{t_N|T}(\cdot|\boldsymbol{x}) = \tilde{p}^{\psi}_{T|T}(\cdot|\boldsymbol{x}) = \delta_{\boldsymbol{x}}$.

Also, let $p_{0,n} := \mathbb{E}_{\boldsymbol{x}_T \sim r_T}\left[p^{\theta}_{0|t_n} \circ \tilde{p}^{\psi}_{t_n|T}(\cdot|\boldsymbol{x}_T)\right]$ for $n = 1, \dots, N$, where $p_{0,N}$ is just given by $p_{0,N} = \mathbb{E}_{\boldsymbol{x}_T \sim r_T}\left[p^{\theta}_{0|T}(\cdot|\boldsymbol{x}_T)\right]$. We first compare $p_{0,n}$ and $p_{0,n+1}$ with the consistency loss.

For each $0 < u < t \le T$, we have

$$\begin{aligned}\mathcal{L}_{\text{consis}}(\theta; \psi, r_t, 0, u, t) &= \mathbb{E}_{\boldsymbol{x}_t \sim r_t}\left[D_{\text{KL}}(p^{\theta}_{0|u} \circ p^{\psi}_{u|t}(\cdot|\boldsymbol{x}_t) \,\|\, p^{\theta}_{0|t}(\cdot|\boldsymbol{x}_t))\right] \\ &\ge D_{\text{KL}}\left(\mathbb{E}_{\boldsymbol{x}_t \sim r_t}\left[p^{\theta}_{0|u} \circ p^{\psi}_{u|t}(\cdot|\boldsymbol{x}_t)\right] \,\middle\|\, \mathbb{E}_{\boldsymbol{x}_t \sim r_t}\left[p^{\theta}_{0|t}(\cdot|\boldsymbol{x}_t)\right]\right)\end{aligned}$$

from the convexity (Proposition 3). If we let $u = t_n$ and $t = t_{n+1}$ for some $1 \le n < N$, we can see

$$\mathbb{E}_{\boldsymbol{x}_t \sim r_t}\left[p^{\theta}_{0|u} \circ p^{\psi}_{u|t}(\cdot|\boldsymbol{x}_t)\right] = \mathbb{E}_{\boldsymbol{x}_T}\left[p^{\theta}_{0|t_n} \circ p^{\psi}_{t_n|t_{n+1}} \circ \tilde{p}^{\psi}_{t_{n+1}|T}(\cdot|\boldsymbol{x}_T)\right] = p_{0,n},$$

and $\mathbb{E}_{\boldsymbol{x}_t \sim r_t}\left[p^{\theta}_{0|t}(\cdot|\boldsymbol{x}_t)\right] = p_{0,n+1}$ hold. By using Pinsker's inequality (Proposition 2), we have

$$d_{\text{TV}}(p_{0,n}, p_{0,n+1}) \le \frac{1}{\sqrt{2}} D_{\text{KL}}(p_{0,n} \,\|\, p_{0,n+1})^{1/2} \le \frac{1}{\sqrt{2}} \mathcal{L}_{\text{consis}}(\theta; \psi, r_{t_{n+1}}, 0, t_n, t_{n+1})^{1/2}. \tag{25}$$

From a similar argument, we have

$$\mathcal{L}_{\text{distil}}(\theta; \psi, r_{t_1}, t_1) = \mathbb{E}_{\boldsymbol{x}_{t_1} \sim r_{t_1}}\left[D_{\text{KL}}(p^{\psi}_{0|t_1}(\cdot|\boldsymbol{x}_{t_1}) \,\|\, p^{\theta}_{0|t_1}(\cdot|\boldsymbol{x}_{t_1}))\right] \ge D_{\text{KL}}(r_0 \,\|\, p_{0,1}),$$

and so

$$d_{\text{TV}}(r_0, p_{0,1}) \le \frac{1}{\sqrt{2}} D_{\text{KL}}(r_0 \,\|\, p_{0,1})^{1/2} \le \frac{1}{\sqrt{2}} \mathcal{L}_{\text{distil}}(\theta; \psi, r_{t_1}, t_1)^{1/2}. \tag{26}$$

By using the triangle inequality of total variation distance, we obtain

$$d_{\text{TV}}(r_0, p_{0,N}) \le d_{\text{TV}}(r_0, p_{0,1}) + \sum_{n=1}^{N-1} d_{\text{TV}}(p_{0,n}, p_{0,n+1}).$$

Finally, applying Eqs. (25) and (26) to its right-hand side yields the desired inequality. □

E.2 Proof of Proposition 4

Proof. Let us first consider the case of $q_1 = q_2$. Then, we have

$$\begin{aligned}&d_{\text{TV}}(\mathbb{E}_{y \sim q_1}[p_1(\cdot|y)], \mathbb{E}_{y \sim q_1}[p_2(\cdot|y)]) \\ &= \frac{1}{2}\sum_x \left|\sum_y p_1(x|y) q_1(y) - \sum_y p_2(x|y) q_1(y)\right| = \frac{1}{2}\sum_x \left|\sum_y (p_1(x|y) - p_2(x|y)) q_1(y)\right| \\ &\le \frac{1}{2}\sum_x \sum_y |p_1(x|y) - p_2(x|y)|\, q_1(y) = \mathbb{E}_{y \sim q_1}[d_{\text{TV}}(p_1(\cdot|y), p_2(\cdot|y))],\end{aligned} \tag{27}$$

where we have used $q_1 \geq 0$ in the inequality. On the other hand, if $p_1 = p_2$, we have

$$\begin{aligned}
&d_{\mathrm{TV}}(\mathbb{E}_{y\sim q_1}[p_2(\cdot|y)], \mathbb{E}_{y\sim q_2}[p_2(\cdot|y)]) \\
&= \frac{1}{2}\sum_x \left|\sum_y p_2(x|y)q_1(y) - \sum_y p_2(x|y)q_2(y)\right| = \frac{1}{2}\sum_x \left|\sum_y p_2(x|y)(q_1(y) - q_2(y))\right| \\
&\leq \frac{1}{2}\sum_x\sum_y p_2(x|y)|q_1(y) - q_2(y)| = \frac{1}{2}\sum_y |q_1(y) - q_2(y)| = d_{\mathrm{TV}}(q_1, q_2), \qquad (28)
\end{aligned}$$

where we have used $p_2 \geq 0$ in the inequality and $\sum_x p_2(x|y) = 1$ in the last equality.

By utilizing the usual triangle inequality of d_{TV} and the inequalities (27) & (28), we obtain

$$\begin{aligned}
&d_{\mathrm{TV}}(\mathbb{E}_{y\sim q_1}[p_1(\cdot|y)], \mathbb{E}_{y\sim q_2}[p_2(\cdot|y)]) \\
&\leq d_{\mathrm{TV}}(\mathbb{E}_{y\sim q_1}[p_1(\cdot|y)], \mathbb{E}_{y\sim q_1}[p_2(\cdot|y)]) + d_{\mathrm{TV}}(\mathbb{E}_{y\sim q_1}[p_2(\cdot|y)], \mathbb{E}_{y\sim q_2}[p_2(\cdot|y)]) \\
&\leq \mathbb{E}_{y\sim q_1}[d_{\mathrm{TV}}(p_1(\cdot|y), p_2(\cdot|y))] + d_{\mathrm{TV}}(q_1, q_2),
\end{aligned}$$

which is the desired inequality. □

E.3 Proof of Proposition 5

Proof. Note that, by Assumption A, $q_{t|s}$ is continuous over $\{(s,t) \in [0,T]^2 \mid s \leq t\}$, and Q_t given by (21) is continuous over $[0,T]$ and satisfies Eqs. (16)–(18), as mentioned in Section D.2 before Assumption B.

Now we work under Assumption B. Let us simply write $x \in \mathcal{X}$ instead of the bold style $\boldsymbol{x} \in \mathcal{S}^D$ in this paragraph. We follow the argument in Sun et al. [46, Section B.2]. Let us consider the conditional probability (namely, the true denoiser) $q_{s|t}$ for $s \leq t$, which is uniquely determined since $q_t > 0$. Then, we have

$$\begin{aligned}
\partial_s q_{s|t}(y|x) &= \partial_s \frac{q_s(y)q_{t|s}(x|y)}{q_t(x)} = \frac{(\partial_s q_s)(y)q_{t|s}(x|y) + q_s(y)(\partial_s q_{t|s})(x|y)}{q_t(x)} \\
&= \frac{1}{q_t(x)}\left(q_{t|s}(x|y)\sum_z Q_s(y,z)q_s(z) - q_s(y)\sum_w q_{t|s}(x|w)Q_s(w,y)\right), \qquad (29)
\end{aligned}$$

where we have used the forward Kolmogorov equation of q_t given as

$$\partial_t q_t(x) = \sum_w \partial_t q_{t|0}(x|w)q_0(w) = \sum_w\sum_z Q_t(x,z)q_{t|0}(z|w)q_0(w) = \sum_z Q_t(x,z)q_t(z)$$

for computing $\partial_s q_s$ and the backward Kolmogorov equation for computing $\partial_s q_{t|s}$. By taking the limit $s \to t - 0$ in (29), we obtain $\lim_{s\to t-0} \partial_s q_{s|t}(y|x) = -\frac{q_t(y)}{q_t(x)}Q_t(x,y)$ if $y \neq x$, given the continuity of $q_{t|s}$ and Q_s. Then, from Taylor's theorem, we obtain a backward counterpart of (16) for $y \neq x$ as

$$q_{t-\epsilon|t}(y|x) = \epsilon\frac{q_t(y)}{q_t(x)}Q_t(x,y) + o(\epsilon), \qquad \epsilon > 0. \qquad (30)$$

Since $\sum_y q_{t-\epsilon|t}(y|x) = 1$ holds always, we also have that $q_{t-\epsilon|t}(x|x) = 1 + \epsilon R_{t,x} + o(\epsilon)$ for the coefficient $R_{t,x} = -\sum_{y\neq x}\frac{q_t(y)}{q_t(x)}Q_t(x,y)$. Therefore, we can prove (a) by letting $R_t(y,x) = \frac{q_t(y)}{q_t(x)}Q_t(x,y)$ for $y \neq x$ and $R_t(x,x) = R_{t,x}$.

We can see (b) from (21) and the concrete form of R_t. □

E.4 Proof of the first half of Theorem 3

We first prove the following auxiliary lemma replacing the $o(\epsilon)$ term in the backward transition by $O(\epsilon^2)$.

Lemma 1. *Under the same setting as in Theorem 3, there is a constant $C > 0$ such that, for any $t \in (0,T]$, $\epsilon \in (0,t]$, and $\boldsymbol{x} \in \mathcal{X}$, we have*

$$d_{\mathrm{TV}}(q_{t-\epsilon|t}(\cdot|\boldsymbol{x}), p_{t-\epsilon|t}(\cdot|\boldsymbol{x})) \leq C\epsilon^2. \qquad (31)$$

Proof. From (29) and Assumption C, $q_{s|t}(\boldsymbol{y}|\boldsymbol{x})$ for $s < t$ is twice-differentiable with regard to s, and $q_t(\boldsymbol{x})\partial_s q_{s|t}(\boldsymbol{y}|\boldsymbol{x})$ can be represented as a polynomial of the function values of q_s, Q_s, $q_{t|s}$, and $\partial_s Q_s$. Thus, there is a constant C_1 depending on $|\mathcal{S}|$, D, $\sup_{s,\boldsymbol{z},\boldsymbol{w}} Q_s(\boldsymbol{z},\boldsymbol{w})$ and $\sup_{s,\boldsymbol{z},\boldsymbol{w}} \partial_s(\boldsymbol{z},\boldsymbol{w})$ such that $\partial_s^2 q_{s|t}(\boldsymbol{y}|\boldsymbol{x}) \le C_1$ for any $s, t, \boldsymbol{y}, \boldsymbol{x}$ (note that $q_{t|s}$ and q_s are within $[0, 1]$).

Now that $\partial_s q_{s|t}$ can be continuously extended to $s \in [0, t]$ from (29) and Assumption B, for each $t \in (0, T]$, $\epsilon \in (0, t]$ and $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{S}^D$, Taylor's theorem yields that

$$\left|q_{t-\epsilon|t}(\boldsymbol{y}|\boldsymbol{x}) - \delta_{\boldsymbol{y},\boldsymbol{x}} - \epsilon R_t(\boldsymbol{y},\boldsymbol{x})\right| = \left|\frac{(\partial_s^2 q_{s|t})(\boldsymbol{y}|\boldsymbol{x})|_{s=\theta}}{2}\epsilon^2\right| \le \frac{C_1}{2}\epsilon^2, \tag{32}$$

for a certain $\theta \in (t-\epsilon, t)$.

Let us next consider the marginal-matching product model $p_{t-\epsilon|t}$. For each d, if $y^d \neq x^d$, we have

$$\begin{aligned}\left|p^d_{t-\epsilon|t}(y^d|\boldsymbol{x}) - \epsilon R_t((y^d, \boldsymbol{x}^{\backslash d}), \boldsymbol{x})\right| &= \left|\sum_{\boldsymbol{y}^{\backslash d}\in\mathcal{S}^{D-1}} q_{t-\epsilon|t}((y^d, \boldsymbol{y}^{\backslash d})|\boldsymbol{x}) - \epsilon R_t((y^d, \boldsymbol{x}^{\backslash d}), \boldsymbol{x})\right| \\ &= \left|\sum_{\boldsymbol{y}^{\backslash d}} \Big(q_{t-\epsilon|t}((y^d, \boldsymbol{y}^{\backslash d})|\boldsymbol{x}) - \epsilon R_t((y^d, \boldsymbol{y}^{\backslash d}), \boldsymbol{x})\Big)\right| \\ &\le \frac{|\mathcal{S}|^{D-1}C_1}{2}\epsilon^2, \end{aligned} \tag{33}$$

where the second equality comes from Proposition 5(b) and the inequality is from (32). If $y^d = x^d$, since $p^d_{t-\epsilon|t}(x^d|\boldsymbol{x}) = 1 - \sum_{y^d\neq x^d} |p^d_{t-\epsilon|t}(y^d|\boldsymbol{x})$ we can use (33) to obtain

$$\begin{aligned}\left|p^d_{t-\epsilon}(x^d|\boldsymbol{x}) - 1 + \epsilon\sum_{y^d\neq x^d} R_t((y^d, \boldsymbol{x}^{\backslash d}), \boldsymbol{x})\right| &\le \sum_{y^d\neq x^d} |p^d_{t-\epsilon|t}(y^d|\boldsymbol{x}) - \epsilon R_t((y^d, \boldsymbol{x}^{\backslash d}), \boldsymbol{x})| \\ &\le \frac{|\mathcal{S}|^D C_1}{2}\epsilon^2.\end{aligned}$$

From (33) and this, by defining $R^d_t : \mathcal{S} \to \mathbb{R}$ as $R^d_t(y^d) = R_t((y^d, \boldsymbol{x}^{\backslash d}), \boldsymbol{x})$ for $y^d \neq x^d$ and $R^d_t(x^d) = -\sum_{y^d\neq x^d} R^d_t(y^d)$, there exists a constant $C_2 > 0$ and a function $A^d : \mathcal{S} \to \mathbb{R}$ (for fixed t and $\boldsymbol{x}$) such that

$$p^d_{t-\epsilon|t}(y^d|\boldsymbol{x}) = \delta_{y^d,x^d} - \epsilon R^d_t(y^d) + \epsilon^2 A^d(y^d, \epsilon), \qquad \sup_{y^d\in\mathcal{S},\ \epsilon} \left|A^d(y^d, \epsilon)\right| \le C_2. \tag{34}$$

Therefore, we have

$$\begin{aligned}p_{t-\epsilon|t}(\boldsymbol{y}|\boldsymbol{x}) &= \prod_{d=1}^{D}\left(\delta_{y^d,x^d} - \epsilon R^d_t(y^d) + \epsilon^2 A^d(y^d, \epsilon)\right) \\ &= \delta_{\boldsymbol{y},\boldsymbol{x}} + \epsilon\sum_{d=1}^{D} R^d_t(y^d)\delta_{\boldsymbol{y}^{\backslash d},\boldsymbol{x}^{\backslash d}} + \epsilon^2 P_3(\epsilon, (\delta_{y^d,x^d},\ R^d_t(y^d),\ A^d(y^d, \epsilon))_{d=1}^{D}),\end{aligned}$$

where P_3 is a certain polynomial of $3D + 1$ variables. Note that, if $\boldsymbol{y} \neq \boldsymbol{x}$, $R^d_t(y^d)\delta_{\boldsymbol{y}^{\backslash d},\boldsymbol{x}^{\backslash d}}$ can be nonzero only if $y^d \neq x^d$ and $\boldsymbol{y}^{\backslash d} = \boldsymbol{x}^{\backslash d}$. In that case, from the definition of $R^d_t(y^d)$, we have

$$p_{t-\epsilon|t}(\boldsymbol{y}|\boldsymbol{x}) = \epsilon R^d_t(y^d) + \epsilon^2 P_3 = \epsilon R_t(\boldsymbol{y}, \boldsymbol{x}) + \epsilon^2 P_3. \tag{35}$$

This equality also holds when $\boldsymbol{y}$ and $\boldsymbol{x}$ differ in more than one entry, since the coefficient of ϵ becomes zero in such a case, and $R_t(\boldsymbol{y}, \boldsymbol{x}) = 0$ from Proposition 5(b). Since the inputs for P_3 are all bounded, by combining it with (32), for $\boldsymbol{y} \neq \boldsymbol{x}$, we have

$$\begin{aligned}|q_{t-\epsilon|t}(\boldsymbol{y}|\boldsymbol{x}) - p_{t-\epsilon|t}(\boldsymbol{y}|\boldsymbol{x})| &\le |q_{t-\epsilon|t}(\boldsymbol{y}|\boldsymbol{x}) - \epsilon R_t(\boldsymbol{y}, \boldsymbol{x})| + |\epsilon R_t(\boldsymbol{y}, \boldsymbol{x}) - q_{t-\epsilon|t}(\boldsymbol{y}|\boldsymbol{x})| \\ &\le \frac{C_1}{2}\epsilon^2 + (\sup P_3)\epsilon^2 \le C_4\epsilon^2\end{aligned}$$

for a constant $C_4 > 0$. In particular, we have

$$\begin{aligned}
d_{\mathrm{TV}}(q_{t-\epsilon|t}(\cdot|\boldsymbol{x}), p_{t-\epsilon|t}(\cdot|\boldsymbol{x})) &= \frac{1}{2}\sum_{\boldsymbol{y}} |q_{t-\epsilon|t}(\boldsymbol{y}|\boldsymbol{x}) - p_{t-\epsilon|t}(\boldsymbol{y}|\boldsymbol{x})| \\
&= \frac{1}{2}\left(\sum_{\boldsymbol{y}\neq\boldsymbol{x}} |q_{t-\epsilon|t}(\boldsymbol{y}|\boldsymbol{x}) - p_{t-\epsilon|t}(\boldsymbol{y}|\boldsymbol{x})| + \left|1 - \sum_{\boldsymbol{y}\neq\boldsymbol{x}} q_{t-\epsilon|t}(\boldsymbol{y}|\boldsymbol{x}) - 1 + \sum_{\boldsymbol{y}\neq\boldsymbol{x}} p_{t-\epsilon|t}(\boldsymbol{y}|\boldsymbol{x})\right|\right) \\
&\leq \sum_{\boldsymbol{y}\neq\boldsymbol{x}} |q_{t-\epsilon|t}(\boldsymbol{y}|\boldsymbol{x}) - p_{t-\epsilon|t}(\boldsymbol{y}|\boldsymbol{x})| \leq |S|^D C_4 \epsilon^2,
\end{aligned} \tag{36}$$

which proves (31). □

By using the lemma and Proposition 4, we can prove the theorem.

Proof of Theorem 3. For each $i = 0, \dots, N$, let us define the compositions

$$\tilde{p}_{0|t_0}(\cdot|\boldsymbol{x}) = \delta_{\boldsymbol{x}}, \qquad \tilde{p}_{0|t_i} := p_{t_0|t_1} \circ \cdots \circ p_{t_{i-1}|t_i}, \quad i = 1, \dots, N.$$

Note also that we have $q_{t_i|T} = q_{t_i|t_{i+1}} \circ \cdots \circ q_{t_{N-1}|t_N}$ from the Markov property of the reverse process. Indeed, for $s < t < u$, we have $q_{u|t}(\boldsymbol{z}|\boldsymbol{y}) = q_{u|s,t}(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{y})$ from the Markov property of the forward process, and so

$$\begin{aligned}
\sum_{\boldsymbol{y}} q_{s|t}(\boldsymbol{x}|\boldsymbol{y}) q_{t|u}(\boldsymbol{y}|\boldsymbol{z}) &= \sum_{\boldsymbol{y}} \frac{q_{s,t}(\boldsymbol{x}, \boldsymbol{y})}{q_t(\boldsymbol{y})} \frac{q_{t,u}(\boldsymbol{y}, \boldsymbol{z})}{q_u(\boldsymbol{z})} \\
&= \sum_{\boldsymbol{y}} \frac{q_{s,t}(\boldsymbol{x}, \boldsymbol{y}) q_{u|t}(\boldsymbol{z}|\boldsymbol{y})}{q_u(\boldsymbol{z})} = \sum_{\boldsymbol{y}} \frac{q_{s,t}(\boldsymbol{x}, \boldsymbol{y}) q_{u|s,t}(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{y})}{q_u(\boldsymbol{z})} \\
&= \frac{\sum_{\boldsymbol{y}} q_{s,t,u}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})}{q_u(\boldsymbol{z})} = \frac{q_{s,u}(\boldsymbol{x}, \boldsymbol{z})}{q_u(\boldsymbol{z})} = q_{s|u}(\boldsymbol{x}|\boldsymbol{z}),
\end{aligned}$$

where we have implicitly used Assumption B. By using the inequality recursively, we can prove the aforementioned identity.

We prove the desired estimate by exploiting the compositions. Recall $q_0 = \mathbb{E}_{\boldsymbol{x}_T \sim q_T}\left[q_{0|t_N}(\cdot|\boldsymbol{x}_T)\right]$. What we want to estimate is $d_{\mathrm{TV}}(\mathbb{E}_{\boldsymbol{x}_T \sim q_T}\left[q_{0|t_N}(\cdot|\boldsymbol{x}_T)\right], \mathbb{E}_{\boldsymbol{x}_T \sim q_T}\left[\tilde{p}_{0|t_N}(\cdot|\boldsymbol{x}_T)\right])$. We bound the distance with the following triangle inequality:

$$\begin{aligned}
&d_{\mathrm{TV}}(\mathbb{E}_{\boldsymbol{x}_T \sim q_T}\left[q_{0|t_N}(\cdot|\boldsymbol{x}_T)\right], \mathbb{E}_{\boldsymbol{x}_T \sim q_T}\left[\tilde{p}_{0|t_N}(\cdot|\boldsymbol{x}_T)\right]) \\
&\leq \sum_{i=0}^{N-1} d_{\mathrm{TV}}(\mathbb{E}_{\boldsymbol{x}_T \sim q_T}\left[\tilde{p}_{0|t_i} \circ q_{t_i|t_N}(\cdot|\boldsymbol{x}_T)\right], \mathbb{E}_{\boldsymbol{x}_T \sim q_T}\left[\tilde{p}_{0|t_{i+1}} \circ q_{t_{i+1}|t_N}(\cdot|\boldsymbol{x}_T)\right]).
\end{aligned} \tag{37}$$

Let us bound each term inside the summation by using Lemma 1 and Proposition 4. First, since $\tilde{p}_{0|t_{i+1}} = \tilde{p}_{0|t_i} \circ p_{t_i|t_{i+1}}$, by letting $p_1 = p_2 = \tilde{p}_{0|t_i}$ in Proposition 4, we have

$$\begin{aligned}
&d_{\mathrm{TV}}(\mathbb{E}_{\boldsymbol{x}_T \sim q_T}\left[\tilde{p}_{0|t_i} \circ q_{t_i|t_N}(\cdot|\boldsymbol{x}_T)\right], \mathbb{E}_{\boldsymbol{x}_T \sim q_T}\left[\tilde{p}_{0|t_{i+1}} \circ q_{t_{i+1}|t_N}(\cdot|\boldsymbol{x}_T)\right]) \\
&\leq d_{\mathrm{TV}}(\mathbb{E}_{\boldsymbol{x}_T \sim q_T}\left[q_{t_i|t_N}(\cdot|\boldsymbol{x}_T)\right], \mathbb{E}_{\boldsymbol{x}_T \sim q_T}\left[p_{t_i|t_{i+1}} \circ q_{t_{i+1}|t_N}(\cdot|\boldsymbol{x}_T)\right]).
\end{aligned} \tag{38}$$

Second, since $q_{t_{i+1}|t_N} = q_{t_i|t_{i+1}} \circ q_{t_{i+1}|t_N}$, by letting $q := q_1 = q_2 = \mathbb{E}_{\boldsymbol{x}_T \sim q_T}\left[q_{t_{i+1}|t_N}(\cdot|\boldsymbol{x}_T)\right]$ in Proposition 4 (note that the indices of q_1, q_2 here are different from time), we have

$$\begin{aligned}
&d_{\mathrm{TV}}(\mathbb{E}_{\boldsymbol{x}_T \sim q_T}\left[q_{t_i|t_N}(\cdot|\boldsymbol{x}_T)\right], \mathbb{E}_{\boldsymbol{x}_T \sim q_T}\left[p_{t_i|t_{i+1}} \circ q_{t_{i+1}|t_N}(\cdot|\boldsymbol{x}_T)\right]) \\
&\leq \mathbb{E}_{\boldsymbol{x} \sim q}\left[d_{\mathrm{TV}}(q_{t_i|t_{i+1}}(\cdot|\boldsymbol{x}), p_{t_i|t_{i+1}}(\cdot|\boldsymbol{x}))\right] \leq \frac{CT^2}{N^2},
\end{aligned} \tag{39}$$

where we have used (31) and $t_{i+1} - t_i = T/N$ in the last inequality. By combining the estimates (37)–(39), we obtain

$$d_{\mathrm{TV}}(\mathbb{E}_{\boldsymbol{x}_T \sim q_T}\left[q_{0|t_N}(\cdot|\boldsymbol{x}_T)\right], \mathbb{E}_{\boldsymbol{x}_T \sim q_T}\left[\tilde{p}_{0|t_N}(\cdot|\boldsymbol{x}_T)\right]) \leq \sum_{i=0}^{N-1} \frac{CT^2}{N^2} = \frac{CT^2}{N},$$

which completes the proof with a replacement of the constant factor. □

E.5 Proof of Proposition 6

Proof. Consider the analytical sampler $p_{s|t}(zw|xy) = q^1_{s|t}(z|x)q^2_{s|t}(w|y)$ for $s < t$. Note that, because of the symmetry between a and b in q_0 and the forward transition, the distributions q_t or those given by the composition of $p_{s|t}$ are also symmetric. Thus, the probability of aa recovers all the information of the distributions we consider over $\mathcal{X}$.

Let us compute several probabilities regarding $q_{s|t}$ and the analytical sampler through (23). First, note that $q_{0|t}(ab|\cdot) = q_{0|t}(ba|\cdot) = 0$. Therefore, we have

$$\begin{aligned} q_{0|t}(aa|aa) &= \frac{q_{t|0}(aa|aa)q_0(aa)}{q_t(aa)} \\ &= \frac{q_{t|0}(aa|aa)q_0(aa)}{q_{t|0}(aa|aa)q_0(aa) + q_{t|0}(aa|bb)q_0(bb)} = \frac{\frac{1}{4}(1+e^{-t})^2}{\frac{1}{4}(1+e^{-t})^2 + \frac{1}{4}(1-e^{-t})^2} = \frac{(1+e^{-t})^2}{2(1+e^{-2t})}, \end{aligned} \tag{40}$$

$$q_{0|t}(bb|aa) = 1 - q_{0|t}(aa|aa) = \frac{(1-e^{-t})^2}{2(1+e^{-2t})}, \tag{41}$$

$$q_{0|t}(aa|ab) = q_{0|t}(bb|ab) = \frac{1}{2}, \tag{42}$$

where (42) is derived from symmetry.

By using (40)–(42) and the general fact (for Markov processes)

$$q_{s|0,t}(\boldsymbol{x}_s|\boldsymbol{x}_0, \boldsymbol{x}_t) = \frac{q_{0,s,t}(\boldsymbol{x}_0, \boldsymbol{x}_s, \boldsymbol{x}_t)}{q_{0,t}(\boldsymbol{x}_0, \boldsymbol{x}_t)} = \frac{q_{s|0}(\boldsymbol{x}_s|\boldsymbol{x}_0)q_{t|0,s}(\boldsymbol{x}_t|\boldsymbol{x}_0, \boldsymbol{x}_s)}{q_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)} = \frac{q_{s|0}(\boldsymbol{x}_s|\boldsymbol{x}_0)q_{t|s}(\boldsymbol{x}_t|\boldsymbol{x}_s)}{q_{t|0}(\boldsymbol{x}_t|\boldsymbol{x}_0)}$$

for $0 \le s \le t$, we can compute $q_{s|t}(\cdot|aa)$ for any $s \in [0, t]$ as follows:

$$\begin{aligned} q_{s|t}(aa|aa) &= q_{0|t}(aa|aa)q_{s|0,t}(aa|aa, aa) + q_{0|t}(bb|aa)q_{s|0,t}(aa|bb, aa) \\ &= \frac{(1+e^{-t})^2}{2(1+e^{-2t})} \frac{\frac{1}{4}(1+e^{-s})^2\frac{1}{4}(1+e^{-(t-s)})^2}{\frac{1}{4}(1+e^{-t})^2} + \frac{(1-e^{-t})^2}{2(1+e^{-2t})} \frac{\frac{1}{4}(1-e^{-s})^2\frac{1}{4}(1+e^{-(t-s)})^2}{\frac{1}{4}(1-e^{-t})^2} \\ &= \frac{((1+e^{-s})^2 + (1-e^{-s})^2)(1+e^{-(t-s)})^2}{8(1+e^{-2t})} = \frac{(1+e^{-2s})(1+e^{-(t-s)})^2}{4(1+e^{-2t})}, \end{aligned} \tag{43}$$

$$\begin{aligned} q_{s|t}(bb|aa) &= q_{0|t}(aa|aa)q_{s|0,t}(bb|aa, aa) + q_{0|t}(bb|aa)q_{s|0,t}(bb|bb, aa) \\ &= \frac{(1+e^{-t})^2}{2(1+e^{-2t})} \frac{\frac{1}{4}(1-e^{-s})^2\frac{1}{4}(1-e^{-(t-s)})^2}{\frac{1}{4}(1+e^{-t})^2} + \frac{(1-e^{-t})^2}{2(1+e^{-2t})} \frac{\frac{1}{4}(1+e^{-s})^2\frac{1}{4}(1-e^{-(t-s)})^2}{\frac{1}{4}(1-e^{-t})^2} \\ &= \frac{((1-e^{-s})^2 + (1+e^{-s})^2)(1-e^{-(t-s)})^2}{8(1+e^{-2t})} = \frac{(1+e^{-2s})(1-e^{-(t-s)})^2}{4(1+e^{-2t})}, \end{aligned} \tag{44}$$

$$q_{s|t}(ab|aa) = q_{s|t}(ba|aa) = \frac{1}{2}(1 - q_{s|t}(aa|aa) - q_{s|t}(bb|aa)) \tag{45}$$

$$\begin{aligned} &= \frac{1}{2} - \frac{(1+e^{-2s})((1+e^{-(t-s)})^2 + (1-e^{-(t-s)})^2)}{8(1+e^{-2t})} \\ &= \frac{1}{2} - \frac{(1+e^{-2s})(1+e^{-2(t-s)})}{4(1+e^{-2t})} = \frac{1}{4} - \frac{e^{-2s} + e^{-2(t-s)}}{4(1+e^{-2t})}. \end{aligned} \tag{46}$$

We can also compute $q_{s|t}(aa|ab) = q_{s|t}(bb|ab)$ as

$$\begin{aligned} q_{s|t}(aa|ab) &= q_{0|t}(aa|ab)q_{s|0,t}(aa|aa, ab) + q_{0|t}(bb|ab)q_{s|0,t}(aa|bb, ab) \\ &= \frac{1}{2}\frac{\frac{1}{4}(1+e^{-s})^2\frac{1}{4}(1+e^{-(t-s)})(1-e^{-(t-s)})}{\frac{1}{4}(1+e^{-t})(1-e^{-t})} + \frac{1}{2}\frac{\frac{1}{4}(1-e^{-s})^2\frac{1}{4}(1+e^{-(t-s)})(1-e^{-(t-s)})}{\frac{1}{4}(1-e^{-t})(1+e^{-t})} \\ &= \frac{((1+e^{-s})^2 + (1-e^{-s})^2)(1-e^{-2(t-s)})}{8(1-e^{-2t})} = \frac{(1+e^{-2s})(1-e^{-2(t-s)})}{4(1-e^{-2t})} \\ &= \frac{1}{4} + \frac{e^{-2s} - e^{-2(t-s)}}{4(1-e^{-2t})}. \end{aligned} \tag{47}$$

Let us now compute the probabilities regarding the analytical sampler. To make it simple, let $q_{s|t}(x*|\cdot) := q_{s|t}(xa|\cdot) + q_{s|t}(xb|\cdot)$ represent marginals; $q_{s|t}(*y|\cdot)$ is defined similarly. By using this notation and (43)–(47), we have

$$\begin{aligned} p_{s|t}(aa|aa) &= q_{s|t}(a*|aa)q_{s|t}(*a|aa) = q_{s|t}(a*|aa)^2 = (q_{s|t}(aa|aa) + q_{s|t}(ab|aa))^2 \\ &= \left(\frac{(1+e^{-2s})(1+e^{-(t-s)})^2}{4(1+e^{-2t})} + \frac{1}{2} - \frac{(1+e^{-2s})(1+e^{-2(t-s)})}{4(1+e^{-2t})}\right)^2 \\ &= \left(\frac{2(1+e^{-2t}) + (1+e^{-2s})((1+e^{-(t-s)})^2 - (1+e^{-2(t-s)}))}{4(1+e^{-2t})}\right)^2 \\ &= \left(\frac{(1+e^{-2t}) + (1+e^{-2s})e^{-(t-s)}}{2(1+e^{-2t})}\right)^2 = \left(\frac{(1+e^{-(t+s)})(1+e^{-(t-s)})}{2(1+e^{-2t})}\right)^2 \qquad (48) \\ p_{s|t}(bb|aa) &= q_{s|t}(b*|aa)q_{s|t}(*b|aa) = q_{s|t}(b*|aa)^2 = (q_{s|t}(bb|aa) + q_{s|t}(ba|aa))^2 \\ &= \left(\frac{(1+e^{-2s})(1-e^{-(t-s)})^2}{4(1+e^{-2t})} + \frac{1}{2} - \frac{(1+e^{-2s})(1+e^{-2(t-s)})}{4(1+e^{-2t})}\right)^2 \\ &= \left(\frac{2(1+e^{-2t}) + (1+e^{-2s})((1-e^{-(t-s)})^2 - (1+e^{-2(t-s)}))}{4(1+e^{-2t})}\right)^2 \\ &= \left(\frac{(1+e^{-2t}) - (1+e^{-2s})e^{-(t-s)}}{2(1+e^{-2t})}\right)^2 = \left(\frac{(1-e^{-(t+s)})(1-e^{-(t-s)})}{2(1+e^{-2t})}\right)^2 \qquad (49) \end{aligned}$$

Let us compute the sum of (48) and (49) as we use it later:

$$\begin{aligned} &p_{s|t}(aa|aa) + p_{s|t}(bb|aa) \\ &= \left(\frac{(1+e^{-(t+s)})(1+e^{-(t-s)})}{2(1+e^{-2t})}\right)^2 + \left(\frac{(1-e^{-(t+s)})(1-e^{-(t-s)})}{2(1+e^{-2t})}\right)^2 \\ &= \frac{((1+e^{-(t+s)})(1+e^{-(t-s)}))^2 + ((1-e^{-(t+s)})(1-e^{-(t-s)}))^2}{4(1+e^{-2t})^2} \\ &= \frac{(1+e^{-2t}+e^{-(t+s)}+e^{-(t-s)})^2 + (1+e^{-2t}-e^{-(t+s)}-e^{-(t-s)})^2}{4(1+e^{-2t})^2} \\ &= \frac{(1+e^{-2t})^2 + (e^{-(t+s)}+e^{-(t-s)})^2}{2(1+e^{-2t})^2} = \frac{1}{2} + \frac{(e^{-(t+s)}+e^{-(t-s)})^2}{2(1+e^{-2t})^2}. \qquad (50) \end{aligned}$$

Next, $p_{s|t}(aa|ab)$ is the product of two marginals — $q_{s|t}(a*|ab)$ and $q_{s|t}(*a|ab)$, which can be computed as follows:

$$\begin{aligned} p_{s|t}(a*|ab) &= q_{0|t}(aa|ab)q^1_{s|0,t}(a|a,a) + q_{0|t}(bb|ab)q^1_{s|0,t}(a|b,a) \\ &= \frac{1}{2}\frac{\frac{1}{2}(1+e^{-s})\frac{1}{2}(1+e^{-(t-s)})}{\frac{1}{2}(1+e^{-t})} + \frac{1}{2}\frac{\frac{1}{2}(1-e^{-s})\frac{1}{2}(1+e^{-(t-s)})}{\frac{1}{2}(1-e^{-t})} \\ &= \frac{((1+e^{-s})(1-e^{-t}) + (1-e^{-s})(1+e^{-t}))(1+e^{-(t-s)})}{4(1-e^{-2t})} \\ &= \frac{(1-e^{-(t+s)})(1+e^{-(t-s)})}{2(1-e^{-2t})} = \frac{1}{2} + \frac{e^{-(t-s)} - e^{-(t+s)}}{2(1-e^{-2t})}, \\ p_{s|t}(*a|ab) &= p_{s|t}(a*|ba) = p_{s|t}(b*|ab) = 1 - p_{s|t}(a*|ab) = \frac{1}{2} - \frac{e^{-(t-s)} - e^{-(t+s)}}{2(1-e^{-2t})}, \end{aligned}$$

where the latter derivation is from the symmetries of the two dimensions and two characters. By using these, we have

$$p_{s|t}(aa|ab) = p_{s|t}(a*|ab)p_{s|t}(*a|ab) = \frac{1}{4} - \left(\frac{e^{-(t-s)} - e^{-(t+s)}}{2(1-e^{-2t})}\right)^2. \qquad (51)$$

Let us consider iteratively denoising from q_T by using $p_{s|t}$. For an $\epsilon > 0$ and nonnegative integers $n \le T/\epsilon - 1$, define

$$p_T^\epsilon := p_T, \qquad p_{T-(n+1)\epsilon}^\epsilon := \mathbb{E}_{\boldsymbol{x} \sim p_{T-n\epsilon}^\epsilon}\left[p_{T-(n+1)\epsilon|T-n\epsilon}(\cdot|\boldsymbol{x})\right], \quad n = 0, 1, \dots.$$

Our goal is to estimate the difference between $p_{T-n\epsilon}^\epsilon$ and $q_{T-n\epsilon}$ for each n. Let us fix n and set $t = T - n\epsilon$ when computing $p_{t-\epsilon}^\epsilon$ in terms of p_t^ϵ. Because of the symmetry, $p_t^\epsilon(aa) = p_t^\epsilon(bb)$ and $p_t^\epsilon(ab) = p_t^\epsilon(ba) = \frac{1}{2} - p_t^\epsilon(aa)$ hold in general. Therefore, by using (50) and (51), we have

$$\begin{aligned}
p_{t-\epsilon}^\epsilon(aa) &= p_{t-\epsilon|t}(aa|aa)p_t^\epsilon(aa) + p_{t-\epsilon|t}(aa|bb)p_t^\epsilon(bb) + p_{t-\epsilon|t}(aa|ab)p_t^\epsilon(ab) + p_{t-\epsilon|t}(aa|ba)p_t^\epsilon(ba) \\
&= p_{t-\epsilon|t}(aa|aa)p_t^\epsilon(aa) + p_{t-\epsilon|t}(bb|aa)p_t^\epsilon(aa) + 2p_{t-\epsilon|t}(aa|ab)\left(\frac{1}{2} - p_t^\epsilon(aa)\right) \\
&= p_{t-\epsilon|t}(aa|ab) + (p_{t-\epsilon|t}(aa|aa) + p_{t-\epsilon|t}(bb|aa) - 2p_{t-\epsilon|t}(aa|ab))p_t^\epsilon(aa) \\
&= \frac{1}{4} - \frac{(e^{-\epsilon} - e^{-(2t-\epsilon)})^2}{4(1-e^{-2t})^2} + \left(\frac{(e^{-\epsilon} + e^{-(2t-\epsilon)})^2}{2(1+e^{-2t})^2} + \frac{(e^{-\epsilon} - e^{-(2t-\epsilon)})^2}{2(1-e^{-2t})^2}\right)p_t^\epsilon(aa). \qquad (52)
\end{aligned}$$

To compare it with $q_{t-\epsilon}$, we also compute a similar recurrence equation by replacing p's with q's and using (45)–(47):

$$\begin{aligned}
q_{t-\epsilon}(aa) &= q_{t-\epsilon|t}(aa|ab) + (q_{t-\epsilon|t}(aa|aa) + q_{t-\epsilon|t}(bb|aa) - 2q_{t-\epsilon|t}(aa|ab))q_t(aa) \\
&= \frac{1}{4} - \frac{e^{-2\epsilon} - e^{-2(t-\epsilon)}}{4(1-e^{-2t})} + \left(\frac{e^{-2\epsilon} + e^{-2(t-\epsilon)}}{2(1+e^{-2t})} + \frac{e^{-2\epsilon} - e^{-2(t-\epsilon)}}{2(1-e^{-2t})}\right)q_t(aa) \qquad (53)
\end{aligned}$$

Let us now compute some quantities regarding the coefficients in (52) and (53).

$$\begin{aligned}
&\frac{e^{-2\epsilon} - e^{-2(t-\epsilon)}}{1-e^{-2t}} - \frac{(e^{-\epsilon} - e^{-(2t-\epsilon)})^2}{(1-e^{-2t})^2} \\
&= \frac{(e^{-2\epsilon} - e^{-2(t-\epsilon)})(1-e^{-2t}) - (e^{-\epsilon} - e^{-(2t-\epsilon)})^2}{(1-e^{-2t})^2} \\
&= \frac{(e^{-2\epsilon} - e^{-2(t-\epsilon)} - e^{-2(t+\epsilon)} + e^{-2(2t-\epsilon)}) - (e^{-\epsilon} - e^{-(2t-\epsilon)})^2}{(1-e^{-2t})^2} \\
&= -\frac{(e^{-(t-\epsilon)} - e^{-(t+\epsilon)})^2}{(1-e^{-2t})^2} = -\frac{e^{-2t}}{(1-e^{-2t})^2}(e^{\epsilon} - e^{-\epsilon})^2, \qquad (54)
\end{aligned}$$

$$\begin{aligned}
&\frac{e^{-2\epsilon} + e^{-2(t-\epsilon)}}{1+e^{-2t}} - \frac{(e^{-\epsilon} + e^{-(2t-\epsilon)})^2}{(1+e^{-2t})^2} \\
&= \frac{(e^{-2\epsilon} + e^{-2(t-\epsilon)})(1+e^{-2t}) - (e^{-\epsilon} + e^{-(2t-\epsilon)})^2}{(1+e^{-2t})^2} \\
&= \frac{(e^{-2\epsilon} + e^{-2(t-\epsilon)} + e^{-2(t+\epsilon)} + e^{-2(2t-\epsilon)}) - (e^{-\epsilon} + e^{-(2t-\epsilon)})^2}{(1+e^{-2t})^2} \\
&= \frac{(e^{-(t-\epsilon)} - e^{-(t+\epsilon)})^2}{(1+e^{-2t})^2} = \frac{e^{-2t}}{(1+e^{-2t})^2}(e^{\epsilon} - e^{-\epsilon})^2, \qquad (55)
\end{aligned}$$

$$\begin{aligned}
&\frac{e^{-2\epsilon} + e^{-2(t-\epsilon)}}{1+e^{-2t}} + \frac{e^{-2\epsilon} - e^{-2(t-\epsilon)}}{1-e^{-2t}} \\
&= \frac{(e^{-2\epsilon} + e^{-2(t-\epsilon)})(1-e^{-2t}) + (e^{-2\epsilon} - e^{-2(t-\epsilon)})(1+e^{-2t})}{1-e^{-4t}} \\
&= 2 + \frac{2(e^{-2\epsilon} - e^{-2(2t-\epsilon)}) - 2(1-e^{-4t})}{1-e^{-4t}} \\
&= 2 + \frac{2(1+e^{2(2t-\epsilon)})}{1-e^{-4t}}(e^{-2\epsilon} - 1). \qquad (56)
\end{aligned}$$

We shall evaluate the difference $\Delta_t^\epsilon := q_t(aa) - p_t^\epsilon(aa)$ by using (52)–(56) as follows:

$$\begin{aligned}
\Delta_{t-\epsilon}^\epsilon &= -\left(\frac{e^{-2\epsilon} - e^{-2(t-\epsilon)}}{4(1-e^{-2t})} - \frac{(e^{-\epsilon} - e^{-(2t-\epsilon)})^2}{4(1-e^{-2t})^2}\right) \\
&\quad + \left(\frac{e^{-2\epsilon} + e^{-2(t-\epsilon)}}{2(1+e^{-2t})} + \frac{e^{-2\epsilon} - e^{-2(t-\epsilon)}}{2(1-e^{-2t})}\right)(p_t^\epsilon(aa) + \Delta_t^\epsilon) \\
&\quad - \left(\frac{(e^{-\epsilon} + e^{-(2t-\epsilon)})^2}{2(1+e^{-2t})^2} + \frac{(e^{-\epsilon} - e^{-(2t-\epsilon)})^2}{2(1-e^{-2t})^2}\right) p_t^\epsilon(aa) \\
&= \frac{e^{-2t}}{4(1-e^{-2t})^2}(e^\epsilon - e^{-\epsilon})^2 + \left(\frac{e^{-2\epsilon} + e^{-2(t-\epsilon)}}{2(1+e^{-2t})} + \frac{e^{-2\epsilon} - e^{-2(t-\epsilon)}}{2(1-e^{-2t})}\right)\Delta_t^\epsilon \\
&\quad + \left(\frac{e^{-2\epsilon} + e^{-2(t-\epsilon)}}{2(1+e^{-2t})} - \frac{(e^{-\epsilon} + e^{-(2t-\epsilon)})^2}{2(1+e^{-2t})^2} + \frac{e^{-2\epsilon} - e^{-2(t-\epsilon)}}{2(1-e^{-2t})} - \frac{(e^{-\epsilon} - e^{-(2t-\epsilon)})^2}{2(1-e^{-2t})^2}\right) p_t^\epsilon(aa) \\
&= \frac{e^{-2t}}{4(1-e^{-2t})^2}(e^\epsilon - e^{-\epsilon})^2 + \left(1 + \frac{1+e^{2(2t-\epsilon)}}{1-e^{-4t}}(e^{-2\epsilon} - 1)\right)\Delta_t^\epsilon \\
&\quad + \left(\frac{e^{-2t}}{2(1+e^{-2t})^2} - \frac{e^{-2t}}{2(1-e^{-2t})^2}\right)(e^\epsilon - e^{-\epsilon})^2 p_t^\epsilon(aa) \\
&= \left(\frac{e^{-2t}}{2(1+e^{-2t})^2} p_t^\epsilon(aa) + \frac{e^{-2t}}{2(1-e^{-2t})^2}\left(\frac{1}{2} - p_t^\epsilon(aa)\right)\right)(e^\epsilon - e^{-\epsilon})^2 \\
&\quad + \left(1 + \frac{1+e^{2(2t-\epsilon)}}{1-e^{-4t}}(e^{-2\epsilon} - 1)\right)\Delta_t^\epsilon. \qquad (57)
\end{aligned}$$

Since $p_t^\epsilon(aa) = p_t^\epsilon(bb) \leq 1/2$, we have

$$\begin{aligned}
&\frac{e^{-2t}}{2(1+e^{-2t})^2} p_t^\epsilon(aa) + \frac{e^{-2t}}{2(1-e^{-2t})^2}\left(\frac{1}{2} - p_t^\epsilon(aa)\right) \\
&\geq \frac{1}{2}\min\left\{\frac{e^{-2t}}{2(1+e^{-2t})^2}, \frac{e^{-2t}}{2(1-e^{-2t})^2}\right\} = \frac{e^{-2t}}{4(1-e^{-2t})^2}.
\end{aligned}$$

Additionally, as the Taylor series of $(e^\epsilon - e^{-\epsilon})^2 = e^{2\epsilon} + e^{-2\epsilon} - 2$ is given by $\sum_{k=1}^\infty \frac{2}{(2k)!}(2\epsilon)^{2k}$, we especially have $(e^\epsilon - e^{-\epsilon})^2 \geq 4\epsilon^2$. Thus, we obtain

$$\begin{aligned}
&\left(\frac{e^{-2t}}{2(1+e^{-2t})^2} p_t^\epsilon(aa) + \frac{e^{-2t}}{2(1-e^{-2t})^2}\left(\frac{1}{2} - p_t^\epsilon(aa)\right)\right)(e^\epsilon - e^{-\epsilon})^2 \\
&\geq \frac{e^{-2t}}{4(1-e^{-2t})^2} \cdot 4\epsilon^2 = \frac{e^{-2t}}{(1-e^{-2t})^2}\epsilon^2. \qquad (58)
\end{aligned}$$

Also, since $e^{-2\epsilon} \geq 1 - 2\epsilon$, we have

$$1 + \frac{1+e^{2(2t-\epsilon)}}{1-e^{-4t}}(e^{-2\epsilon} - 1) \geq 1 - \frac{2(1+e^{2(2t-\epsilon)})}{1-e^{-4t}}\epsilon \geq 1 - \frac{4}{1-e^{-4t}}\epsilon. \qquad (59)$$

Suppose we are working on the time interval $[\delta, T]$ for some $\delta, T > 0$. Let us take $\epsilon \leq \delta/2$, then we have

$$1 - \frac{4}{1-e^{-4t}}\epsilon \geq 1 - \frac{4}{4t}\epsilon \geq 1 - \frac{\epsilon}{\delta} > 0. \qquad (60)$$

For (58), we have

$$\frac{e^{-2t}}{(1-e^{-2t})^2}\epsilon^2 \geq e^{-2t}\epsilon^2 \geq e^{-2T}\epsilon^2. \qquad (61)$$

By combining (57)–(61), we first see that Δ_t^ϵ is nonnegative for all $t = T - n\epsilon$ by induction on $n = 0, 1, \ldots$ (assuming $\epsilon \leq \delta/2$ and $t \in [\delta, T]$). Then, we obtain the following simple inequality:

$$\Delta_{t-\epsilon}^\epsilon \geq \left(1 - \frac{\epsilon}{\delta}\right)\Delta_t^\epsilon + e^{-2T}\epsilon^2$$

By recalling that $t = T - n\epsilon$, we can rewrite it as

$$\left(1 - \frac{\epsilon}{\delta}\right)^{-(n+1)}\Delta_{T-(n+1)\epsilon}^\epsilon \geq \left(1 - \frac{\epsilon}{\delta}\right)^{-n}\Delta_{T-n}^\epsilon + \left(1 - \frac{\epsilon}{\delta}\right)^{-(n+1)} e^{-2T}\epsilon^2.$$

Since $\Delta_T^\epsilon = 0$, we have

$$\Delta_{T-n\epsilon}^\epsilon \geq \left(1 - \frac{\epsilon}{\delta}\right)^n \sum_{k=1}^n \left(1 - \frac{\epsilon}{\delta}\right)^{-k} e^{-2T}\epsilon^2 = \sum_{k=0}^{n-1} \left(1 - \frac{\epsilon}{\delta}\right)^k e^{-2T}\epsilon^2. \tag{62}$$

Since $n \leq T/\epsilon$ and $(1 - 1/x)^x$ is increasing over $x > 1$, for $k = 0, \dots, n-1$, we have

$$\left(1 - \frac{\epsilon}{\delta}\right)^k \geq \left(1 - \frac{\epsilon}{\delta}\right)^n \geq \left(1 - \frac{\epsilon}{\delta}\right)^{T/\epsilon} = \left(\left(1 - \frac{\epsilon}{\delta}\right)^{\delta/\epsilon}\right)^{T/\delta} \geq \left(\left(1 - \frac{1}{2}\right)^2\right)^{T/\delta} = 2^{-2T/\delta},$$

where we have exploited the assumption $\epsilon \leq \delta/2$ (so that $\delta/\epsilon \geq 2$). By applying this to (62), we obtain

$$\Delta_{T-n\epsilon}^\epsilon \geq (2^{1/\delta}e)^{-2T} n\epsilon^2.$$

Now, let $\epsilon = (T - \delta)/N$ for the given N. Since $N \geq \frac{2(T-\delta)}{\delta}$ and so $\epsilon \leq \delta/2$, we have

$$\Delta_\delta^\epsilon = \Delta_{T-N\epsilon}^\epsilon \geq (2^{1/\delta}e)^{-2T} N\epsilon^2 = (2^{1/\delta}e)^{-2T} \frac{(T-\delta)^2}{N}.$$

Finally, as $d_{\mathrm{TV}}(q_\delta, p_\delta^{(T-\delta)/N}) \geq \Delta_\delta^\epsilon$, the constant $c = (2^{1/\delta}e)^{-2T}(T-\delta)^2$ satisfies (24). □

F Control variates

When we want to compute an expectation $\mathbb{E}[f(\boldsymbol{x})]$, instead of directly doing the Monte Carlo estimate $\frac{1}{N}\sum_{i=1}^N f(\boldsymbol{x}_i) \approx \mathbb{E}[f(\boldsymbol{x})]$, we can find a function $g \approx f$ such that $\mathbb{E}[g(\boldsymbol{x})]$ is tractable, and then do the Monte Carlo estimate for the remainder term:

$$\frac{1}{N}\sum_{i=1}^N (f(\boldsymbol{x}_i) - g(\boldsymbol{x}_i)) + \mathbb{E}[g(\boldsymbol{x})] \approx \mathbb{E}[f(\boldsymbol{x})]\,. \tag{63}$$

This left-hand side is still an unbiased estimator of $\mathbb{E}[f(\boldsymbol{x})]$, and ideally has a lower variance than the vanilla Monte Carlo estimator $\frac{1}{N}\sum_{i=1}^N f(\boldsymbol{x}_i)$ if $g \approx f$ is a good function approximation. The role of g in (63) is called a *control variate* [15, 35].

F.1 Marginal-matching product model as control variate

We briefly discuss how the product model $\overline{p}^\theta$ given in (13) satisfies the following favorable properties (already shown in Section B.2) for being control variate:

(i) it reasonably approximates $p_{s|t}^\theta(\cdot|\boldsymbol{x}_t)$, and

(ii) $\mathbb{E}_{\boldsymbol{x}\sim q}[g(\boldsymbol{x})]$ is easy to compute/approximate.

For the point (i), note that $\overline{p}^\theta$ is defined as the product model having the same marginal as p^θ. Since dimensionally independent modeling (when combined with multi-step sampling) works as in Theorem 1, $\overline{p}^\theta$ should approximate p^θ to a certain degree; see also Lemma 1 for quantitative understanding. The remainder $p^\theta - \overline{p}^\theta$ can then be regarded as the dimensional correlation captured by p^θ, with which we conduct a usual Monte Carlo integration.

Regarding (ii), given a product distribution $\overline{p}(\boldsymbol{x}) = \prod_{d=1}^D \overline{p}^d(x^d)$ over $\mathcal{X} = \mathcal{S}^D$, we can indeed compute $H(q, \overline{p})$ by a Monte Carlo integral using samples of η as

$$\begin{aligned} H(q, \overline{p}) &= \mathbb{E}_{\boldsymbol{x}_s\sim q}[-\log \overline{p}(\boldsymbol{x}_s)] = \mathbb{E}_\eta \mathbb{E}_{\boldsymbol{x}_s\sim q^\eta}[-\log \overline{p}(\boldsymbol{x}_s)] \\ &= \mathbb{E}_\eta[H(q^\eta, \overline{p})] = \mathbb{E}_\eta\left[-\sum_{d=1}^D \sum_{x_s^d\in\mathcal{S}} q^\eta(x_s^d)\log \overline{p}^d(x_s^d)\right]. \end{aligned} \tag{64}$$

While it still requires Monte Carlo with η to estimate this, it utilizes the product structure of each q^η and $\overline{p}$ for exactly computing $H(q^\eta, \overline{p})$. Thus, we heuristically expect it to be more accurate than the Monte Carlo estimate using samples from q.

F.2 Derivations of dimension-wise computable control variates for mixture model

Convex upper bound as control variate. To simplify the notation and situation, suppose we are given probability distributions $q = \mathbb{E}_\eta[q^\eta]$ and $p^\theta = \mathbb{E}_\lambda\left[p^{\theta,\lambda}\right]$, where q^η and $p^{\theta,\lambda}$ are product distributions, i.e., we have

$$q^\eta(\boldsymbol{x}) = \prod_{d=1}^{D} q^{\eta,d}(x^d), \qquad p^{\theta,\lambda}(\boldsymbol{x}) = \prod_{d=1}^{D} p^{\theta,\lambda,d}(x^d).$$

By letting H be the (cross) entropy, we want to minimize

$$D_{\mathrm{KL}}(q \| p^\theta) = H(q, p^\theta) - H(q) = \mathbb{E}_{\boldsymbol{x}\sim q}\left[-\log p^\theta(\boldsymbol{x})\right] - \mathbb{E}_{\boldsymbol{x}\sim q}[-\log q(\boldsymbol{x})]\,.$$

Since q is fixed, we simply want to minimize

$$H(q, p^\theta) = \mathbb{E}_{\boldsymbol{x}\sim q}\left[-\log p^\theta(\boldsymbol{x})\right] = \mathbb{E}_\eta \mathbb{E}_{\boldsymbol{x}\sim q^\eta}\left[-\log p^\theta(\boldsymbol{x})\right]$$

with regard to θ. However, it might have a high variance when we only sample $\boldsymbol{x} \sim q$ and execute Monte Carlo. One option is using the following upper bound ike negative ELBO given by Jensen's inequality (convex inequality) as a control variate:

$$-\log p^\theta(\boldsymbol{x}) = -\log \mathbb{E}_\lambda\left[p^{\theta,\lambda}(\boldsymbol{x})\right] \leq \mathbb{E}_\lambda\left[-\log p^{\theta,\lambda}(\boldsymbol{x})\right]\,.$$

Indeed, its expectation regarding $\boldsymbol{x} \sim q$ is dimension-wise computable as

$$\begin{aligned}
&\mathbb{E}_{\boldsymbol{x}\sim q}\mathbb{E}_\lambda\left[-\log p^{\theta,\lambda}(\boldsymbol{x})\right] \\
&= \mathbb{E}_\eta \mathbb{E}_{\boldsymbol{x}\sim q^\eta}\mathbb{E}_\lambda\left[-\log p^{\theta,\lambda}(\boldsymbol{x})\right] = \mathbb{E}_\eta \mathbb{E}_\lambda \mathbb{E}_{\boldsymbol{x}\sim q^\eta}\left[-\log p^{\theta,\lambda}(\boldsymbol{x})\right] \\
&= \mathbb{E}_\eta \mathbb{E}_\lambda \sum_{d=1}^{D} \mathbb{E}_{x^d \sim q^{\eta,d}}\left[-\log p^{\theta,\lambda,d}(x^d)\right] = \mathbb{E}_\eta \mathbb{E}_\lambda \left[-\sum_{d=1}^{D} \sum_{x^d} q^{\eta,d}(x^d) \log p^{\theta,\lambda,d}(x^d)\right],
\end{aligned}$$

which does not require Monte Carlo sampling of $\boldsymbol{x}$. Overall, we can decompose the computation as

$$H(q, p^\theta) = \underbrace{\mathbb{E}_{\boldsymbol{x}\sim q}\left[-\log p^\theta(\boldsymbol{x}) + \mathbb{E}_\lambda\left[\log p^{\theta,\lambda}(\boldsymbol{x})\right]\right]}_{\text{Monte Carlo approximation}} + \underbrace{\mathbb{E}_{\boldsymbol{x}\sim q}\mathbb{E}_\lambda\left[-\log p^{\theta,\lambda}(\boldsymbol{x})\right]}_{\text{dim-wise computable}}\,.$$

Marginal control variate. The previous convex upper bound seems good, but since

$$\mathbb{E}_{\boldsymbol{x}\sim q}\mathbb{E}_\lambda\left[-\log p^{\theta,\lambda}(\boldsymbol{x})\right] = \mathbb{E}_\lambda\left[H(q, p^{\theta,\lambda})\right] \geq \inf_\lambda H(q, p^{\theta,\lambda}),$$

it might be a very loose bound (we want the mixture to outperform the best product distribution $p^{\theta,\lambda}$). To make it more practical, we can consider its dimension-wise tractable lower bound as follows:

$$\mathbb{E}_{\boldsymbol{x}\sim q}\mathbb{E}_\lambda\left[-\log p^{\theta,\lambda}(\boldsymbol{x})\right] = \mathbb{E}_\eta \sum_{d=1}^{D} \mathbb{E}_{x^d\sim q^{\eta,d}}\mathbb{E}_\lambda\left[-\log p^{\theta,\lambda,d}(x_d)\right] \geq -\mathbb{E}_\eta \sum_{d=1}^{D} \mathbb{E}_{x^d\sim q^{\eta,d}} \log \mathbb{E}_\lambda\left[p^{\theta,\lambda,d}(x_d)\right]\,,$$

which is given by Jensen's inequality as well. Therefore, if we define the product distribution

$$\overline{p}^\theta(\boldsymbol{x}) = \prod_{d=1}^{D} \overline{p}^{\theta,d}(x_d), \qquad \overline{p}^{\theta,d}(x_d) = \mathbb{E}_\lambda\left[\overline{p}^{\theta,d}(x_d)\right]\,,$$

we have $\mathbb{E}_{\boldsymbol{x}\sim q}\mathbb{E}_\lambda\left[-\log p^{\theta,\lambda}(\boldsymbol{x})\right] \leq \mathbb{E}_{\boldsymbol{x}\sim q}\left[-\log \overline{p}^\theta(\boldsymbol{x})\right]$ and this alternative is also dimension-wise computable. Since p^θ and $\overline{p}^\theta$ coincides in each one-dimensional marginal, the difference between these two can be regarded as the result of dimensional correlation.

Therefore, we propose the following decomposition, which is also discussed in Section B.2:

$$H(q, p^\theta) = \underbrace{\mathbb{E}_{\boldsymbol{x}\sim q}\left[-\log p^\theta(\boldsymbol{x}) + \log \overline{p}^\theta(\boldsymbol{x})\right]}_{\text{Monte Carlo approximation}} + \underbrace{\mathbb{E}_{\boldsymbol{x}\sim q}\left[-\log \overline{p}^\theta(\boldsymbol{x})\right]}_{\text{dim-wise computable}}\,.$$

F.3 Product teacher model as control variate

For two models with the same marginals, we have the following proposition:

Proposition 7. *Let q, $\tilde{q}$ be probability distributions on $\mathcal{X} = \mathcal{S}^D$ with the same marginals $q^d = \tilde{q}^d$. Then, for a product distribution $p(\boldsymbol{x}) = \prod_d p^d(x^d)$ over $\mathcal{X}$, we have $H(q,p) = H(\tilde{q},p)$.*

Proof. It suffices to prove $H(q,p)$ can be computed only by using the marginals q^d. Indeed, we have

$$\mathbb{E}_{\boldsymbol{x}\sim q}[\log p(\boldsymbol{x})] = \mathbb{E}_{\boldsymbol{x}\sim q}\left[\sum_{d=1}^{D} \log p^d(x^d)\right] = \sum_{d=1}^{D} \mathbb{E}_{\boldsymbol{x}\sim q}\left[\log p^d(x^d)\right] = \sum_{d=1}^{D}\sum_{x^d} q^d(x^d)\log p^d(x^d),$$

and it yields the desired conclusion. □

From this proposition, under $p_{0|t}^{\psi,d} \approx q_{0|t}^d$ and the fact that $\overline{p}^\theta$ is a product model, we have

$$\mathbb{E}_{\boldsymbol{x}_t\sim q_t}\left[H(q_{0|t}(\cdot|\boldsymbol{x}_t), \overline{p}_{0|t}^\theta(\cdot|\boldsymbol{x}_t))\right] \approx \mathbb{E}_{\boldsymbol{x}_t\sim q_t}\left[H(p_{0|t}^\psi(\cdot|\boldsymbol{x}_t), \overline{p}_{0|t}^\theta(\cdot|\boldsymbol{x}_t))\right].$$

Since $H(p_1, p_2) = D_{\mathrm{KL}}(p_1 \,\|\, p_2) + H(p_2,p_2)$ this right-hand side can be rewritten as

$$\mathbb{E}_{\boldsymbol{x}_t\sim q_t}\left[H(p_{0|t}^\psi(\cdot|\boldsymbol{x}_t), \overline{p}_{0|t}^\theta(\cdot|\boldsymbol{x}_t))\right] = \mathbb{E}_{\boldsymbol{x}_t\sim q_t}\left[D_{\mathrm{KL}}(p_{0|t}^\psi(\cdot|\boldsymbol{x}_t) \,\|\, \overline{p}_{0|t}^\theta(\cdot|\boldsymbol{x}_t)))\right] + const.,$$

where the constant term is independent of θ. Since the KL divergence between two product distributions decomposes into the sum of the KL divergence between each marginal, we obtain the approximation (15).

G Experimental details

G.1 Sampling schemes

In the experiments, we use the following two sampling schemes when evaluating the already trained product teacher model.

$\boldsymbol{\tau}$-leaping. In Campbell et al. [6], the authors first approximate the infinitesimal transition rate by using each marginal $p_{0|t}^{\psi,d}$. Indeed, the transition rate can be represented only with $q_{0|t}^d$ and does not require the joint conditional distribution [6, Proposition 3]. After estimating the transition rate, they conduct a dimensionally parallel sampling method called τ-leaping [14] coming from computational chemistry. Simply put, τ-leaping is a sort of generalization of the Euler method for solving the backward SDE, exploiting the ordinal structure of $\mathcal{S}$. We omit the corrector steps; the τ-leaping in Table 1 corresponds to τLDR-0 in Campbell et al. [6].

Analytical sampling. Although the τ-leaping (or Euler method) is efficient with a large NFE, we find that it deteriorates when we reduce the NFE seemingly due to the discretization error. The analytical sampling [46], which is simply a parallel exact sampling of each dimension given as

$$q_{s|t}^d(x_s^d|\boldsymbol{x}_t) = \sum_{x_0^d} q_{s|0,t}^d(x_s^d|x_0^d, x_t^d) q_{0|t}^d(x_0^d|\boldsymbol{x}_t) \approx \sum_{x_0^d} q_{s|0,t}^d(x_s^d|x_0^d, x_t^d) p_{0|t}^{\psi,d}(x_0^d|\boldsymbol{x}_t), \tag{65}$$

does not suffer so much from the discretization. This is also mentioned in Gu et al. [17] as a fast inference strategy, though they do not discuss dimensional correlations.

Note that these schemes are both dimensionally independent in the sense of (1) while not explicitly modeling $p_{s|t}$. Indeed, the dimensional independence is ubiquitous even when modeling $p_{s|t}$ implicitly. First, the reparametrization $p_{s|t}(\boldsymbol{x}_s|\boldsymbol{x}_t) = \sum_{\boldsymbol{x}_0} p_{0|t}(\boldsymbol{x}_0|\boldsymbol{x}_t) q_{s|0,t}(\boldsymbol{x}_s|\boldsymbol{x}_0, \boldsymbol{x}_t)$ [2, 17], also used in analytical sampling, is dimensionally independent, provided that $p_{0|t}(\cdot|\boldsymbol{x}_t)$ is given by a product model and the forward diffusion is dimensionally independent. Second, we can apparently avoid the heuristic in the above modeling through the estimation of the transition rate in the continuous-time discrete diffusion [6, Proposition 3], but the existing sampling schemes of $\boldsymbol{x}_s$ given $\boldsymbol{x}_t$ in continuous-time settings including τ-leaping [6] and the Euler-based method [46, 29] are still dimensionally independent.

Sampling in the actual experiment given an NFE N is as follows: We first set the timesteps $0 = t_0 < t_1 < \cdots < t_N = 1$, with $t_i = 0.01 + 0.99 \times \frac{i-1}{N-1}$ for $i \geq 1$. Given a terminal noise $\boldsymbol{x}_{t_N}$, we sample $\boldsymbol{x}_{t_i}$ with our $p_{t_i|t_{i+1}}$ iteratively for $i = N-1, N-2, \dots, 1$. Finally, we sample $\boldsymbol{x}_0 \in \operatorname{argmax} p^{\psi}_{0|t_1}(\cdot|\boldsymbol{x}_{t_1})$ when using the teacher product model and $\boldsymbol{x}_0 \in \operatorname{argmax} p^{\theta}_{0|t_1}(\cdot|\boldsymbol{x}_{t_1}; \lambda)$ with a random λ when using the student mixture model.

G.2 Implementation and training

As explained in Section 5, the state-space has $D = 3 \times 32 \times 32$ dimensions, and each dimension has 256 possibilities of pixel values which corresponds to $\mathcal{S} = \{0, \dots, 255\}$. The forward diffusion process is defined through a discretized Gaussian transition rate with $T = 1$ [6, Section E].

All the models are based on the implementation explained in Campbell et al. [6, Section H.2], where $p^{\psi}_{0|t}$ is parameterized with a U-net [20] that has feature resolutions from 32×32 to 4×4. Since the output of the original U-net architecture [20] is a D-dimensional sequence (in $\mathcal{S}^D$) rather than D marginal distributions, Campbell et al. [6] adjusted the network so that it first outputs a Gaussian distribution over the real line for each marginal and then normalized it to obtain a distribution over $\mathcal{S}$. The time t in their implementation is passed to a transformer-based positional embedding, and this embedding is fed to the up-/down-sampling layers of the U-net after passing through SiLU-activated linear layers [12]. See Campbell et al. [6, Section H.2] and their GitHub repository for more details on the original implementation. All the models output the estimation of $q_{0|t}$, and we conduct denoising from time t to time s by using the dimension-wise analytical sampling (65), except for the τ-leaping benchmark in Table 1.

The only change we made on the architecture is the insertion of λ. We sample λ from the uniform distribution over $[0, 1]$, so we can basically use the same embedding architecture as the time t. For the down-sampling layers, the embedding of λ is concatenated with the time embedding, and then fed to the linear layers. After the linear layers, similarly to the time embedding, it is added to the latent vector of the image. For the up-sampling layers, we concatenate the embeddings of λ, t, and the pixel-wise average of the 4×4 resolution latent tensor, and the remaining process is the same as for the down-sampling layers.

Since our model is an expansion of the original model for p^{ψ}, we trained (finetuned) our student model p^{θ} from the checkpoint of p^{ψ}. The bias terms and the final layers concerning the embeddings of λ are zero-intialized, and the rest are randomly intialized following the default setting of the original model.

For training, we followed the original setting in terms of the use of Adam optimizer and the learning rate 2×10^{-4} as well as other hyperparameters. The two primary differences in training are loss functions and the training steps/minibatch size (due to the Monte Carlo for λ). For the former point, we basically used

$$\mathcal{L}_{\mathrm{distil}}(\theta; \psi, q_\delta, \delta) + \mathcal{L}_{\mathrm{consis}}(\theta; \psi, q_t, 0, t - \Delta t, t) + \alpha_t \mathcal{L}_{\mathrm{corr}}(\theta; t) + \mathcal{L}_{\mathrm{marginal}}(\theta; \psi, q_t, t),$$

with techniques described in Section B. Additional details are as follows.

- Sampling from q_δ and q_t is based on the same sample of $\boldsymbol{x}_0 \sim q_0$.
- $\delta = 0.01$ with probability $1/2$; otherwise δ is taken uniformly from $[0.01, 0.02]$.
- Δt is sampled from a log-uniform distribution over $[0.001, 0.01]$; t is then sampled uniformly from $[0.01 + \Delta t, 1]$.
- We can use several α_t as in the ablation study in the following section. In the main model p^{θ} given in Table 1, we used the following sigmoid-based function as α_t:

$$g(t) = \frac{1}{1 + \exp(10 - 20t)}. \tag{66}$$

Regarding the training steps/minibatch details, the original teacher model checkpoint had been trained for 2M steps, where each step uses 128 images from the CIFAR-10 dataset as a minibatch. We stopped all the trainings in 320K steps (without warm-ups). Each step uses a minibatch of $128/L$ images from the CIFAR-10 dataset, where L is a batch size for λ in the Monte Carlo estimates; we set $M = N = L$ in (11). $L = 16$ is adopted in our model in Table 1, while the ablation study in the following section compares various choices of L.

Finally, for evaluation, we measured FID and IS with the PyTorch-based implementation[2] following Campbell et al. [6].

G.3 Ablation study

Table 2: Ablation study on α_t and the use of control variates.

Method	NFE 10		NFE 20		NFE 40	
	FID	IS	FID	IS	FID	IS
p^ψ + analytical	32.61	7.59±0.10	12.36	8.55±0.13	**8.01**	**8.77±0.09**
$\alpha_t = 0$	26.23	8.02±0.09	11.55	**8.59±0.07**	9.01	8.65±0.14
$\alpha_t = 0$, w/o CV	44.09	6.79±0.10	26.16	7.54±0.10	22.20	7.72±0.08
$\alpha_t = 1$	24.14	7.54±0.08	12.30	8.06±0.07	10.32	8.14±0.10
$\alpha_t = 1$, w/o CV	26.92	8.12±0.08	13.77	8.57±0.14	10.59	8.66±0.05
$\alpha_t = t$	24.21	8.10±0.11	10.85	8.55±0.08	9.27	8.51±0.10
$\alpha_t = g(t)$ (see (66))	**22.77**	**8.19±0.08**	**10.07**	8.54±0.12	9.01	8.42±0.11

As an ablation study, we compared several loss functions, mainly changing α_t, which controls the degree of dimensional correlations we aim to learn from datapoints. We also investigate whether the use of control variates is effective. The results are shown in Table 2, where "w/o CV" means that the control variates are not used in training. The efficiency of control variates is consistent, while $\alpha_t = 0$ and $\alpha_t = 1$ have pros and cons. Non-constant functions of α_t work better, partially matching the hypothesis discussed at the end of Section 5.

Table 3: Ablation study on the Monte Carlo sample size of λ.

Method	NFE 10		NFE 20		NFE 40	
	FID	IS	FID	IS	FID	IS
p^ψ + analytical	32.61	7.59±0.10	12.36	8.55±0.13	**8.01**	**8.77±0.09**
$L = 2$	27.29	8.00±0.01	11.42	**8.67±0.12**	8.94	8.64±0.09
$L = 4$	24.94	8.05±0.14	10.66	8.60±0.11	8.90	8.59±0.07
$L = 8$	22.77	8.19±0.08	10.07	8.54±0.12	9.01	8.42±0.11
$L = 16$	20.64	**8.29±0.13**	**9.77**	8.52±0.08	9.66	8.28±0.10
$L = 32$	20.25	8.28±0.13	9.93	8.44±0.10	9.91	8.26±0.13
$L = 64$	**19.26**	8.13±0.10	10.13	8.26±0.11	10.59	8.02±0.15

Additionally, we compared different batch-sizes of λ in Table 3 (also see the end of the previous section). The non-constant $\alpha_t = g(t)$ is used in all the setteings. L in the table represents the batch size of λ in Monte Carlo sampling. There is a certain trade-off between FID and IS in 10- or 20-step sampling; we can expect better FID with larger L (smaller data batch) while smaller L tends to result in better IS.

[2] `https://github.com/w86763777/pytorch-image-generation-metrics`, which got renamed from `pytorch-gan-metrics` to `pytorch-image-generation-metrics`.