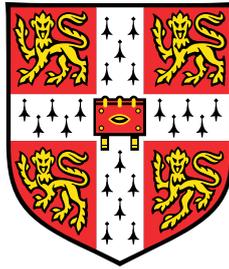


# Controlled Sampling in Diffusion Models



**Pablo Monteagudo Lago**

Department of Engineering  
University of Cambridge

This dissertation is submitted for the degree of  
*Master of Philosophy in Machine Learning and Machine Intelligence*

Churchill College

August 2024



To everyone who supported me throughout this journey.



## Declaration

I, Pablo Monteagudo Lago of Churchill College, being a candidate for the MPhil in Machine Learning and Machine Intelligence, hereby declare that this report and the work described in it are my own work, unaided except as may be specified below, and that the report does not contain material that has already been used to any substantial extent for a comparable purpose.

**Software Declaration:** The experimental set-up in this work is adapted from the implementation of the Controlled Monte Carlo sampler (Vargas et al., 2024), which was modified by including the methods presented in this dissertation, along with additional visualisations, experiments and benchmarks. Besides, the Many Well benchmark, implemented in Midgley et al. (2022), is also used. The repository can be found at [https://github.com/pablomlago/controlled\\_diffusion](https://github.com/pablomlago/controlled_diffusion).

**Word count:** 10798.

Pablo Monteagudo Lago  
August 2024



## **Acknowledgements**

First, I would like to extend my gratitude to my parents, my sister Mariña and my girlfriend Vera for their unconditional support throughout this journey.

Also, I would like to acknowledge my supervisor, Prof. José Miguel Hernández-Lobato for his guidance.

Last, but not least, I want to thank my friends from Cambridge who made this master's degree a once-in-a-lifetime experience.



## Abstract

Diffusion models have become the state-of-the-art family for generative modelling, enabling the generation of high-quality samples from complex distributions. However, these models generally require a substantial amount of training data to achieve optimal performance, thus limiting their applicability in domains where training data is scarce. On the other hand, if an unnormalised density for the target distribution is available, samples can be generated by relying on Monte Carlo or variational inference methods, even if no training data is available.

Since these scenarios are not mutually exclusive, we consider the problem of improving the quality of the samples generated by a sub-optimal diffusion model, possibly trained in insufficient data, by leveraging knowledge of an unnormalised target density. This problem can be formulated from the broader perspective of divergence minimisation for path measures defined in terms of controlled stochastic differential equations which transport the target distribution to a simple prior distribution, and vice versa. To this end, we propose two sampling methods: the Controlled Ancestral sampler and the Controlled Annealed Langevin sampler, which rely on controls, parameterised by neural networks, which can be trained, in an end-to-end fashion, to improve the quality of the samples generated by the diffusion model.



# Table of contents

<b>Nomenclature</b>	<b>xiii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Contributions . . . . .	2
1.2 Outline . . . . .	2
<b>2 Background</b>	<b>3</b>
2.1 Diffusion Probabilistic Models . . . . .	3
2.2 Diffusion with Stochastic Differential Equations . . . . .	6
2.3 Sampling Error . . . . .	8
2.4 Sampling Problem . . . . .	10
2.5 Related Works . . . . .	11
2.5.1 Time-Reversed Diffusion Sampler (DIS) . . . . .	13
2.5.2 Controlled Monte Carlo Diffusion Sampler (CMCD) . . . . .	14
<b>3 Controlled Sampling from Diffusion Models</b>	<b>15</b>
3.1 Motivation . . . . .	16
3.2 Controlled Ancestral Sampler . . . . .	18
3.3 Controlled Annealed Langevin Sampler . . . . .	21
3.4 Performance Metrics . . . . .	22
3.4.1 Estimation of Normalisation Constants . . . . .	22
3.4.2 Effective Sample Size . . . . .	23
3.4.3 Wasserstein Distance . . . . .	24
3.4.4 Benchmark Targets . . . . .	25
3.5 Experimental Settings . . . . .	26
3.5.1 Score-model Training . . . . .	26
3.5.2 Control Architecture . . . . .	28
3.6 Evaluation . . . . .	29

---

3.6.1	2-D GMM . . . . .	30
3.6.2	1-D GMM . . . . .	33
3.6.3	Funnel . . . . .	36
3.6.4	Many Well . . . . .	38
3.7	Analysis . . . . .	40
3.7.1	Importance of Control Tying . . . . .	41
3.7.2	Inaccurate Score Estimation . . . . .	42
3.7.3	Effective Sample Sizes . . . . .	43
3.7.4	Varying Number of Discretisation Steps . . . . .	44
3.8	Summary . . . . .	44
<b>4</b>	<b>Conclusions</b>	<b>47</b>
4.1	Limitations . . . . .	47
4.2	Future Works . . . . .	48
	<b>References</b>	<b>49</b>
	<b>Appendix A Additional Figures</b>	<b>55</b>

# Nomenclature

## Acronyms / Abbreviations

AIS	Annealed Importance Sampling
CALS	Controlled Annealed Langevin Sampler
CAS	Controlled Ancestral Sampler
CMCD	Controlled Monte Carlo Diffusion sampler
DDPM	Denosing Diffusion Probabilistic Models
DIS	Time-Reversed Diffusion Sampler
DSM	Denosing Score-Matching
EM	Euler-Maruyama
IS	Importance Sampling
KL	Kullback–Leibler
LV	Log-Variance
MC	Monte Carlo
MCMC	Markov Chain Monte Carlo
RND	Radon-Nikodym Derivative
SDE	Stochastic Differential Equation
VE	Variance Exploding
VI	Variational Inference

VLB Variational Lower Bound

VP Variance Preserving

# Chapter 1

## Introduction

The generative problem involves modelling the underlying distribution of some data to enable the generation of new samples, which resemble those of the original dataset. Diffusion models have become the state-of-the-art family for this task, surpassing other generative approaches, such as Generative adversarial networks (Goodfellow et al., 2020), Variational Autoencoders (Kingma and Welling, 2013) or Flow-based generative models (Rezende and Mohamed, 2015). These models have been particularly successful in image synthesis, as demonstrated by their utilisation in a number of commercial products, such as DALL-E (Ramesh et al., 2021) or Stable Diffusion (Rombach et al., 2022). However, their applications span a wide variety of domains, including image super-resolution (Saharia et al., 2022), speech synthesis (Chen et al., 2020) and video generation (Ho et al., 2022).

Moreover, their ability to generate samples from high-dimensional distributions makes them particularly well-suited for various applications in computational chemistry. For instance, diffusion models can be leveraged to generate novel foldable protein structures, thus potentially facilitating the process of designing proteins with specific functional properties (Watson et al., 2022; Wu et al., 2024; Yim et al., 2023). Additionally, they have been employed for generating chemically valid molecular structures Hoogetboom et al. (2022); Zhang et al. (2023b).

On the other hand, there are numerous problems in physical sciences for which no training data is available but, instead, the target density is known to be proportional to a Boltzmann-type distribution, i.e.  $p_{\text{target}} \sim \exp(-\xi(\cdot))$ , which can be evaluated (and possibly differentiated). In this setting, diffusion models are not readily applicable, although samples can be generated by resorting to Monte Carlo techniques, such as Annealed Importance Sampling (Neal, 2001) or Sequential Monte Carlo (Del Moral et al., 2006). Alternatively, variational approaches can be leveraged, thus approximating the target distribution using

a surrogate model, from which sampling can be done efficiently. However, sample quality depends on how well the surrogate model fits the target distribution.

However, in a data-constrained scenario, where an unnormalised density is also available, a sub-optimal diffusion model can be trained. Then, importance sampling can be used to remove any bias in subsequent computations by reweighing the samples (Midgley et al., 2022) by their importance sampling weights, although this method can be highly inefficient if the diffusion model is a poor approximation to the target.

To tackle this challenge, we propose two methods for improving sample quality of a diffusion model by leveraging knowledge of the unnormalised target distribution: the Controlled Ancestral sampler and the Controlled Annealed Langevin sampler. Experimental results are provided, demonstrating that the quality of the samples generated by the sub-optimal diffusion model can be improved.

## 1.1 Contributions

The contributions of this work are the following:

- Literature review of diffusion-based approaches to sampling.
- Two sampling approaches, the Controlled Ancestral and Annealed Langevin samplers, that enable to improve sampling for a pre-trained diffusion by optimisation of controlled stochastic differential equations.

## 1.2 Outline

This dissertation is organised as follows:

- Chapter 2 provides relevant background in diffusion models and the sampling problem, followed by a review of diffusion-inspired sampling approaches.
- Chapter 3 introduces two approaches to improve sample quality of diffusion models.
- Chapter 4 discusses limitations of the proposed approaches and outlines future work.

# Chapter 2

## Background

This chapter outlines the theoretical principles of generative modeling using diffusion models. Following, sampling problems are described, discussing the challenges faced when applying diffusion models in this context. At the end of the chapter, relevant work is briefly outlined.

### 2.1 Diffusion Probabilistic Models

The key idea in diffusion models, which is loosely-inspired in non-equilibrium statistical physics, consists in transforming the complex underlying distribution of the data into a prior distribution from which sampling is simple, e.g. isotropic Gaussian noise. This is usually referred to as forward (or noising) diffusion process, and it is done by progressively destroying the structure in the data by corrupting it with additive Gaussian noise across multiple steps.

Therefore, assume that  $p_{\text{target}}$  is the probability density of the data (which is generally unknown) and a set of samples  $X_0 \sim p_{\text{target}}$  is available. Diffusion Probabilistic Models (Ho et al., 2020) define a forward Markov chain by progressively adding Gaussian noise to the data, thus generating a sequence of noisy augmentations  $\{X_{\tau_i}\}_{i=1}^T$ , with  $0 = \tau_0 < \tau_1 < \dots < \tau_T = 1$ , defined as follows

$$X_{\tau_i} = \alpha_{\tau_i|\tau_{i-1}} X_{\tau_{i-1}} + \sigma_{\tau_i|\tau_{i-1}} \epsilon_{\tau_i} \text{ where } \epsilon_{\tau_i} \sim \mathcal{N}(0, I) \text{ and } i \in \{1, \dots, T\}. \quad (2.1)$$

Note that the forward diffusion is a first-order Gaussian auto-regressive process, so the marginal conditionals  $q(x_{\tau_i}|x_0)$  are also Gaussian. In fact, the noising parameters  $\{(\alpha_{\tau_i|\tau_{i-1}}, \sigma_{\tau_i|\tau_{i-1}})\}_{i=1}^T$  can be expressed in terms of the parameters  $\{(\alpha_t, \sigma_t)\}_{t \in [0,1]}$ , which

define the marginal conditionals (Kingma et al., 2021, Appendix A):

$$q(x_t|x_0) = \mathcal{N}(\alpha_t x_0, \sigma_t^2 I), \quad t \in [0, 1]. \quad (2.2)$$

A common choice is setting  $\alpha_t = \sqrt{1 - \sigma_t^2}$ , which corresponds to the variance-preserving diffusion process (Song and Ermon, 2019). For this setting, the forward diffusion is fully-specified by  $\sigma_t^2$  which is an increasingly monotonic function with range  $(0, 1)$  determining how fast the data is corrupted across noise levels. For illustration, Figure 2.1 shows the continuous approximation to the scheduler in Song and Ermon (2019), which is used in subsequent experiments.

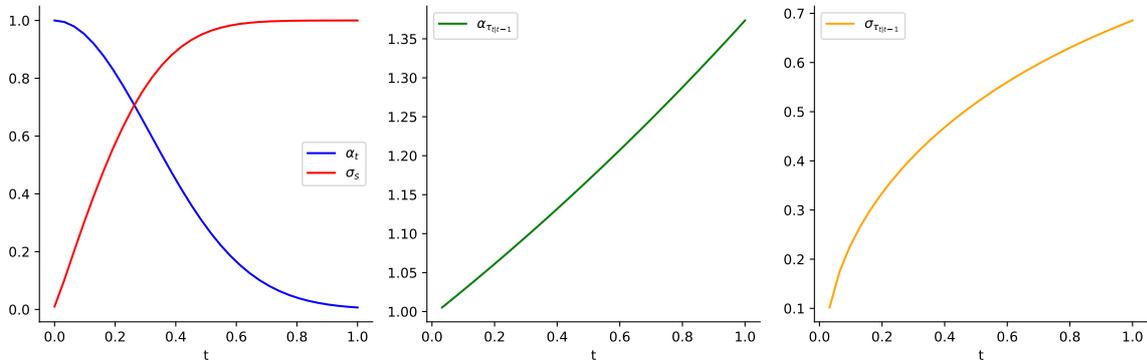


Fig. 2.1 Evolution of the noising parameters  $\sigma_t$  and  $\alpha_t = \sqrt{1 - \sigma_t^2}$  as a function of time for the continuous approximation of the scheduler in Song and Ermon (2019) (left). The schedule choice determines  $\alpha_{\tau_{i+1}|\tau_i}$  (middle) and  $\sigma_{\tau_{i+1}|\tau_i}$  (right), where  $\tau_i$  are equally spaced points in the interval  $[0, 1]$ , i.e.  $\tau_i = i/T$  ( $T = 32$ ,  $i \in \{0, \dots, T\}$ ).

Moreover, note that for the variance-preserving process, if  $\sigma_1^2 \approx 1$ , the marginal at the last step follows, approximately, an standard Gaussian distribution, i.e.  $q_1(x_1) \approx \mathcal{N}(x_1; 0, I)$ , which constitutes the *prior* distribution. Consequently, the forward diffusion process transports the target distribution into the prior distribution, by adding Gaussian noise at each step.

Once the forward diffusion is defined, the idea is to learn a reverse Markov chain which allows to navigate from the prior distribution,  $p_{\text{prior}} \equiv p_1$  to the target distribution  $p_{\text{target}} \equiv q_0(\cdot)$ . Typically, the reverse kernels are also Gaussian (a choice which is optimal when  $T \rightarrow \infty$ ), with their means, and possibly covariances, parameterised by a neural network which takes the current step as input (in the remainder of this section, we use  $\tau_i \equiv i$  to declutter the notation):

$$p_\theta(x_{i-1}|x_i) = \mathcal{N}(x_{i-1}, \mu_\theta(x_i, i), \sigma_\theta^2(x_i, i)). \quad (2.3)$$

Noting that  $q(x_{i-1}|x_i, x_0)$  is Gaussian, and its mean is a linear combination of  $x_0$  and  $x_i$ , i.e.

$$\mu_{i-1|0,i} = a_{i-1}x_0 + b_{i-1}x_i, \quad (2.4)$$

the learnable mean of the kernel  $p(x_{i-1}|x_i)$  can be parameterised, by analogy, as

$$\mu_\theta(x_i, i) = a_{i-1}\hat{x}_0^\theta(x_i, i) + b_{i-1}x_i, \quad (2.5)$$

so the network predicts instead the clean data from the noisy observation  $x_i$  (see Song and Ermon (2019) for the expressions of the coefficients  $a_i, b_i$ ). Alternatively, the network can be trained to extract the noise  $\varepsilon_i$  from the noisy observation, so the clean data point can be recovered as

$$\hat{x}_0^\theta(x_i, i) = \frac{x_i - \sigma_i \hat{\varepsilon}_i^\theta(x_i, i)}{\alpha_i}, \quad (2.6)$$

which is the parameterisation used in our experiments. Since the noise added at neighbouring steps is similar, this parameterisation builds an inductive bias which eases learning across noise levels. To train the model, one possibility would be to minimise the negative likelihood of the parameters, i.e.

$$\mathbb{E}_{q(x_0)} [-\log p_\theta(x_0)], \quad (2.7)$$

but the term  $p_\theta(x_0) = \int p_\theta(x_{0:T}) dx_{1:T}$  is intractable. Instead, Jensen's inequality can be applied to derive a variational lower bound

$$\mathbb{E}_{q(x_0)} [-\log p_\theta(x_0)] \leq \mathbb{E}_{q(x_{0:T})} \left[ -\log \left( \frac{p_\theta(x_{0:T})}{q(x_{1:T}|x_0)} \right) \right] := \mathcal{L}_{VLB}, \quad (2.8)$$

and this upper bound can be further simplified by factorising the backward process as

$$p_\theta(x_{0:T}) = p(x_T) \prod_{i=1}^T p_\theta(x_{i-1}|x_i), \quad (2.9)$$

while the forward process admits the expression

$$q(x_{1:T}|x_0) = q(x_T|x_0) \prod_{i=2}^T q(x_{i-1}|x_i, x_0), \quad (2.10)$$

so the variational bound can be simplified as follows

$$\mathcal{L}_{VLB} = \overbrace{\mathbb{E}_{q(x_1, x_0)} [-\log p_\theta(x_0|x_1)]}^{\mathcal{L}_0: \text{reconstruction loss}} + \overbrace{\mathbb{E}_{q(x_0)} [D_{KL}(q(x_T|x_0)||p(x_T))]}^{\mathcal{L}_T: \text{prior loss}} \quad (2.11)$$

$$+ \underbrace{\sum_{i=2}^T \mathbb{E}_{q(x_i, x_0)} [D_{KL}(q(x_{i-1}|x_i, x_0)||p_\theta(x_{i-1}|x_i))]}_{\mathcal{L}_{i-1}: \text{consistency loss}}. \quad (2.12)$$

Since the number of steps  $T$  is generally large (e.g. Song and Ermon (2019) set  $T = 1000$ ), computing the summation across all noise levels can be prohibitively expensive. Instead, it can be expressed as the expectation of an uniform random variable over the steps, and approximated, during training, using a Monte Carlo estimator. Therefore, the simplified training objective is

$$\mathcal{L}_{VLB} = C + \frac{T}{2} \cdot \mathbb{E}_{x_0 \sim q_0(\cdot), \varepsilon \sim \mathcal{N}(0, I), i \sim U\{1, \dots, T\}} \left[ w(i) \|\varepsilon_i - \hat{\varepsilon}_i^\theta(\alpha_i x_0 + \sigma_i \varepsilon, i)\|^2 \right], \quad (2.13)$$

where  $C$  contains the terms that do not depend on the parameters  $\theta$ , while  $w(i)$  depends on the schedule parameters and forward and backward variances (see Appendix A in Kingma et al. (2021) for concrete expressions). Therefore, the evaluation of this training objective does not require simulation of the entire trajectory of noisy augmentations  $\{x_i\}_{i=1}^T$ , thus enabling efficient training.

## 2.2 Diffusion with Stochastic Differential Equations

In Section 2.1, a discrete number of noised augmentations  $\{X_{\tau_i}\}_{i=1}^T$  were used. Alternatively, a continuum of noise scales can be considered by letting  $T \rightarrow \infty$ , thus modelling the forward diffusion as a stochastic process  $X = \{X_t\}_{t \in [0,1]}$  governed by the SDE

$$dX_t = f(X_t, t)dt + g(t)dW_t, \quad X_0 \sim p_{\text{target}}, \quad (2.14)$$

where  $W_t$  represents Brownian motion, and  $f, g$  are the *drift* and *diffusion* coefficients respectively (Song et al., 2020). The forward SDE for the parameterisation in Section 2.1 can be recovered for the choices

$$f(X_t, t) = \frac{1}{2} \frac{d \log \alpha_t^2}{dt} X_t, \quad g(t)^2 = \frac{d \sigma_t^2}{dt} - \frac{d \log \alpha_t^2}{dt} \sigma_t^2, \quad (2.15)$$

and in particular, for the variance-preserving SDE ( $\alpha_t^2 = 1 - \sigma_t^2$ ):

$$f(X_t, t) = -\frac{1}{2} \underbrace{\left( -\frac{d}{dt} \log(1 - \sigma_t^2) \right)}_{\beta(t)} X_t = -\frac{1}{2} \beta(t) Y_t, \quad g(t) = \sqrt{-\frac{d}{dt} \log(1 - \sigma_t^2)} = \sqrt{\beta(t)}. \quad (2.16)$$

Analogously to the discrete diffusion, the marginal  $p_{X_1}$  is required to approximately follow a tractable prior distribution  $p_{\text{prior}}$  from which sampling can be done efficiently. In this regard, the forward SDE defines a *transport* between the target distribution  $p_{\text{target}}$  and the prior  $p_{\text{prior}}$  (Villani et al., 2009). This is illustrated in Figure 2.2, which shows the evolution of the marginals for the stochastic processes governed by the forward diffusion SDE (left), and its time-reversal (right).

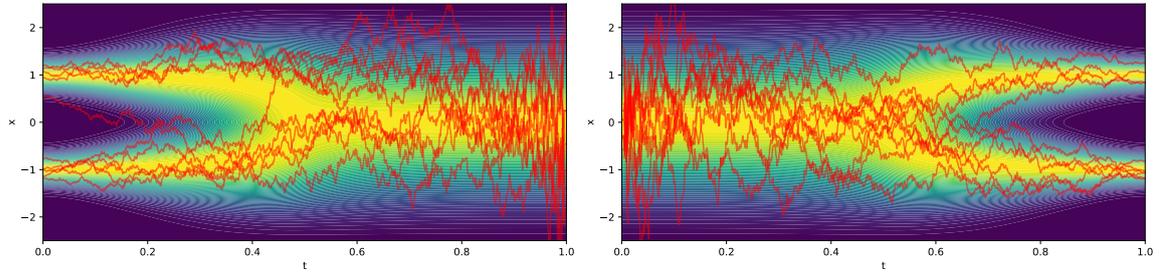


Fig. 2.2 Evolution of the marginals for a stochastic process governed by the forward diffusion SDE with initial boundary condition a 1-D GMM (left). Its time-reversal is also a diffusion process, whose initial condition matching the marginal at the last step of the forward diffusion (right). Trajectories obtained by simulating each SDE are depicted in red.

Therefore, sampling from the target distribution requires simulating the time-reversal of the forward diffusion, thus transporting the samples from  $p_{\text{prior}}$  to  $p_{\text{target}}$  (see Figure 2.2, right). Anderson (1982) proved that this time-reversal is also a diffusion process, governed by the backward SDE (we use the notation  $\overleftarrow{f}$  to indicate time-reversals for time-dependent functions, i.e.  $\overleftarrow{f}(t) = f(1-t)$ ):

$$dY_t = \left[ -\overleftarrow{f}(Y_t, t) + \overleftarrow{g}(t)^2 \nabla_x \log \overleftarrow{p}_{X_t}(Y_t) \right] dt + \overleftarrow{g}(t) dW_t, \quad Y_0 \sim p_{X_1}, \quad (2.17)$$

where  $\nabla_x \log p_{X_t}(Y_t)$  denotes the score of the marginal density at step  $t$ , which is generally not known. However, if samples from  $p_{\text{target}}$  are available, it can be estimated by training a score-based model  $s_\theta(x_t, t)$  via denoising score-matching (Vincent, 2011):

$$\mathcal{L}_{DSM} = \mathbb{E}_{X_0 \sim p_{\text{target}}, X_t \sim p_t(\cdot|X_0), t \sim U[0,1]} \left[ w(t) \|s_\theta(X_t, t) - \nabla_x \log p(X_t|X_0)\|^2 \right]. \quad (2.18)$$

For  $f(x, t)$  affine in the spatial coordinate, the transition kernel  $p(x_t|x_0)$  is Gaussian with (possibly) closed-form parameters, thus enabling the objective computation. In particular, for  $f(x, t) = -\frac{1}{2}\beta(t)x$  and  $g(t) = \sqrt{\beta(t)}$ , it can be expressed as

$$\nabla_x \log p(x_t|x_0) = -\frac{x_t - \alpha_t x_0}{\sigma_t^2} = -\frac{\varepsilon_t}{\sigma_t}, \text{ with } \varepsilon_t = \frac{x_t - \sigma_t x_0}{\sigma_t}. \quad (2.19)$$

By parameterising  $s_\theta(x, t) = -\hat{\varepsilon}_t^\theta(x, t)/\sigma_t$  (Song and Ermon, 2020), the DSM loss can be related to the discrete optimisation objective in Equation 2.13. Alternatively, the DSM objective can be derived from the forward KL divergence of the path measures induced by the backward process and the time-reversal of the forward diffusion, i.e.  $D_{\text{KL}}(\mathbb{P}_{\leftarrow X} || \mathbb{P}_Y)$  (Vargas et al., 2023). This perspective is specially relevant to theoretically ground novel sampling schemes, as shown in Section 2.4

To sum up, the SDE perspective provides a unifying framework for studying score-based generative models. For instance, methods such as SMLD (Song and Ermon, 2019) and DDPM (Ho et al., 2020) can be interpreted as Euler-Maruyama discretisations of a variance-exploding and variance-preserving SDEs respectively (Song et al., 2021, Section 2).

However, since the time-reversal cannot be exactly simulated, the sampling process is subject to numerical errors which, along with other factors, can negatively impact sample quality. Following, we analyse the different sources of error.

## 2.3 Sampling Error

Theoretically, diffusion models can generate unbiased samples from the target distribution, when trained to optimality with sufficient data (De Bortoli, 2022), and assuming that the time-reversal of the diffusion process is simulated exactly. However, in practice, numerical errors are inevitable, and sample quality can be affected by three factors (Chen et al., 2023):

1. The error due to initialising the backward SDE using a prior distribution  $p_{\text{prior}}$ , instead of the marginal at the last step of the forward process  $p_{X_1}$ . This irreducible error is given by  $D_{\text{KL}}(\mathbb{P}_{X_1} || \mathbb{P}_{Y_0})$ .
2. The numerical error arising from the discretisation of the backward SDE in a finite number of steps.
3. The discrepancy between the ground-truth score and its parameterised approximation, i.e.  $\mathbb{E}_{t \sim U(0,1), X_t \sim p_{X_t}(\cdot)} [ \|s_\theta(X_t, t) - \nabla_x \log p_{X_t}(X_t)\|^2 ]$ .

Regarding (1), it is possible to reduce this error by choosing a noise schedule so that the signal-to-noise ratio is approximately zero close to the end boundary condition of the forward

SDE (Kingma et al., 2021). Generally, this error is negligible since the forward process marginals converge exponentially fast to the prior Gaussian measure (Cao et al., 2024).

On the other hand, the SDE simulation error (2) can be systematically reduced by increasing the number of discretisation steps. Moreover, for a fixed number of steps, the discretisation error can be reduced by relying on the exponential integrator scheme to leverage the semi-linear structure of the backward SDE (Zhang and Chen, 2022), or resorting to higher-order method (Karras et al., 2022).

However, having fixed the discretisation scheme, as well as the number of steps, it is relevant to study how to choose the sampling parameters, such as the discretisation points, to improve sample quality.

For instance, the ancestral sampling method in DDPMs (Song and Ermon, 2019) corresponds to an Euler-Maruyama discretisation of the backward SDE for a sufficiently small step size (Song et al., 2021, Appendix E). However, using the same variance for the forward and backward diffusion is only optimal when the number of discretisation step goes to infinity (see Equation 2.17). Precisely, the optimal variance admits an analytical expression in terms of the score Bao et al. (2022), which can be approximated via a Monte Carlo estimator. As this estimator is biased if the score is not properly estimated, a clipping strategy can be introduced to avoid relying on inconsistent values.

On the other hand, the choice of discretisation points can considerably impact sample quality, specially if the score of the target distribution is not sufficiently smooth. In this setting, it is suggested to use exponentially decaying step sizes (Chen et al., 2023). Alternatively, if the discretisation points are fixed, the selection of an optimal sub-sampling trajectory, for a given number of sub-steps, can be cast into a least-cost-path problem, which can be efficiently solved using dynamic programming (Bao et al., 2022; Watson et al., 2021).

Finally, if the score is not properly estimated (3), the optimal sampling strategy (in the sense of KL divergence minimisation) might depend (at least partially) on the noise scales in which the perturbations concentrate (Cao et al., 2024). On the other hand, if an unnormalised target density is available, it might be possible to improve sample quality, even for a sub-optimal score model.

Therefore, in the next section, diffusion-based approaches to the problem of sampling from an arbitrary (unnormalised) density are discussed, thus setting the theoretical ground for the approaches presented in Chapter 4.

## 2.4 Sampling Problem

In some scenarios, data is scarce or even no data is available. As training competitive diffusion models requires large amounts of data, specially for high-dimensional data distributions, these models are not readily applicable.

However, it is possible that an unnormalised density  $\rho(x)$  for the target distribution is known. In this setting, the target density can be recovered as  $p_{\text{target}}(x) = \rho(x)/Z$ , where  $Z = \int \rho(x)dx$  is the normalisation constant (or partition function) of  $\rho$ . Generally, the integral  $\int \rho(x)dx$  is not tractable, although it can be approximated using Monte Carlo methods. Sampling from an unnormalised density is a recurrent problem in fields such as Bayesian inference (Neal, 1993) or statistical mechanics (Noé et al., 2019), and the main approaches to this problem are variational inference and Monte Carlo techniques.

Among the MC approaches, methods derived from Annealed Importance Sampling (Neal, 2001) and Sequential Monte Carlo (Del Moral et al., 2006) are regarded as the state-of-the-art in numerous sampling tasks. Although these methods can recover the target distribution in the asymptotic regime, they can suffer from slow mixing times and inability to capture mode diversity (Desjardins et al., 2010).

On the other hand, variational inference methods approximate the intractable target density by selecting, within a family of parameterised distribution (from which sampling can be done efficiently), the one which provides the best approximation to the target. Consequently, these methods cast the sampling task into an optimisation problem. A popular family of VI methods are normalising flows (Rezende and Mohamed, 2015), which define a sequence of diffeomorphisms mapping the unnormalised target distribution to a tractable distribution. In combination with AIS, normalising flows have demonstrated state-of-the-art performance in modelling the Boltzmann distribution of physical systems (Midgley et al., 2022). However, normalising flows require the transformation mappings to be bijective, as well as its Jacobian to be easily computable, thus limiting their expressivity.

Alternatively, the sampling problem can be framed as a transport of a tractable prior distribution to the target density via diffusion stochastic dynamics. Contrary to most MCMC approaches, diffusion-inspired methods can exhibit fast convergence to the target distribution, even in high-dimensions (Vargas et al., 2023), as the mixing properties of the reversed diffusion are inherited from those of the forward process (De Bortoli et al., 2021, Theorem 1).

Additionally, this perspective has allowed to draw formal connections linking diffusion models with Schrödinger bridges (Chen et al., 2021), stochastic optimal control (Vargas et al., 2023; Zhang et al., 2023a; Zhang and Chen, 2021), annealed importance sampling (Doucet

et al., 2022) and optimal transport (Vargas et al., 2024). Some of these approaches are further discussed in the following section.

## 2.5 Related Works

In the following section, relevant diffusion-based approaches to the sampling problem are briefly introduced. To this end, we rely on the notation used in Berner et al. (2022), to present them under a common framework. This section relies heavily on the concept of path measure, which corresponds to a probability measure on the functional space  $\mathcal{C}([0, 1], \mathbb{R}^d)$ . As each solution to an SDE induces a path measure, this concept is fundamental to characterise “how far” two SDEs are from being time-reversals to each other. For a formal introduction to path measures, we refer the reader to Léonard (2013).

The approaches in this section are defined in terms of two stochastic processes. One is induced by the controlled *inference* SDE

$$dY_t^v = (f + g \cdot v)(Y_t, t)dt + g(t)dW_t, Y_0 \sim p_{\text{target}}, \quad (2.20)$$

and the other by the controlled *generative* SDE

$$dX_t^u = (-\overleftarrow{f} + \overleftarrow{g} \cdot u)(X_t, t)dt + \overleftarrow{g}(t)dW_t, X_0 \sim p_{\text{prior}}. \quad (2.21)$$

Note that inference SDE can be regarded, for  $v = 0$ , as the forward diffusion, defined in Section 2.2, while the uncontrolled generative SDE, i.e.  $v = 0$ , corresponds to the backward diffusion SDE (the roles of  $X$  and  $Y$  are interchanged with respect to the notation in Section 2.2 for consistency with the notation in Berner et al. (2022)).

In this setting, the goal is to identify suitable controls  $u, v \in \mathcal{U}$  so that the stochastic processes are reversals to each other, thus verifying, in particular, that  $X_1^u \sim p_{\text{target}}$  and  $Y_1^v \sim p_{\text{prior}}$ . Formally, this problem can be formulated as follows:

**Problem (Richter and Berner, 2023):** Let  $\mathbb{P}_{X^u}$  and  $\mathbb{P}_{\overleftarrow{Y}^v}$  be the path space measures of the processes  $X^u$  and  $\overleftarrow{Y}^v$  (where  $\overleftarrow{Y}^v$  denotes the stochastic process induced by the time-reversal of  $Y^v$ ). Let  $D : \mathcal{P} \times \mathcal{P} \rightarrow \mathbb{R}_{\geq 0}$  be a divergence between path measures, thus satisfying  $D(\mathbb{P}, \mathbb{Q}) = 0$  if and only if  $\mathbb{P} = \mathbb{Q}$ . The goal is to obtain optimal controls  $u^*, v^*$  such that

$$u^*, v^* \in \arg \min_{u, v \in \mathcal{U} \times \mathcal{U}} D\left(\mathbb{P}_{X^u} \parallel \mathbb{P}_{\overleftarrow{Y}^v}\right). \quad (2.22)$$

The most common choice for  $D$  across the diffusion-based sampling literature is the Kullback-Leibler divergence

$$D_{KL}(\mathbb{P}_{X^u} \parallel \mathbb{P}_{\tilde{Y}^v}) = \mathbb{E}_{\mathbb{P}_{X^u}} \left[ \log \frac{\mathbb{P}_{X^u}}{\mathbb{P}_{\tilde{Y}^v}} \right], \quad (2.23)$$

although this objective exhibits mode seeking behaviour (Midgley et al., 2022), as well as non-zero variance Monte Carlo gradient estimators, even for optimal controls, thus potentially hindering convergence to the optimum (Roeder et al., 2017).

Alternatively, the log-variance divergence can be used

$$D_{LV}^{\mathbb{P}_Z}(\mathbb{P}_{X^u} \parallel \mathbb{P}_{\tilde{Y}^v}) = \mathbb{V}_{\mathbb{P}_Z} \left[ \log \frac{\mathbb{P}_{X^u}}{\mathbb{P}_{\tilde{Y}^v}} \right], \quad (2.24)$$

where  $\mathbb{P}_Z$  is the law of a reference stochastic process, thus enabling off-policy training of the controls. Consequently, it can be beneficial to rely on an exploratory reference process  $Z$  in the early-stages of training (thus potentially preventing mode collapse), while exploiting the current control, by transitioning to  $Z = X^u$  later in the optimisation (Richter and Berner, 2023).

The log-variance divergence exhibits the *sticking-the-landing* property (Roeder et al., 2017), i.e. the variance of its Monte Carlo gradient estimator vanishes at the optimal solution, as opposed to the KL divergence loss, thus potentially stabilising the training procedure close to the optimum. Moreover, for the choice  $Z = X^u$ , its gradient estimator can be regarded as a control-variate version of the gradient of the KL loss, thus it can be expected to exhibit lower variance Monte Carlo estimation (Richter and Berner, 2023).

Both divergences require computing the Radon-Nikodym derivative between the path measures  $\mathbb{P}_{X^u}$  and  $\mathbb{P}_{\tilde{Y}^v}$ , a technical derivation relying in Girsanov's theorem (Üstünel and Zakai, 2013). For the sake of conciseness, we refer the reader to Appendix A.2. (Richter and Berner, 2023).

However, Problem 2.5 has infinitely many solutions. In particular, the presented objectives only enforce Nelson's identity (Nelson, 2020):

$$u^* + v^* = g^T \nabla \log \overleftarrow{p}_{Y^{v^*}}. \quad (2.25)$$

Consequently, the diffusion-based approaches, presented in the following section, use specific choices for the controls  $u$  and  $v$  to render the solution of the time-reversal problem unique.

### 2.5.1 Time-Reversed Diffusion Sampler (DIS)

Berner et al. (2022) characterise the marginal log-densities of an stochastic process, governed by a SDE, as the solution to an Hamilton-Jacobi-Bellman equation, thus allowing to connect score-based generative modelling with stochastic optimal control. Moreover, they rely on this connection to derive their *time-reversed diffusion sampler* (DIS).

This sampler can be derived from the framework presented in Section 2.5. In particular, to solve the non-uniqueness of the controlled sampling problem, the authors set  $v = 0$ , thus relying on an uncontrolled inference SDE. This choice constrains the drift  $f$  and diffusion  $g$  coefficients to be chosen so that  $Y_1^0$  is distributed approximately as  $p_{\text{prior}}$ . To this end, they use the variance-preserving (VP) SDE (Song and Ermon, 2019), and fix  $p_{\text{prior}} = \mathcal{N}(0, I)$ .

Therefore, only the control  $u$  of the generative SDE is optimised, which is parameterised as

$$u_{\theta}(x, t) = \Phi_{\theta}^{(1)}(x, t) + \Phi_{\theta}^{(2)}(x, t)g(t)s(x, t), \quad (2.26)$$

with

$$s(x, t) := t\nabla\log p_{\text{prior}}(x) + (1-t)\nabla\log p_{\text{target}}(x) \quad (2.27)$$

thus linearly interpolating between the score of the prior and target distributions. Since the optimal control is  $u^*(x, t) = g(t)\nabla_x\log \overleftarrow{p}_{Y^0}$  (from Nelson’s identity), the authors initialise  $\Phi_{\theta}^{(1)}$  and  $\Phi_{\theta}^{(2)}$  to 0 and 1 respectively, thus ensuring that, at initialisation, the parameterised control is optimal at the initial and terminal boundary conditions.

To optimise  $u$ , they minimise the control loss  $\mathcal{L}_{DIS}(u)$ , which is equivalent to indirectly minimising the reverse KL divergence  $D_{KL}(\mathbb{P}_{X^u}, \mathbb{P}_{\overleftarrow{Y}^0})$  since

$$\mathcal{L}_{DIS}(u) = D_{KL}(\mathbb{P}_{X^u} || \mathbb{P}_{\overleftarrow{Y}^0}) - \log Z \quad (2.28)$$

where  $\mathcal{L}_{DIS}(u)$  involves an unnormalised density  $\rho$  of the target distribution, and  $Z$  denotes its corresponding normalisation constant, i.e.  $p_{\text{target}} = \rho/Z$ . This loss can be derived from Proposition 2.3 (Richter and Berner, 2023) for the specific choice  $v = 0$ , and it is approximated via Monte Carlo, by simulating trajectories of the stochastic process  $X^u$  using the Euler-Maruyama scheme.

Regarding the numerical results, the authors showed that DIS consistently outperformed the Path Integral Sampler (Zhang and Chen, 2021) in the estimation of normalising constants, expectations and standard deviations for three sampling benchmarks.

## 2.5.2 Controlled Monte Carlo Diffusion Sampler (CMCD)

In DIS, the forward diffusion is uncontrolled, i.e.  $v = 0$ , while the control  $u$  of the backward process is optimised to minimise the KL divergence  $D_{KL}(\mathbb{P}_{X^u} || \mathbb{P}_{\overleftarrow{Y}_0})$ .

Vargas et al. (2024) argue that simultaneously adapting the forward and backward controls can lead to improved sampling results. However, if the controls  $u$  and  $v$  are independent, the optimal solution to the time-reversal problem is not unique, as shown in Section 2.5. To this end, they tie the controls  $u$  and  $v$  to ensure uniqueness.

Concretely, they set the drift and diffusion coefficients of the forward process to

$$f(Y_t, t) = \sigma^2 \nabla \log \pi(Y_t, t), \quad g(t) = \sigma \sqrt{2}. \quad (2.29)$$

Similarly to DIS,  $\nabla \log \pi(Y_t, t)$  is defined as a linear interpolation between scores of the prior and target

$$\nabla \log \pi(x, t) := t \nabla \log p_{\text{prior}}(x) + (1 - t) \nabla \log p_{\text{target}}(x), \quad (2.30)$$

thus ensuring optimality at the boundary conditions. Additionally, they parameterise the forward control with a neural network, i.e.  $g \cdot v = -\Phi_\theta$ . To motivate their choice for the control  $u$ , they observe that the optimal controls are related to each other by Nelson's identity:

$$g \cdot u^* + g \cdot v^* = g^2 \cdot \nabla \log \overleftarrow{p}_{Y^{v^*}}. \quad (2.31)$$

Therefore, considering that  $\nabla \log \pi(Y_t, t)$  can be regarded as an approximation to  $\nabla \log \overleftarrow{p}_{Y^{v^*}}$ , the authors set the backward control to

$$g \cdot u = g^2 \cdot \overleftarrow{\nabla \log \pi} - g \cdot v = 2\sigma^2 \overleftarrow{\nabla \log \pi} + \Phi_\theta, \quad (2.32)$$

so the controlled generative SDE is given by

$$dX_t^u = (\sigma^2 \overleftarrow{\nabla \log \pi} + \overleftarrow{\Phi}_\theta)(X_t, t) dt + \sigma \sqrt{2} dW_t, \quad X_0 \sim p_{\text{prior}} \quad (2.33)$$

The tied controls  $u$  and  $v$  are jointly optimised by indirectly minimising the reverse KL divergence  $D_{KL}(\mathbb{P}_{X^u} || \mathbb{P}_{Y^v})$ , using an objective analogous to that of DIS (Richter and Berner, 2023, Proposition 2.1), which only involves an unnormalised density for the target distribution. Moreover, the Euler-Maruyama scheme is used to simulate trajectories from the generative SDE.

## Chapter 3

# Controlled Sampling from Diffusion Models

In the previous chapter, the theoretical foundations of diffusion models for generative modelling were introduced, and their sources of errors characterised into three different components. Moreover, it was highlighted that these models excel when training data is abundant. If, instead, data is scarce, but the target (unnormalised) density is known, variational inference and Monte Carlo methods are the standard approaches to sample from the target distribution. In this context, two approaches to diffusion-based sampling were introduced: the Time-Reversed Diffusion Sampler (DIS) and the Controlled Monte Carlo Diffusion Sampler (CMCD). These were motivated from the perspective of identifying optimal controls for two stochastic processes to be time-reversals of each other. Both of these methods rely on interpolating the scores of the prior and target distribution, either by using a learnable interpolation for the backward control (DIS) or by using a fixed linear interpolation of the scores to define the drift coefficient of the inference SDE (CMCD).

In this regard, approaches to generative modelling and sampling problems generally assume two mutually exclusive scenarios: either data is available, but the target density is not known; or an unnormalised density is available, and there is no access to data from the target distribution.

However, it is interesting to study a middle-point scenario, i.e. some data is available, thus allowing to train a (probably) sub-optimal. Moreover, if an (unnormalised) density is known, it can be possible to improve the quality of the samples generated by the generative model, for instance by re-weighting the samples by their importance sampling coefficients (Midgley et al., 2022). In turn, the refined samples might be bootstrapped to further train the diffusion model.

To this end, we study how to improve sample quality for a pre-trained diffusion by optimising relevant controls in the inference and generative SDEs.

### 3.1 Motivation

In the following sections, it is assumed that a score-model model  $s_\theta(x, t)$ , trained via score-matching to generate samples from a target distribution  $p_{\text{target}}$ , is available (see Section 2.2). Additionally, an unnormalised density  $\rho$  for  $p_{\text{target}}$  is known, i.e.  $p_{\text{target}} = \rho/Z$ , but the integral  $\int \rho dx$  is not tractable, so the normalisation constant  $Z$  cannot be analytically computed.

Following the notation in Section 2.5, the forward diffusion in DDPMs (Song and Ermon, 2019) can be regarded as the uncontrolled *inference* SDE

$$dY_t^0 = f(Y_t, t)dt + g(t)dW_t, Y_0 \sim p_{\text{target}}, \quad (3.1)$$

where the roles of the stochastic processes  $X$  and  $Y$  are interchanged with respect to the notation used in Section 2.2. Therefore, the variance-preserving SDE can be recovered by setting the diffusion and drift coefficients as

$$f(Y_t, t) = -\frac{1}{2}\beta(t)Y_t \text{ and } g(t) = \sqrt{\beta(t)}, \text{ where } \beta(t) = -\frac{d}{dt}\log(1 - \sigma_t^2). \quad (3.2)$$

On the other hand, for the *generative* SDE

$$dX_t^u = (-\overleftarrow{f} + \overleftarrow{g} \cdot u)(X_t, t)dt + \overleftarrow{g}(t)dW_t, X_0 \sim p_{\text{prior}}, \quad (3.3)$$

the control can be parameterised as  $u_\theta(X_t, t) = g(t)s_\theta(X_t, t)$ . Therefore, if  $p_{Y_1} \approx p_{\text{prior}}$  and  $s_\theta(X_t, t) \approx \nabla_x \log p_{Y_t}(X_t)$ , the generative SDE (or backward diffusion) is the time-reversal of the inference SDE (Anderson, 1982). If this was the case, the reverse KL divergence for the path measures, induced by these stochastic processes, should be close to zero, i.e.  $D_{KL}(\mathbb{P}_{X^u} || \mathbb{P}_{\overleftarrow{Y}_0}) \approx 0$ .

However, if the score model is not accurate, these path measures can greatly differ, so the simulation of the generative SDE can be expected to yield low-quality samples (see Section 2.3). This can be problematic, for instance, if these samples are used in the Monte Carlo estimation of some statistic of the target distribution, as these estimates would be biased. More generally, for a functional in path space  $\phi : C([0, 1], \mathbb{R}^d) \rightarrow \mathbb{R}^n$ , we would need to

approximate

$$\mathbb{E}_{\mathbb{P}_{\overleftarrow{Y}^0}}[\phi(X)] \approx \mathbb{E}_{\mathbb{P}_{X^u}}[\phi(X)]. \quad (3.4)$$

However, it is possible to correct the bias by using importance sampling in path space, thus re-weighting the trajectories by the Radon-Nikodym derivative for the path measures of the time-reversal  $\mathbb{P}_{\overleftarrow{Y}^0}$  and the sub-optimal controlled process  $\mathbb{P}_{X^u}$ :

$$\mathbb{E}_{\mathbb{P}_{\overleftarrow{Y}^0}}[\phi(X)] = \mathbb{E}_{\mathbb{P}_{X^u}}[\phi(X) \underbrace{\frac{d\mathbb{P}_{\overleftarrow{Y}^0}}{d\mathbb{P}_{X^u}}(X)}_{w^u(X)}]. \quad (3.5)$$

However, the variance of this estimator can scale exponentially in the KL divergence  $D_{KL}(\mathbb{P}_{X^u} || \mathbb{P}_{\overleftarrow{Y}^0})$ , thus being very sensitive to how far the control  $u$  is from optimality (Hartmann and Richter, 2024). Therefore, to reduce the variance of the importance sampling estimates, it is paramount to reduce the divergence between these path measures.

To this end, note that the choices for the stochastic processes in the present section not only match the SDE formulation of generative modelling (Section 2.2), but also closely resemble those of the Time-Reversed Diffusion Sampler (DIS): the inference SDE is uncontrolled but, instead of parameterising the control  $u$  as

$$u_\theta(x, t) = \Phi_\theta^{(1)}(x, t) + \Phi_\theta^{(2)}(x, t)g(t)s(x, t), \quad (3.6)$$

the pre-trained score model is leveraged

$$u_\theta(x, t) = g(t)s_\theta(x, t). \quad (3.7)$$

Therefore, one possibility would be to further improve the control  $u$  by optimising the DIS objective, i.e.  $D_{KL}(\mathbb{P}_{X^u} || \mathbb{P}_{\overleftarrow{Y}^0})$ , with respect to the score model parameters  $\theta$ . However, this might not be desirable for two reasons:

- The computation of  $D_{KL}(\mathbb{P}_{X^u} || \mathbb{P}_{\overleftarrow{Y}^0})$  involves unrolling the trajectory of  $X^u$ , thus requiring to compute gradients across  $N$  model evaluations (where  $N$  denotes the number of discretisation steps). Since modelling the score across noise levels  $\nabla_x \log p_{Y_t^0}(x, t)$  generally requires a model with high capacity, training can become prohibitively expensive, both in terms of memory and computational cost. However, the *stochastic adjoint sensitivity method* (Kidger et al., 2021) could be used to alleviate this issue.
- Since the score model parameters  $\theta$  were trained using the DSM objective, re-training them for the reverse divergence loss might result in a performance drop for the original

task, a phenomenon commonly observed in continuous learning (Hess et al., 2023). This is undesirable, specially if the score model is intended to be further optimised as new samples from the target distribution are available.

These observations suggest that the parameters  $\theta$  should remain fixed, thus enabling to detach them from the gradient computations. However, the control  $u$  would no longer be learnable, thus rendering the proposed DIS variation invalid.

## 3.2 Controlled Ancestral Sampler

Following the discussion on the previous section, an alternative learnable control needs to be incorporated into the generative SDE to enable the optimisation of the reverse divergence, while fixing the parameters of the score model  $s_\theta(x, t)$ . Moreover, given the promising results of the CMCD sampler (Vargas et al., 2024), we consider controlling, not only the generative SDE, but also the inference one.

To this end, we parameterise the inference control  $v$  as  $g \cdot v = -\Phi_\omega$ . To guarantee uniqueness of the time-reversal problem, we leverage Nelson’s identity, in a similar fashion as Vargas et al. (2024), but replacing the linear interpolation of the score  $\nabla_x \log \pi$  by the pre-trained score model. Therefore, we set

$$g \cdot u = g^2 \cdot s_\theta - g \cdot v = g^2 \cdot s_\theta + \Phi_\omega. \quad (3.8)$$

Consequently, the generative and inference SDEs are defined as follows

$$dY_t^v = [f(Y_t, t) - \Phi_\omega(Y_t, t)]dt + g(t)dW_t, Y_0 \sim p_{\text{target}} \quad (3.9)$$

$$dX_t^u = [-\overleftarrow{f}(X_t, t) + \overleftarrow{g}(t)^2 \overleftarrow{s}_\theta(X_t, t) + \overleftarrow{\Phi}_\omega(Y_t, t)]dt + \overleftarrow{g}(t)dW_t, X_0 \sim p_{\text{prior}}. \quad (3.10)$$

Note that, for an optimal score model, i.e.  $s_\theta(X_t, t) = \nabla_x \log p_{Y_t^0}(X_t)$ , and assuming that  $Y_1^0 \sim p_{\text{prior}}$ , the optimal control is  $v = -\Phi_\omega = 0$  (Anderson, 1982). However, it is highly unlikely that the score model  $s_\theta$  is perfectly accurate. For that scenario, the network  $\Phi_\omega$  can be optimised to ensure that the generative process matches the time-reversal of the inference SDE.

Therefore, for a given divergence between path measures  $D$ , the parameters  $\omega$  of the control network can be optimised to minimise  $D(\mathbb{P}_{X^u} || \mathbb{P}_{\overleftarrow{Y}^v})$ , similarly to DIS and CMCD (see Section 2.5).

To simulate the generative SDE, we can consider a sub-trajectory of time-steps  $0 = \tau_0 < \dots < \tau_T = 1$  and apply the Euler-Maruyama discretisation scheme:

$$X_{\tau_{i+1}} = X_{\tau_i} + (-\overleftarrow{f} + \overleftarrow{g} \cdot \overleftarrow{\delta}_\theta + \overleftarrow{\Phi}_\omega)(X_{\tau_i}, \tau_i) \Delta_{\tau_{i+1}} + \overleftarrow{g}(t) \underbrace{(W_{\tau_{i+1}} - W_{\tau_i})}_{\sim \mathcal{N}(0, \Delta_{\tau_{i+1}} I)}, \quad (3.11)$$

where  $\Delta_{\tau_{i+1}} = \tau_{i+1} - \tau_i$ . Moreover, if we set the drift  $f$  and diffusion  $g$  coefficients so that the uncontrolled inference SDE matches the variance-preserving SDE (Song and Ermon, 2019), i.e.

$$f(x, t) = -\frac{1}{2}\beta(t), g(t) = \sqrt{\beta(t)}, \text{ where } \beta(t) = -\frac{d}{dt} \log(1 - \sigma_t^2). \quad (3.12)$$

Assuming that  $\Delta_{\tau_i}$  is sufficiently close to 0 for every  $i \in \{1, \dots, T\}$ , the Euler-Maruyama discretisation matches the ancestral sampler in DDPMs (Song et al., 2021, Appendix E), with the addition of the control  $\Phi_\omega$ :

$$p(X_{\tau_{i+1}} | X_{\tau_i}) = \mathcal{N} \left( X_{\tau_{i+1}}; \frac{1}{\overleftarrow{\alpha}_{\tau_{i+1} | \tau_i}} X_{\tau_i} + \frac{\overleftarrow{\sigma}_{\tau_{i+1} | \tau_i}^2}{\overleftarrow{\alpha}_{\tau_{i+1} | \tau_i}} \overleftarrow{\delta}_\theta(X_{\tau_i}, \tau_i) + \Delta_{\tau_{i+1}} \overleftarrow{\Phi}_\omega(X_{\tau_i}, \tau_i), \overleftarrow{\sigma}_{\tau_{i+1} | \tau_i}^2 I \right) \quad (3.13)$$

where  $\overleftarrow{\alpha}_{\tau_{i+1} | \tau_i} \equiv \alpha_{1-\tau_i | 1-\tau_{i+1}}$  and similarly for the rest of noising parameters (see Section 2.1). Similarly, the EM discretisation of the generative SDE, for a sufficiently small step size (so that  $\alpha_{t|s} \approx -\frac{1}{2}\beta(s)$  and  $\sigma_{t|s} \approx \sqrt{\beta(s)}$ ), can be expressed as

$$q(X_{\tau_i} | X_{\tau_{i+1}}) = \mathcal{N}(\overleftarrow{\alpha}_{\tau_{i+1} | \tau_i} X_{\tau_{i+1}} - \Delta_{\tau_{i+1}} \overleftarrow{\Phi}_\omega(X_{\tau_{i+1}}, \tau_{i+1}); \overleftarrow{\sigma}_{\tau_{i+1} | \tau_i}^2 I). \quad (3.14)$$

Moreover, for the Euler-Maruyama discretisation, Proposition E.1. in Vargas et al. (2024) can be applied to obtain a time-discretisation of the Radon-Nikodym derivative  $\frac{d\mathbb{P}_{X^u}}{d\mathbb{P}_{Y^v}}$ . By analogy, relying on the ancestral sampling integrator, the discretised objective for  $D = D_{KL}$  is

$$D_{KL}(\mathbb{P}_{X^u} || \mathbb{P}_{Y^v}) \approx \mathbb{E}_{X^u} \left[ \log \left( Z \cdot \frac{p_{\text{prior}}(X_{\tau_0}) \prod_{i=0}^{T-1} p(X_{\tau_{i+1}} | X_{\tau_i})}{\rho(X_{\tau_T}) \prod_{i=0}^{T-1} q(X_{\tau_i} | X_{\tau_{i+1}})} \right) \right], \text{ where } \rho = Z p_{\text{target}}. \quad (3.15)$$

Therefore, the discrete objective corresponds to the ratio of transitions densities for two discrete Markov chains, which can be interpreted as controlled versions of the forward and backward discrete diffusions in DDPMs (see Section 2.1).

Due to this analogy, it is sensible to consider relying on a learnable variance for the discretisation of the generative SDE. To this end, the following result characterises the optimal mean and variance of the backward transition kernels for the uncontrolled setting:

**Lemma** (Bao et al., 2022, Appendix E.1): For the discrete diffusion, presented in Section 2.1, the optimal means and variances (in the sense of KL divergence minimisation) for the backward transition kernels  $p(X_s|X_t) = \mathcal{N}(X_s; \mu_{s|t}(X_t), \sigma_{s|t}^2 I)$  are:

$$\mu_{s|t}^*(X_t) = \frac{1}{\alpha_{t|s}} \left( X_t + \sigma_{t|s}^2 \nabla_x \log p_{Y_t^0}(X_t) \right) \quad (3.16)$$

$$\sigma_{s|t}^{2*} = \frac{\sigma_{t|s}^2}{\alpha_{t|s}} \left( 1 - \sigma_{t|s}^2 \mathbb{E}_{p_{Y_t^0}} \frac{\|\nabla_x \log p_{Y_t^0}(X_t)\|^2}{d} \right). \quad (3.17)$$

Note that the optimal mean matches that mean of the ancestral sampling backward kernel when  $s_\theta$  is an accurate model of the score. Moreover, this lemma states that the optimal variance only depends on the noise level in which the score is evaluated, and not on the iterate  $X_t$ . Therefore, one possibility would be to parameterise the variance using a neural network taking as input the noise level  $t$ , thus modelling the squared norm of the score. However, this result only holds when considering isotropic Gaussian backward kernels.

Alternatively, to provide greater flexibility, a diagonal covariance  $\Sigma$  can be considered, allowing to interpolate, in a per-dimension fashion, between the variances of  $q(X_t|X_s)$  and  $q(X_s|X_t, X_0)$  (Nichol and Dhariwal, 2021):

$$\Sigma(X_t, t) = \exp \left( v(X_t, t) \underbrace{\sigma_{t|s}^2}_{q(X_t|X_s)} + (1 - v(X_t, t)) \underbrace{\sigma_{t|s}^2 \frac{\sigma_s^2}{\sigma_t^2}}_{q(X_s|X_t, X_0)} \right), \quad (3.18)$$

which are optimal in the limiting scenarios where  $T \rightarrow \infty$  or the data comes from a Dirac distribution respectively (Turner et al., 2024). Note that the iterate  $X_t$  is used as input, since the optimal variance characterisation does not hold in the non-isotropic case, and can be expected to depend on this value.

Finally, in Section 2.3, the importance of the selection of the discretisation points was highlighted. In particular, as the regularity of the score worsens for low noise scales, an exponentially decaying step size can be helpful to decrease the discretisation error (Chen et al., 2023). However, instead of relying in a hand-crafted strategy to fix the discretisation points, we opt for optimising them along with the control parameters  $\omega$ .

In particular, we consider the learnable free-parameters  $\beta = (\beta_1, \dots, \beta_T) \in \mathbb{R}^T$ , and compute the discretisation trajectory as

$$\tau_0 = 0, \tau_i = \sum_{j=1}^i \sigma_j(\beta), \text{ where } \sigma_j(\beta) = \frac{e^{\beta_j}}{\sum_{k=1}^T e^{\beta_k}} \text{ and } i \in \{1, \dots, T\}. \quad (3.19)$$

Therefore, a uniform initialisation of the discretisation points can be obtained by setting  $\beta_i = 0$ . The parameterisation of  $\beta$  in log-space (due to the application of the softmax) is essential for numerical stability, as modeling exponentially decaying step sizes in natural scale can potentially lead to truncation errors, specially when relying on 32-bit floating point numbers. Precisely, note that exponentially decaying step sizes can be recovered by setting  $\beta_i = -i \log r$ .

### 3.3 Controlled Annealed Langevin Sampler

For the Controlled Ancestral sampler, the inference SDE was defined as a controlled version of the variance-preserving SDE in DDPMs. Alternatively, we can consider a controlled annealed-based inference SDE, inspired in the CMCD sampler and the corrector-only samplers used in score-based generative modelling (Song et al., 2021, Section E):

$$dY_t^v = \underbrace{\sigma^2 s_\theta(Y_t, t)}_{f(Y_t, t)} + \underbrace{(-\Phi_\omega(Y_t, t))}_{(g \cdot v)(Y_t, t)} dt + \underbrace{\sigma \sqrt{2}}_{g(t)} dW_t, Y_0 \sim p_{\text{target}}. \quad (3.20)$$

From Nelson's identity, we parameterise  $g \cdot u$  in terms of  $v$  to ensure uniqueness for the time-reversal problem

$$g \cdot u = g^2 s_\theta + \Phi_\omega = 2\sigma^2 s_\theta + \Phi_\omega, \quad (3.21)$$

so the generative SDE is given by

$$dX_t^u = \left[ -\underbrace{\sigma^2 \overleftarrow{s}_\theta(X_t, t)}_{\overleftarrow{f}(X_t, t)} + \underbrace{2\sigma^2 \overleftarrow{s}_\theta(X_t, t) + \overleftarrow{\Phi}_\omega(X_t, t)}_{\overleftarrow{(g \cdot u)}(X_t, t)} \right] dt + \underbrace{\sigma \sqrt{2}}_{\overleftarrow{g}(t)} dW_t, X_0 \sim p_{\text{prior}}, \quad (3.22)$$

which simplifies to

$$dX_t^u = [\sigma^2 \overleftarrow{s}_\theta(X_t, t) + \overleftarrow{\Phi}_\omega(X_t, t)] dt + \sigma \sqrt{2} dW_t, X_0 \sim p_{\text{prior}}, \quad (3.23)$$

Similarly to the ancestral sampler, we rely on the Euler-Maruyama scheme to simulate the generative SDE. Therefore, for the discretisation steps  $0 = \tau_0 < \dots < \tau_T = 1$ , we have

$$X_{\tau_{i+1}} = X_{\tau_i} + (\sigma^2 \overleftarrow{s}_\theta + \overleftarrow{\Phi}_\omega)(X_{\tau_i}, \tau_i) \Delta_{\tau_{i+1}} + \sigma \sqrt{2} \underbrace{(W_{\tau_{i+1}} - W_{\tau_i})}_{\sim \mathcal{N}(0, \Delta_{\tau_{i+1}} I)}. \quad (3.24)$$

For this scheme, the time-discretisation of the path measure divergence  $D(\mathbb{P}_{X^u} \parallel \mathbb{P}_{\overleftarrow{Y}^v})$  can be obtained by applying Proposition E.1. in Vargas et al. (2024) for  $D = D_{KL}$ :

$$D_{KL}(\mathbb{P}_{X^u} \parallel \mathbb{P}_{\overleftarrow{Y}^v}) \approx \mathbb{E}_{X^u} \left[ \log \left( Z \cdot \frac{p_{\text{prior}}(X_{\tau_0}) \prod_{i=0}^{T-1} \mathcal{N}(X_{\tau_{i+1}}; X_{\tau_i} + (\sigma^2 \overleftarrow{\delta}_{\theta} + \overleftarrow{\Phi}_{\omega})(X_{\tau_i}, \tau_i) \Delta_{\tau_{i+1}}, 2\sigma^2 \Delta_{\tau_{i+1}})}{\rho(X_{\tau_T}) \prod_{i=0}^{T-1} \mathcal{N}(X_{\tau_i}; X_{\tau_{i+1}} + (\sigma^2 \overleftarrow{\delta}_{\theta} - \overleftarrow{\Phi}_{\omega})(X_{\tau_{i+1}}, \tau_{i+1}) \Delta_{\tau_{i+1}}, 2\sigma^2 \Delta_{\tau_{i+1}})} \right) \right]. \quad (3.25)$$

## 3.4 Performance Metrics

Following, we outline the performance metrics that are used to evaluate the proposed methods.

### 3.4.1 Estimation of Normalisation Constants

In the following experiments, the target distribution  $p_{\text{target}}$  is only known up to a multiplicative constant, i.e.  $p_{\text{target}} = \rho/Z$ , where  $Z$  is unknown. Consequently, a common task in sampling problem is to estimate this constant, thus enabling to approximate the ground-truth normalised density.

This normalisation constant can be unbiasedly estimated in path space, by applying Proposition 2.3. in Richter and Berner (2023), which states that

$$\frac{d\mathbb{P}_{\overleftarrow{Y}^v}}{d\mathbb{P}_{X^u}}(X) = \frac{1}{Z} \exp \left( - \left( R_{f_{u,v,u}}^{\text{Bridge}} + S_{u+v} + B \right) \right) (X), \quad (3.26)$$

where  $R_{f_{u,v,u}}^{\text{Bridge}}$ ,  $S_{u+v}$  and  $B$  only depend on the coefficients of the inference and generative SDEs and the unnormalised density  $\rho$ . Moreover, since

$$\mathbb{E}_{\mathbb{P}_{X^u}} \left[ \frac{d\mathbb{P}_{\overleftarrow{Y}^v}}{d\mathbb{P}_{X^u}}(X^u) \right] = 1, \quad (3.27)$$

and taking expectations in Equation 3.26

$$Z = \mathbb{E}_{\mathbb{P}_{X^u}} \left[ \exp \left( - \left( R_{f_{u,v,w}}^{\text{Bridge}} + S_{u+v} + B \right) \right) (X^u) \right]. \quad (3.28)$$

Therefore, by applying Jensen's inequality,  $\log Z$  can be bounded bellow as

$$\log Z \geq \mathbb{E}_{\mathbb{P}_{X^u}} \left[ - \left( R_{f_{u,v,w}}^{\text{Bridge}} + S_{u+v} + B \right) (X^u) \right]. \quad (3.29)$$

Moreover, if the controls are optimal, i.e.  $\frac{d\mathbb{P}_{\overleftarrow{Y}^v}}{d\mathbb{P}_{X^u}} = 1$ , a zero-variance estimator is obtained:

$$Z = \exp \left( - \left( R_{f_{u^*,v^*,u^*}}^{\text{Bridge}} + S_{u^*+v^*} + B \right) \right) (X^{u^*}), \quad X^{u^*} \sim \mathbb{P}_{X^{u^*}} \quad (3.30)$$

Similarly, the normalisation constant can be estimated by discretising the trajectories using the integrators presented in the former section. For instance, for the Controlled Annealed Langevin Sampler, it is verified that

$$\log Z \geq \mathbb{E}_{X^u} \left[ \log \left( \frac{\rho(X_{\tau_T}) \prod_{i=0}^{T-1} \mathcal{N}(X_{\tau_i}; X_{\tau_{i+1}} + (\sigma^2 \overleftarrow{\delta}_\theta - \overleftarrow{\Phi}_\omega)(X_{\tau_{i+1}}, \tau_{i+1}) \Delta_{\tau_{i+1}}, 2\sigma^2 \Delta_{\tau_{i+1}} I)}{p_{\text{prior}}(X_{\tau_0}) \prod_{i=0}^{T-1} \mathcal{N}(X_{\tau_{i+1}}; X_{\tau_i} + (\sigma^2 \overleftarrow{\delta}_\theta + \overleftarrow{\Phi}_\omega)(X_{\tau_i}, \tau_i) \Delta_{\tau_{i+1}}, 2\sigma^2 \Delta_{\tau_{i+1}} I)} \right) \right]. \quad (3.31)$$

Consequently, the normalisation constant for the unnormalised target density can be unbiasedly approximated via a Monte Carlo estimate of this expectation. Since the variance of this estimator depends on the divergence between the measures induced by the SDEs (Hartmann and Richter, 2024), it can be expected that the optimisation of the objectives presented before can lead to lower variance estimates of the normalisation constant.

### 3.4.2 Effective Sample Size

As explained before, importance sampling is a method which enables to compute unbiased expectations for statistics of a target distribution. This is done by re-weighting the expectation by the Radon–Nikodym derivative of the measures

$$\mathbb{E}_{\mathbb{P}_{\overleftarrow{Y}_0}}[\phi(X)] = \mathbb{E}_{\mathbb{P}_{X^u}}[\phi(X) \underbrace{\frac{d\mathbb{P}_{\overleftarrow{Y}_0}}{d\mathbb{P}_{X^u}}(X)}_{w^u(X)}]. \quad (3.32)$$

These weighting factors are called importance sampling weights and their variance determines, to a great extent, the quality of the IS estimate of the target statistic (Neal, 2001). In path space, these are defined as

$$w^{u,v} = \frac{d\mathbb{P}_{\overleftarrow{Y}^v}}{d\mathbb{P}_{X^u}}(X^u), \quad (3.33)$$

but if the partition function of  $p_{\text{target}}$  is unknown, only the unnormalised IS weights can be computed

$$\hat{w}^{u,v} = Z \cdot \frac{d\mathbb{P}_{\overleftarrow{Y}^v}}{d\mathbb{P}_{X^u}}(X^u) \equiv \widehat{\frac{d\mathbb{P}_{\overleftarrow{Y}^v}}{d\mathbb{P}_{X^u}}}(X^u), \quad (3.34)$$

which do not depend on  $Z$  (Richter and Berner, 2023, Appendix A.12). Moreover, since  $\mathbb{E}_{\mathbb{P}_{X^u}}[\hat{w}^{u,v}] = Z$ , the initial expectation can be expressed in terms of the unnormalised IS weights

$$\mathbb{E}_{\mathbb{P}_{\overleftarrow{Y}_0}}[\phi(X)] = \frac{\mathbb{E}_{\mathbb{P}_{X^u}}[\phi(X) \hat{w}^{u,v}(X)]}{\mathbb{E}_{\mathbb{P}_{X^u}}[\hat{w}^{u,v}(X)]}. \quad (3.35)$$

If we were to approximate this expectation via Monte Carlo, the variance of the estimator would be inversely proportional to the number of independent samples used to approximate it. However, for the IS estimator, since the samples are drawn, instead, from the proposal distribution, the effective sample size (ESS) is  $N/(1 + \mathbb{V}(\hat{w}^{u,v}))$ , where  $N$  denotes the number of samples drawn from the proposal (Neal, 2001, Equation 18). Consequently, low ESS implies that Monte Carlo estimators relying on those samples are likely to suffer from high variance.

Therefore, since the variance of the normalised importance weights is expected to decrease as we optimise the controls, we also rely on these metric to evaluate the proposed approaches.

In our implementation, instead of computing the RND directly, we compute its logarithm to ease numerical stability, thus obtaining the log-unnormalised IS weights  $\log \hat{w}^{u,v}$ . Consequently, the normalised IS weights can be computed as

$$w_i^{u,v} = \exp \left( \log \hat{w}_i^{u,v} - \underbrace{\log \left( \sum_{j=1}^N \exp(\hat{w}_j^{u,v}) \right)}_{\text{LogSumExp}(\hat{w}^{u,v})} + \log N \right), \quad (3.36)$$

thus relying on operations with numerically stable primitives.

### 3.4.3 Wasserstein Distance

For sub-optimal controls, there exists the possibility that the support of the marginal distribution at the last step of the generative SDE is disjoint with that of the target distribution, e.g. if a mode is missed. In this scenario, optimal transport distances compare favourably to other f-divergences (Cuturi, 2013), such as the Kullback-Leibler  $KL(P||Q)$ , which require absolute continuity for the measures ( $P \ll Q$ ).

From a high-level perspective, this family of distances quantify the minimum cost of transporting one mass of probability into another. In this work, we use the 2-Wasserstein distance  $\mathcal{W}_2^\gamma$  (so the cost is defined in terms of the Euclidean distance) with entropic regularisation (Cuturi, 2013), where  $\gamma$  denotes the regularisation weight.

### 3.4.4 Benchmark Targets

**1-D Gaussian Mixture Model (GMM)** : We define a simple 1-D GMM, which despite being a simple target, is helpful to develop visualisations. This is defined as

$$\rho(x) = \frac{1}{2} \mathcal{N}(x; -1, 0.2^2) + \frac{1}{2} \mathcal{N}(x; 1, 0.2^2). \quad (3.37)$$

**2-D Gaussian Mixture Model (GMM):** We consider the standardised bi-dimensional GMM with 40 components in Vargas et al. (2024)

$$\rho(x) = \sum_{m=1}^{40} \pi_m \mathcal{N}(x; \mu_m, \sigma_m^2 I). \quad (3.38)$$

Note that, despite the simplicity of this target, it is particularly interesting due to its multimodality, and the fact that the modes can be separated by low-probability regions in which methods, such as Metropolis-Hastings, can become stuck.

Moreover, the marginal for the forward diffusion at each noise scale (see Section 2.1), also follows a GMM density:

$$q(x_t) = \sum_{m=1}^{40} \pi_m \mathcal{N}(x; \alpha_t \mu_m, (\sigma_t^2 + \alpha_t \sigma_m^2) I). \quad (3.39)$$

Consequently, this target can be used to conduct ablations where the only source of error is due to the discretisation SDE, thus allowing to evaluate, for instance, if the choice of the discretisation steps can lead to improved performance.

**Funnel:** This is a 10-dimensional target, which is commonly used in the evaluation of MCMC approaches (Neal, 2003). This is defined as

$$\rho(x) = \mathcal{N}(x_1; 0, \sigma^2) \prod_{i=2}^{10} \mathcal{N}(x_i; 0, e^{x_1}). \quad (3.40)$$

Note the score of the Funnel distribution is

$$\nabla_{x_1} \log \rho(x) = -\frac{x_1}{\sigma^2} + e^{-x_1} \sum_{i=2}^{10} \|x_i\|^2, \quad \nabla_{x_i} \log \rho(x) = -\frac{x_i}{e^{x_1}}, \quad (3.41)$$

so its magnitude varies exponentially with the value of  $x_1$ , making its estimation challenging, specially at low noise levels (De Bortoli et al., 2024).

**Double Well:** We consider the bi-dimensional Double Well distribution in Wu et al. (2020), which is defined by the log-density

$$\log p(x_1, x_2) = -x_1^4 + 6x_1^2 + \frac{1}{2}x_1 - \frac{1}{2}x_2^2 + C. \quad (3.42)$$

This distribution has two modes, separated by a low-probability region, often referred to as *energy barrier*, in the context of molecular dynamics. These energy barriers often result in slow convergence for methods based on the simulation of a Langevin SDE, so a high number of iterations are required to converge to a mode, corresponding to a stable configuration of a molecule.

To increase the difficulty of this target, we consider the 4-particle Double Well potential (Akhound-Sadegh et al., 2024), obtained from the product of 4 copies of the Double Well distribution. Therefore, the target distribution has  $2^4 = 16$  modes, well-separated by energy barriers.

Using the same approach as Midgley et al. (2022), samples from this distribution are obtained by independently sampling from each copy, and using rejection sampling to obtain unbiased samples from the marginal  $p(x_1)$ . Sampling from  $p(x_2)$  is straightforward, as it follows a standard normal distribution.

Additionally, the Funnel and Double Well datasets were scaled to the interval  $[-1, 1]^d$ , to ease training of the score model. Thus, the change of variable theorem was applied to obtain the log-densities for the transformed variables. These benchmark targets, along with ground-truth samples, are shown in Figure A.1.

## 3.5 Experimental Settings

The experimental settings used to conduct the evaluation of the proposed samplers are summarised in Table 3.5. Following, we delve deeper into the specific architectural choices and training procedure.

### 3.5.1 Score-model Training

The evaluation of the presented approaches requires training a score model before-hand, so it can be incorporated in the control of the generative SDE. The architecture used for the score model is a multi-layer perceptron with skip connections, inspired in model in Kingma et al. (2021).

To encode the time-step, we use fixed Fourier embeddings, inspired in the positional encoding in Ott et al. (2019), which are defined as

$$\text{emb}(x) = \left[ \sin(\omega^0 x), \dots, \sin(\omega^{\frac{d}{2}} x), \cos(\omega^0 x), \dots, \cos(\omega^{\frac{d}{2}} x) \right]. \quad (3.43)$$

Moreover, since the goal is to improve the likelihood of the data generated by the model, and considering that the first noising steps have a major impact on the VLB (Nichol and Dhariwal, 2021), it is particularly relevant to capture the fine perturbations that occur at low noise levels. To this end, we compute an additional set of high-frequency Fourier features to amplify these small perturbations, similarly to Kingma et al. (2021). These, along with the time embeddings and the noisy observation, constitute the input to the network.

Additionally, instead of modeling the score  $s_\theta(x, t)$  directly, the network is trained to predict the noise  $\varepsilon_\theta(x, t)$  that was added to the observation. This preconditioning eases training, since the score magnitude varies considerably across noise levels. In this regard, the score can be recovered as  $s_\theta(x, t) = \varepsilon_\theta(x, t) / \sigma_t$ .

To train the noise-prediction network, instead of optimising the variational lower bound directly, we opt for using the unweighted variant

$$\mathcal{L} = \frac{1}{2} \mathbb{E}_{X_0 \sim p_{\text{target}}, \varepsilon \sim \mathcal{N}(0, I), t \sim U(0, 1)} \left[ \|\varepsilon_t - \hat{\varepsilon}_t^\theta(\alpha_t X_0 + \sigma_t \varepsilon, t)\|^2 \right], \quad (3.44)$$

which generally exhibits lower gradient variance compared to the VLB objective (Nichol and Dhariwal, 2021). To further reduce the variance in the loss computation, we used the low-discrepancy sampler for the steps  $t$  in Kingma et al. (2021). Therefore, for a batch of  $N$  elements, we sample  $u_0 \sim U(0, 1)$  and select the time-step as  $t_i = \text{mod}(u_0 + i/N, 1)$ , thus obtaining a more uniform partition of the interval  $[0, 1]$  which helps to reduce the objective variance. For the noise schedule, we consider a continuous approximation to the discrete schedule in (Song and Ermon, 2019).

We train the score model using the Adam optimiser (Kingma and Ba, 2014) with moment decay rates  $\beta_1 = 0.9$  and  $\beta_2 = 0.999$ , gradient clipping and tuning the learning for each benchmark. To improve training stability, we compute an exponential moving average of the parameters, setting the momentum parameter to  $m = 0.999$ .

Finally, we would like to highlight that we did not perform an extensive hyper-parameter sweep, or consider introducing more advanced techniques for diffusion model training (such as using a learnable noise schedule), since the focus of the work is improving the quality of the samples generated by a sub-optimal diffusion model.

### 3.5.2 Control Architecture

Once the score-model is trained, its sample quality can be improved by optimising the control in the generative SDE.

For the parameterisation of the control, we rely on the PIS-NN architecture, presented in (Zhang and Chen, 2021). This is a multi-layer perceptron with GeLU activations (Hendrycks and Gimpel, 2016). The input to this network is the current iterate in the integration trajectory, as well as the discretisation step, which is encoded using Fourier embeddings, similarly to the score-model. At initialisation, the last layer is initialised to zero so, before training, both the inference and generative SDEs are uncontrolled, which was empirically found to stabilise training.

In Section 3.2, it was argued that learning the variance of the generative SDE for the Controlled Ancestral Sampler could be beneficial, as relying on the noise variance of the generative SDE is only optimal in the limit of infinite discretisation steps ( $T \rightarrow \infty$ ). To this end, we parameterise the variance as the interpolation

$$\Sigma(X_t, t) = \exp\left(v_\omega(X_t, t)\sigma_{t|s}^2 + (1 - v_\omega(X_t, t))\sigma_{t|s}^2 \frac{\sigma_s^2}{\sigma_t^2}\right), \quad (3.45)$$

and parameterise  $v_\omega$  using a 3-layer perceptron, with skip-connections. The soft-plus activation is used for the intermediate outputs, while the sigmoid activation is added after the last linear layer to constrain the network outputs to the interval  $[0, 1]$ .

Finally, for the Controlled Annealed Langevin Sampler, the diffusion coefficient  $\sigma$ , which modulates the noise in both the inference and generative SDEs, needs to be properly adjusted. To highlight its importance, consider the Euler-Maruyama discretisation of the generative SDE:

$$X_{\tau_{i+1}} = X_{\tau_i} + (\sigma^2 \overleftarrow{s}_\theta + \overleftarrow{\Phi}_\omega)(X_{\tau_i}, \tau_i)\Delta\tau_{i+1} + \sigma\sqrt{2}\underbrace{(W_{\tau_{i+1}} - W_{\tau_i})}_{\sim \mathcal{N}(0, \Delta\tau_{i+1}I)}. \quad (3.46)$$

In particular, several possibilities are explored for  $\sigma$ : (1) using a (possibly learned) single value across all noise scales, (2) annealing  $\sigma$  using a fixed schedule, (3) jointly learning  $\sigma$ , along with the control.

Regarding the former choice, we parameterise  $\sigma\Delta\tau_{i+1}$  (note that the step size is embedded into the parameterisation) using a multi-layer perceptron with GeLU activations, whose inputs are the Fourier embeddings of the discretisation points  $\tau_{i+1}$  and  $\tau_i$ .

## 3.6 Evaluation

In this section, we present the numerical results obtained for the Controlled Ancestral Sampler and the Controlled Annealed Langevin Sampler. For each sampler, sample quality is evaluated across a number of performance metrics: mean log-likelihood, Wasserstein distance, effective sample size and estimation of normalisation constants. Given that samples from each target distribution are available, these are used to compute Monte-Carlo estimators for those metrics that cannot be easily computed analytically, such as the mean log-likelihood and the Wasserstein distance, and these estimators are used as ground-truth values for comparison with the proposed approaches. For the normalisation constant, since it is known for all the evaluated benchmark, its analytic value is used in the comparison.

Following the approach of (Richter and Berner, 2023), the samplers were trained for the two divergences presented in Section 3.1, which are repeated for reference:

- **Kullback-Leibler divergence**

$$D_{KL}(\mathbb{P}_X^u || \mathbb{P}_{\bar{Y}_v}) := \mathbb{E}_{\mathbb{P}_X^u} \left[ \log \frac{\mathbb{P}_X^u}{\mathbb{P}_{\bar{Y}_v}} \right]. \quad (3.47)$$

This is the standard choice in the sampling literature, but is prone to ignoring areas with small probability mass, a phenomenon referred to as *mode-seeking* behaviour (Midgley et al., 2022; Minka et al., 2005).

- **Log-variance divergence**

$$D_{LV}(\mathbb{P}_X^u || \mathbb{P}_{\bar{Y}_v}) := \mathbb{V}_{\mathbb{P}_X^u} \left[ \log \frac{\mathbb{P}_X^u}{\mathbb{P}_{\bar{Y}_v}} \right]. \quad (3.48)$$

This divergence involves a reference measure, which we set to the measure induced by the learnable controlled generative SDE, i.e.  $\mathbb{P}_X^u$ , thus effectively training the control in an on-policy fashion. For this setting, this loss can be regarded as a control-variate variant of  $D_{KL}$ , thus potentially exhibiting lower variance.

For each sampler, the performance metrics before and after training are reported, and these are compared with their corresponding ground-truth values. The numerical results are summarised in Tables 3.2 (GMM, d=1), 3.1 (GMM, d=2), 3.3 (Funnel, d=10) and 3.4 (Many Well, d=8). Since each method admits multiple variants (e.g. the discretisation points can be optimised along with the controls), each configuration is evaluated separately, thus enabling to identify which factors contribute the most to sample quality. To distinguish each configuration, we include the following legend:

- $\beta$ . Parameters defining the discretisation points  $(\tau_1, \dots, \tau_T)$  are optimised.
- $\Phi$ . The network controlling both the inference and generative SDE is learnable. Otherwise, it is set to zero.
- $\Delta$  (only available for the Controlled Langevin sampler). The diffusion coefficient is controlled by a fixed schedule. Otherwise, it is set to a fix value, shared across noise levels.
- $\Delta_\omega$  (only available for the Controlled Langevin sampler). Indicates that the diffusion coefficient is learnable.
- $\sigma$  (only available for the Ancestral sampler). Indicates that the variance of the generative SDE is learnt.

Note that the numerical results are reported for a fixed number of discretisation steps: 8 for the 1-D GMM, and 32 for the rest of benchmark targets. For the sake of conciseness, we report the results for varying discretisation steps in Appendix A. Moreover, we want to highlight that this number of steps is relatively low, even for the simple benchmark targets that are being considered (e.g. Berner et al. (2022) use 256 discretisation steps, although their problem is arguably harder, as they do not have access to samples from the target to train a diffusion model). In this setting, the sample quality before training is generally poor, as illustrated in the performance metrics reported in Tables 3.2, 3.1, 3.3 and 3.4. This situation is specially clear when evaluating the samples qualitatively. Consequently, in the following subsection, we analyse into greater detail the results obtained for each target.

### 3.6.1 2-D GMM

The numerical results for the 2-D GMM with 40 modes (Vargas et al., 2024) are reported in Table 3.1. First of all, it is relevant to highlight that the mean log-likelihood  $\mathbb{E}_{\mathbb{P}_{x_1^0}}[\log p_{\text{target}}(x)]$  (denotes as **LL** in Table 3.1) is significantly lower than the expected ground-truth value before training (e.g. its value for the ancestral sampler is below  $-15$ , while the ground-truth value surpasses 1). The inspection of the samples generated by the model before optimisation, shown in Figure 3.2 (left), shows that, although all the modes of the target distribution are captured, the marginal distribution  $\mathbb{P}_{x_1^0}$  of the generative SDE overestimates the variance of the target distribution, as it is qualitatively reflected in the greater dispersion of the samples, as compared to those obtained from the target distribution (see Figure A.1). Moreover, for the three samplers evaluated, the effective sample size is quite low at initialisation ( $< 5\%$ ),

and it does not significantly improve after training. This issue is further discussed in Section 3.7.3.

However, it is possible to considerably improve sample quality by optimisation of the divergence objective, as it illustrated in Figure 3.2 (right), which shows the samples generated by the Controlled Ancestral sampler (with learnable variance), after tuning the control and sampling trajectory to minimise the Kullback-Leibler divergence between the discretised measures induced by the inference and generative SDEs. Specifically, as it is reflected in Table 3.1, the performance metrics for the Controlled Ancestral sampler can significantly improve, as compared to their initial values at initialisation, when training either the discretisation points on isolation, or the control and the discretisation trajectory simultaneously. Notably, only training the control does not significantly improve sample quality, as demonstrated by the higher Wasserstein distance and the mean log-likelihood remaining low.

The importance of tuning the discretisation trajectory can be understood from the perspective of the rate at which the structure in the data is destroyed, due to its perturbation with additive Gaussian noise. Specifically, this rate is modulated by the noise schedule, which uniquely defines the drift and diffusion coefficients of the variance-preserving inference SDE. Moreover, in this work, we rely on a continuous approximation to the noise schedule used by Song and Ermon (2019), which can corrupt the data too swiftly (Nichol and Dhariwal, 2021).

This phenomenon is illustrated in Figure 3.4, which shows the marginals of the stochastic process induced by the inference SDE at different time steps. Note that, even at  $t = 0.2$ , the marginal distribution is still uni-modal, closely resembling the standard Gaussian (which constitutes the prior distribution, i.e.  $p_{\text{prior}} = \mathcal{N}(0, I)$ ). Consequently, it can be argued that the Euler-Maruyama updates occurring within that interval will not significantly change the marginal measures, and progress towards the target measure will be slow. This is particularly clear for the Controlled Annealed Langevin sampler, whose discretisation is given by

$$X_{\tau_{i+1}} = X_{\tau_i} + (\sigma^2 \overleftarrow{s}_\theta + \overleftarrow{\Phi}_\omega)(X_{\tau_i}, \tau_i) \Delta\tau_{i+1} + \sigma \sqrt{2\Delta\tau_{i+1}} \varepsilon_{\tau_{i+1}}, \varepsilon_{\tau_{i+1}} \sim \mathcal{N}(0, I). \quad (3.49)$$

In this regard, if  $X_{\tau_i} \sim p_{\text{prior}} = \mathcal{N}(0, I)$ ,  $s_\theta(x, t) \approx \mathcal{N}(x; 0, I)$ , and the control is zero (as it is the case at initialisation), this update corresponds to an ULA transition (Roberts and Tweedie, 1996), which leaves the measure  $\mathcal{N}(0, I)$  invariant. Consequently,  $X_{\tau_{i+1}} \sim \mathcal{N}(0, I)$ , and no progress is made towards the target distribution.

Therefore, placing the discretisation steps uniformly within the interval  $[0, 1]$  can lead to insignificant updates within the first steps, due to the score being close to that of the prior distribution, while taking too aggressive steps for the less noisy levels, when the score changes more significantly between discretisation steps. Consequently, given the small number of Euler-Maruyama steps used, it is sensible to expect that it might not be possible

to compensate for overly large step sizes at low noise levels exclusively by optimising the control, as the numerical evidence suggests for the evaluated benchmarks.

Additionally, to further illustrate this point, we study the evolution of the discretisation trajectory  $\tau_i$  by measuring the difference between their values at each step and their uniform initialisation values, as shown in Figure 3.1. Since this difference is positive and increases quite steadily during training for both runs, it indicates that the steps tend to accumulate themselves at low noise levels, which is consistent with the observation that smaller steps can be beneficial to simulate the SDE, due to the faster variation of the score at low noise levels.

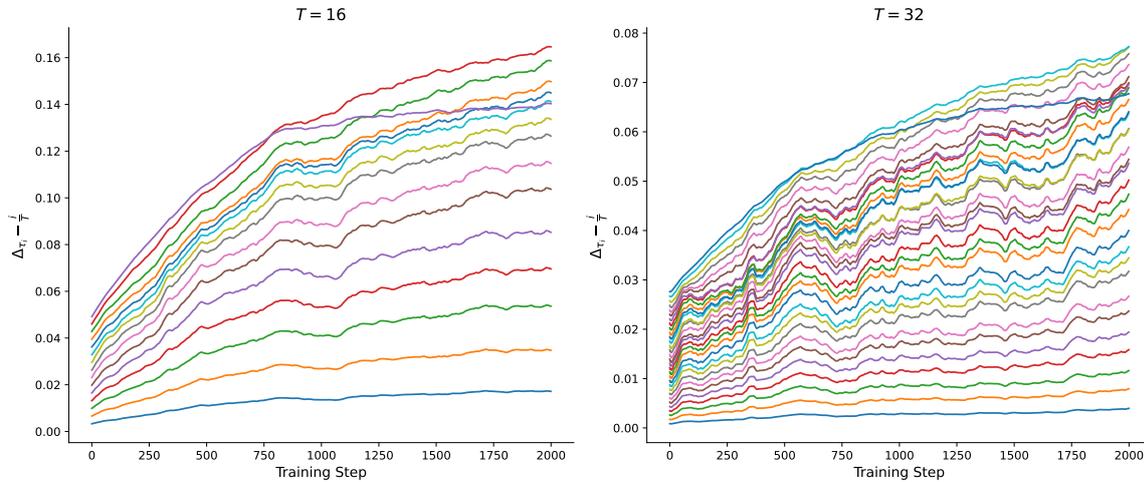


Fig. 3.1 Evolution of the difference between the discretisation points  $\tau_i$  and their initial values  $i/T$  for  $T = 16$  (left) and  $T = 32$  (right). These correspond to Controlled Ancestral sampler runs, with fixed variance.

It is remarkable to highlight that, while the log-variance divergence performance is similar to that of the Kullback-Leibler loss for the Controlled Ancestral sampler with fixed variance, this is not the case when the variance of the inference SDE is learnt. For this setting, Table 3.1 shows that the estimators of the normalisation constant are highly biased (e.g.  $\approx 8.099$  for the variant  $\beta/\Phi$ ). Moreover, the mean log-likelihood stays below  $-10$ , thus not improving with respect to its value at initialisation. Moreover, Figure 3.3 compares the samples obtained when optimising each objective, thus evidencing that, despite relying on the same sampler, their performance can greatly differ depending on the divergence choice. This anomalous behaviour can be explained in terms of the non-uniqueness of the optimal control problem, as both the Kullback-Leibler and log-variance divergence only enforce Nelson’s identity (Nelson, 2020) (see Section 2.4). In this regard, specific choices of the controls were used for the proposed samplers to render the solution to the problem unique. However, we hypothesise that relying on a learnable variance in the generative SDE for the ancestral sampler can,

effectively, decouple the controls, thus restoring non-uniqueness and leading to pathological optimum, as the one shown in Figure 3.3 (right).

Finally, it is relevant to highlight that the performance of the Controlled Annealed Langevin sampler is quite poor for all the configurations evaluated in Table 3.1. Moreover, there is an important gap in performance when comparing the metrics obtained at initialisation of the ancestral and Langevin samplers. This is particularly clear when comparing the mean log-likelihood at initialisation, as it is an order of magnitude lower for the Langevin sampler. To some extent, this is explained by the fact that, while the uncontrolled inference and generative SDEs for the ancestral sampler approximately transport the target distribution into the prior distribution, and viceversa by construction (as they are defined in terms of the variance-preserving SDE), this is not the case for the annealed Langevin sampler, whose score is only accurate in the terminal points, similarly to the CMCD sampler (Vargas et al., 2024). In this regard, it is observed that the Controlled Annealed Langevin sampler generally requires careful hyperparameter tuning to match the performance of the ancestral sampler.

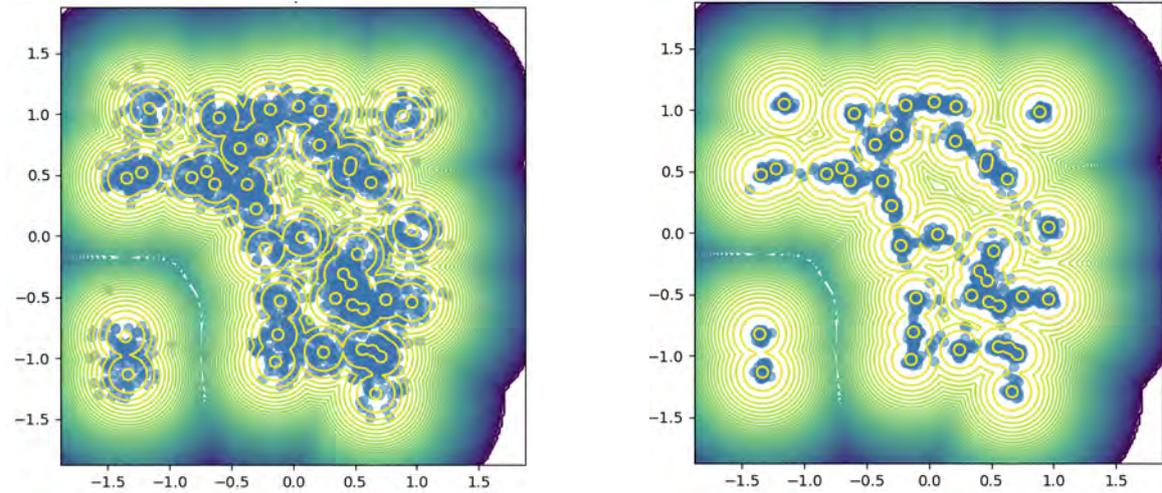


Fig. 3.2 Qualitative comparison of sample quality for the ancestral sampler method, with learnable backward variance, using 32 discretisation steps, before training (left) and after optimisation of the control and discretisation steps (right).

### 3.6.2 1-D GMM

The numerical results for the 1-D GMM benchmark target are outlined in Table 3.2. Given the simplicity of this target, only 8 discretisation steps are used, to increase the difficulty of the sampling problem. Despite the reduced number of steps, it is possible, after optimisation

Sampler	Variant	Loss	$\log Z$	$\mathcal{W}_2^\gamma \downarrow$	ESS $\uparrow$	LL $\uparrow$
Target	-	-	0.000	0.009 $\pm$ 0.001	-	1.047 $\pm$ 0.040
Ancestral	Init	-	-0.216 $\pm$ 0.266	0.010 $\pm$ 0.001	0.048 $\pm$ 0.030	-10.719 $\pm$ 0.760
	$\beta$	KL	-0.248 $\pm$ 0.543	<b>0.010<math>\pm</math>0.001</b>	0.061 $\pm$ 0.041	-1.079 $\pm$ 0.311
		LV	-0.288 $\pm$ 0.488	0.010 $\pm$ 0.001	<b>0.075<math>\pm</math>0.049</b>	-0.920 $\pm$ 0.260
	$\Phi$	KL	<b>-0.107<math>\pm</math>0.693</b>	0.013 $\pm$ 0.002	0.046 $\pm$ 0.024	-10.640 $\pm$ 0.586
		LV	-3.155 $\pm$ 0.945	0.035 $\pm$ 0.003	0.014 $\pm$ 0.009	-11.815 $\pm$ 0.538
	$\beta/\Phi$	KL	-0.304 $\pm$ 0.294	0.025 $\pm$ 0.003	0.050 $\pm$ 0.036	-0.915 $\pm$ 0.274
		LV	-0.384 $\pm$ 0.299	0.013 $\pm$ 0.002	0.042 $\pm$ 0.028	<b>-0.763<math>\pm</math>0.242</b>
	Ancestral/ $\sigma$	Init	-	-0.136 $\pm$ 0.270	0.010 $\pm$ 0.001	0.044 $\pm$ 0.020
$\beta$		KL	-0.235 $\pm$ 0.516	<b>0.010<math>\pm</math>0.001</b>	0.066 $\pm$ 0.046	<b>-0.874<math>\pm</math>0.273</b>
		LV	27.100 $\pm$ 8.208	0.010 $\pm$ 0.001	0.003 $\pm$ 0.002	-13.487 $\pm$ 0.662
$\Phi$		KL	<b>-0.109<math>\pm</math>0.492</b>	0.010 $\pm$ 0.002	0.043 $\pm$ 0.029	-10.699 $\pm$ 0.667
		LV	3.818 $\pm$ 3.205	0.137 $\pm$ 0.009	0.005 $\pm$ 0.005	-11.546 $\pm$ 0.486
$\beta/\Phi$		KL	-0.251 $\pm$ 0.472	0.010 $\pm$ 0.001	<b>0.067<math>\pm</math>0.048</b>	-0.936 $\pm$ 0.264
		LV	8.099 $\pm$ 3.939	0.127 $\pm$ 0.009	0.004 $\pm$ 0.002	-10.481 $\pm$ 0.571
Langevin		Init	-	-0.592 $\pm$ 0.554	0.015 $\pm$ 0.001	0.015 $\pm$ 0.006
	$\beta$	KL	<b>0.002<math>\pm</math>0.270</b>	0.012 $\pm$ 0.001	0.031 $\pm$ 0.011	-51.325 $\pm$ 3.306
		LV	-0.117 $\pm$ 0.570	0.012 $\pm$ 0.001	0.030 $\pm$ 0.012	-49.970 $\pm$ 2.910
	$\Phi$	KL	-6.552 $\pm$ 2.534	0.021 $\pm$ 0.001	0.004 $\pm$ 0.001	-92.066 $\pm$ 7.509
		LV	*	*	*	*
	$\beta/\Phi$	KL	-0.404 $\pm$ 0.941	0.034 $\pm$ 0.003	0.012 $\pm$ 0.006	-44.500 $\pm$ 2.590
		LV	-0.882 $\pm$ 1.056	0.068 $\pm$ 0.006	0.011 $\pm$ 0.006	-27.360 $\pm$ 1.421
	$\Phi/\Delta$	KL	-0.943 $\pm$ 0.976	0.118 $\pm$ 0.009	0.024 $\pm$ 0.015	-13.323 $\pm$ 0.677
LV		*	*	*	*	
$\beta/\Phi/\Delta$	KL	-0.136 $\pm$ 0.444	<b>0.011<math>\pm</math>0.002</b>	<b>0.034<math>\pm</math>0.014</b>	<b>-10.111<math>\pm</math>0.435</b>	
	LV	*	*	*	*	

Table 3.1 Numerical results for varying samplers and configurations in the GMM ( $d=2$ ) benchmark problem using 32 discretisation steps. Performance metrics were estimated from 500 samples for 30 different seeds, and means and standard deviations are reported. First row indicates the ground-truth values for the performance metrics: analytical value of the normalisation constant ( $\log Z$ ) and Monte-Carlo estimators for the Wasserstein distance ( $\mathcal{W}_2^\gamma$ ) and log-likelihood (LL) using ground-truth samples. The arrows  $\uparrow$  and  $\downarrow$  indicate whether the performance metrics are maximised or minimised. Variant *Init* denotes the performance metrics before training. Bolded values indicate the best performance within each sampler. KL stands for Kullback-Leibler divergence, while LV corresponds to the log-variance divergence loss (Vargas et al., 2024). Divergent runs are signaled with  $*$ .

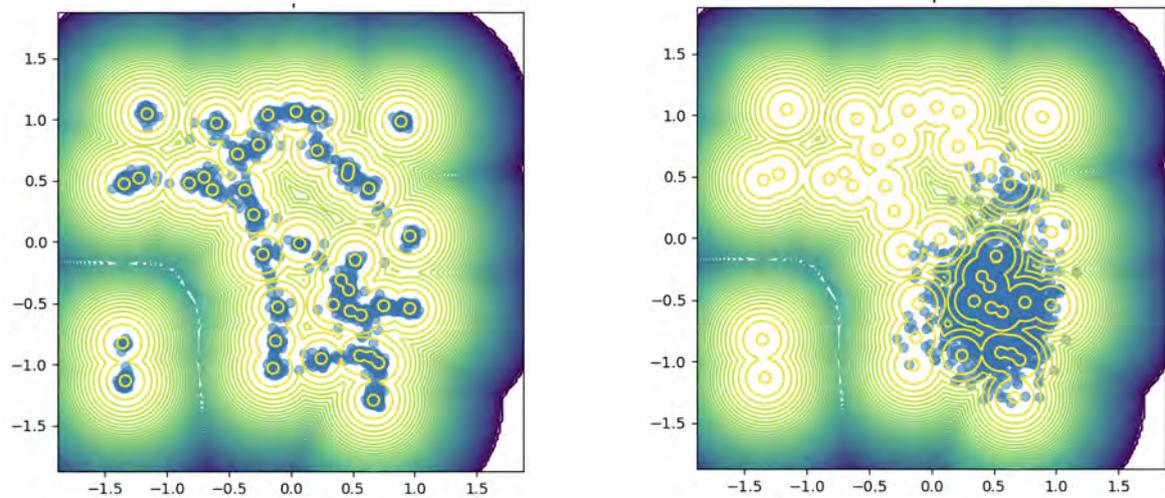


Fig. 3.3 Qualitative comparison of sample quality for the ancestral sampler method, with learnable backward variance, using 32 discretisation steps, after optimisation using the Kullback-Leibler loss (left) and the log-variance divergence (right).

of the sampling trajectory and controls, to generate samples matching, almost perfectly, the target distribution, as illustrated for the Controlled Ancestral sampler in Figure 3.5.

From a high-level perspective, the numerical results for this target are aligned with the observations made for the 2-D GMM, such as the importance of training the discretisation points and pathological behaviour of the log-variance objective for the ancestral sampler with learnable variance.

However, two main differences can be observed. First, the effective sample sizes are much higher for the three samplers, as it can be expected given the lower dimensionality of the problem. Moreover, the performance of the annealed Langevin sampler is closer to that of the ancestral sampler, surpassing the latter in terms of effective sample size, for some configurations.

To provide additional insight on the performance of the annealed Langevin sampler, Figure 3.6 qualitatively compares the samples generated by the model before and after training the sampling trajectory and controls. First of all, note that the samples generated before training are approximately distributed as a standard normal, which constitutes the prior distribution. This can be expected considering that, at initialisation, the generative SDE is uncontrolled (thus recovering the annealed Langevin algorithm (Thin et al., 2021, Equation 14)), with the score changing slowly for the first steps, for an uniform initialisation, so that only 8 updates are not sufficient to transport the prior measure to the target distribution.

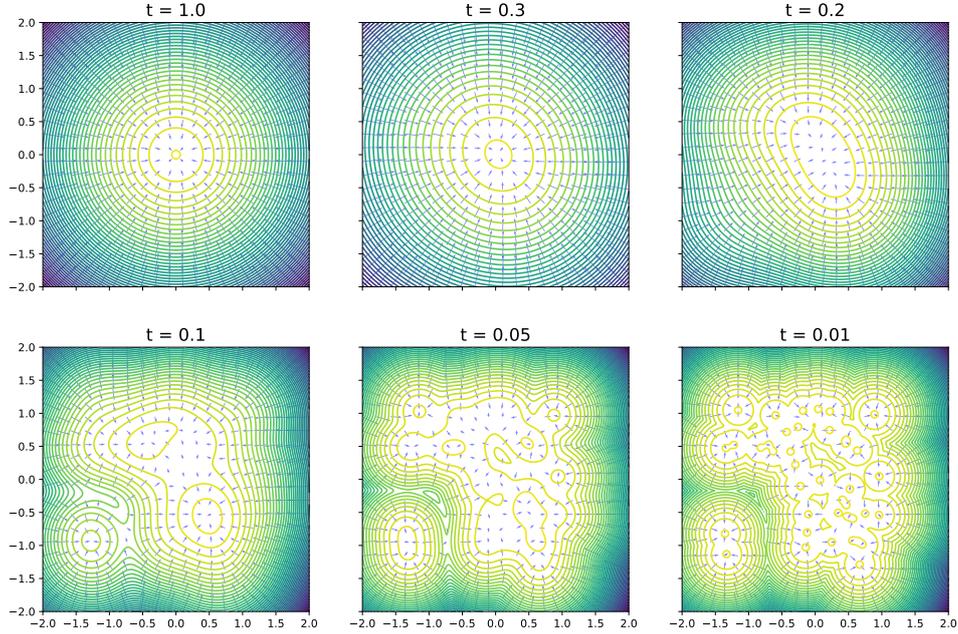


Fig. 3.4 Density contours for the marginals of the variance-preserving SDE, using as target the 2-D GMM distribution, evaluated at varying noise scales. The vector field represents the score  $\nabla_x p_{X_t}(x, t)$  at each grid position.

On the other hand, after training, the generated samples match more closely the target distribution, although its marginal variance is slightly higher (see Figure 3.5, right). However, the overestimation of the variance can be beneficial in terms of the effective sample size. Precisely, if the proposal distribution  $\mathbb{P}_{X^u}$  has heavier tails than those of the target measure  $\mathbb{P}_{\tilde{Y}^v}$ , the density ratio  $d\mathbb{P}_{\tilde{Y}^v}/d\mathbb{P}_{X^u}$  for the discretised measures is bounded (assuming absolute continuity) and, consequently, the variance of the importance sampling weights. Therefore, as the effective sample size is inversely proportional to the variance of the normalised importance weights, a higher effective sample size might be expected in this scenario. This is generally the case for the samples generated by the controlled annealed Langevin sampler in this target, which might explain the slightly better effective sample sizes obtained by this method.

### 3.6.3 Funnel

The numerical results for the Funnel target are summarised in Table 3.3. Results for the Langevin sampler are not included due to divergence of the runs. As highlighted in Section

Sampler	Variant	Loss	log Z	$\mathcal{W}_2^Y \downarrow$	ESS $\uparrow$	LL $\uparrow$
Target	-	-	0.000	0.007 $\pm$ 0.004	-	-0.513 $\pm$ 0.039
Ancestral	Init	-	-0.101 $\pm$ 0.072	<b>0.008<math>\pm</math>0.002</b>	0.360 $\pm$ 0.133	-1.584 $\pm$ 0.079
	$\beta$	KL	-0.063 $\pm$ 0.089	0.007 $\pm$ 0.003	0.365 $\pm$ 0.181	-0.662 $\pm$ 0.037
		LV	-0.107 $\pm$ 0.081	0.007 $\pm$ 0.003	0.315 $\pm$ 0.139	-0.731 $\pm$ 0.045
	$\Phi$	KL	-0.079 $\pm$ 0.045	0.007 $\pm$ 0.001	0.445 $\pm$ 0.094	-1.436 $\pm$ 0.070
		LV	-0.079 $\pm$ 0.048	0.007 $\pm$ 0.001	<b>0.450<math>\pm</math>0.089</b>	-1.454 $\pm$ 0.074
	$\beta/\Phi$	KL	<b>-0.052<math>\pm</math>0.078</b>	0.006 $\pm$ 0.003	0.386 $\pm$ 0.187	<b>-0.595<math>\pm</math>0.044</b>
		LV	*	*	*	*
	Ancestral/ $\sigma$	Init	-	-0.077 $\pm$ 0.072	0.009 $\pm$ 0.001	0.357 $\pm$ 0.084
$\beta$		KL	-0.071 $\pm$ 0.088	0.007 $\pm$ 0.003	0.378 $\pm$ 0.172	<b>-0.615<math>\pm</math>0.027</b>
		LV	4.200 $\pm$ 2.131	0.009 $\pm$ 0.001	0.006 $\pm$ 0.010	-2.352 $\pm$ 0.104
$\Phi$		KL	-0.079 $\pm$ 0.054	0.007 $\pm$ 0.002	<b>0.427<math>\pm</math>0.104</b>	-1.443 $\pm$ 0.078
		LV	5.332 $\pm$ 2.923	0.011 $\pm$ 0.001	0.006 $\pm$ 0.015	-3.549 $\pm$ 0.146
$\beta/\Phi$		KL	<b>-0.057<math>\pm</math>0.064</b>	<b>0.007<math>\pm</math>0.004</b>	0.413 $\pm$ 0.178	-0.637 $\pm$ 0.052
		LV	8.901 $\pm$ 4.396	0.009 $\pm$ 0.001	0.003 $\pm$ 0.001	-2.729 $\pm$ 0.124
Langevin		Init	-	0.001 $\pm$ 0.050	0.013 $\pm$ 0.001	0.345 $\pm$ 0.015
	$\beta$	KL	-0.002 $\pm$ 0.042	0.007 $\pm$ 0.002	0.515 $\pm$ 0.077	-1.523 $\pm$ 0.076
		LV	0.005 $\pm$ 0.053	0.007 $\pm$ 0.002	0.506 $\pm$ 0.097	-1.496 $\pm$ 0.069
	$\Phi$	KL	-0.008 $\pm$ 0.064	0.009 $\pm$ 0.001	0.364 $\pm$ 0.056	-2.296 $\pm$ 0.113
		LV	-0.017 $\pm$ 0.049	0.014 $\pm$ 0.002	0.390 $\pm$ 0.019	-3.164 $\pm$ 0.149
	$\beta/\Phi$	KL	-0.001 $\pm$ 0.042	0.007 $\pm$ 0.002	0.516 $\pm$ 0.080	-1.521 $\pm$ 0.077
		LV	0.008 $\pm$ 0.084	<b>0.007<math>\pm</math>0.002</b>	0.485 $\pm$ 0.140	-1.523 $\pm$ 0.073
	$\Phi/\Delta$	KL	-0.004 $\pm$ 0.059	0.011 $\pm$ 0.001	0.397 $\pm$ 0.066	-3.155 $\pm$ 0.171
		LV	0.005 $\pm$ 0.077	0.011 $\pm$ 0.001	0.346 $\pm$ 0.110	-3.417 $\pm$ 0.149
	$\beta/\Phi/\Delta$	KL	0.004 $\pm$ 0.044	0.013 $\pm$ 0.005	<b>0.553<math>\pm</math>0.100</b>	<b>-1.141<math>\pm</math>0.070</b>
LV		<b>0.000<math>\pm</math>0.071</b>	0.008 $\pm$ 0.003	0.457 $\pm$ 0.152	-1.694 $\pm$ 0.074	

Table 3.2 Same as Table 3.1 but for the GMM ( $d = 1$ ) benchmark problem using 8 discretisation steps.

3.4.4, this target is particularly challenging due to the widely varying magnitude of the target score, as the conditional marginals for  $x_{2:10}$  are approximately Dirac for  $x_1 \ll 0$ . However, the generative SDE is only simulated until step  $1 - \varepsilon$ , with  $\varepsilon > 0$ , and, although the sampling trajectory is optimised (thus allowing to use an arbitrarily small  $\varepsilon$ , up to the machine precision), some residual noise can still be present. This results in extremely values for the mean log-likelihood on initialisation, as the samples have marginal conditionals  $x_{2:10}$  for with non-zero variance at the noise level  $1 - \varepsilon$ , even for  $x_1 \ll 0$ . To tackle this limitation, the

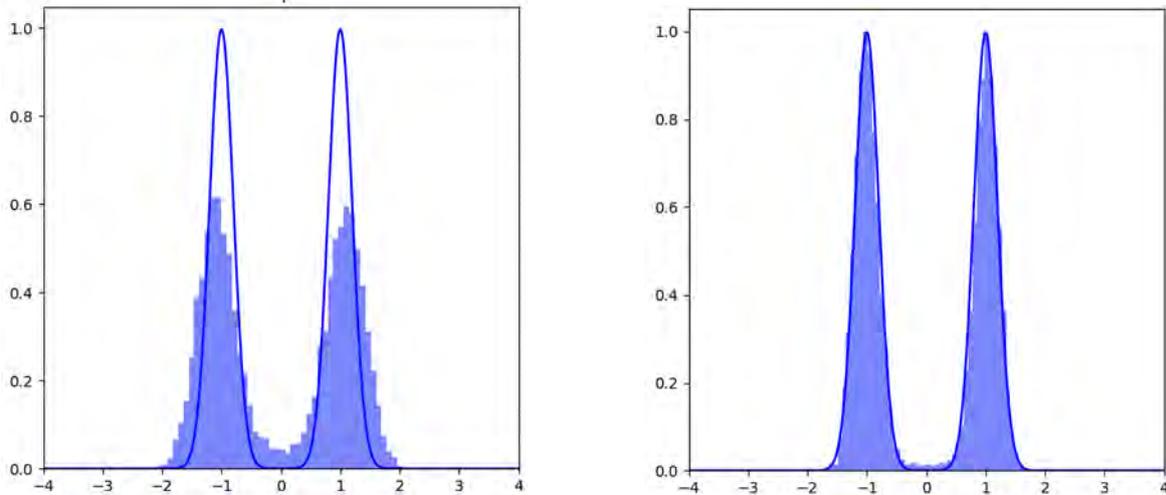


Fig. 3.5 Qualitative comparison of sample quality, in the 1-D GMM benchmark, for the ancestral sampler method, with learnable backward variance, using 8 discretisation steps, before training (left) and after optimisation of the control and discretisation steps (right).

residual noise can be removed by applying Tweedie’s formula (Efron, 2011) to the samples generated by the model, as shown in Figure 3.7. However, we found that using denoised samples in the optimisation led to unstable training, so the performance metrics reported in Table 3.3 correspond to estimations from samples with residual noise, thus explaining the poor performance for this benchmark.

### 3.6.4 Many Well

The 8-dimensional Many Well benchmark is obtained as the product of 4 copies of the 2-dimensional Double Well target, defined in Section 3.4.4. Numerical results for this target, using 32 Euler-Maruyama discretisation steps, are displayed in Table 3.4.

First of all, it is relevant to highlight that the three samplers provide faithful approximations of the normalisation constant (within  $[\log Z - 0.2, \log Z + 0.2]$ ), which is indicative that the  $2^4 = 16$  modes of the target are captured. Moreover, learning the variance of the generative SDE, as done for the Ancestral/ $\sigma$  method, leads to significantly higher effective samples sizes, as compared to other methods. This might be due to the fact that, if the noise added at each step in the simulation of the SDE is too high, the simulated trajectories can reach low density regions of the target with non-zero probability. This phenomenon can result in an IS weights distribution with heavy right tails, thus inflating its variance, and resulting in low effective sample size values.

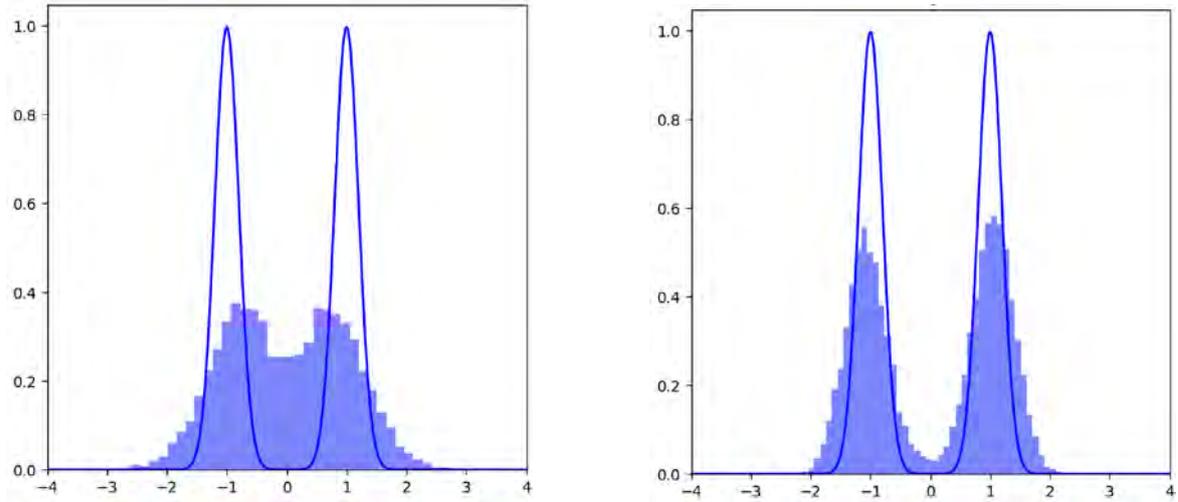


Fig. 3.6 Qualitative comparison of sample quality, in the 1-D GMM benchmark, for the annealed Langevin sampler (variant  $\beta/\Phi/\Delta$ ), using 8 discretisation steps, before training (left) and after optimisation of the control and discretisation steps (right).

Sampler	Variant	Loss	$\log Z$	$\mathcal{W}_2^Y \downarrow$	ESS $\uparrow$	LL $\uparrow$
Target	-	-	0.000	0.009 $\pm$ 0.001	-	32.161 $\pm$ 0.693
Ancestral	Init	-	-2.286 $\pm$ 0.986	<b>0.024<math>\pm</math>0.006</b>	0.011 $\pm$ 0.007	-108509.641 $\pm$ 201074.234
	$\beta$	KL	-11.414 $\pm$ 2.497	0.080 $\pm$ 0.022	0.004 $\pm$ 0.002	-1.971 $\pm$ 5.141
		LV	*	*	*	*
	$\Phi$	KL	-2.885 $\pm$ 1.104	0.100 $\pm$ 0.029	0.017 $\pm$ 0.014	-0.600 $\pm$ 1.459
		LV	*	*	*	*
$\beta/\Phi$	KL	<b>-1.770<math>\pm</math>0.650</b>	0.071 $\pm$ 0.026	<b>0.017<math>\pm</math>0.014</b>	<b>7.673<math>\pm</math>0.273</b>	
	LV	*	*	*	*	
Ancestral/ $\sigma$	Init	-	-3.348 $\pm$ 0.777	<b>0.036<math>\pm</math>0.010</b>	0.008 $\pm$ 0.004	-248855.844 $\pm$ 548453.625
	$\beta$	KL	-11.433 $\pm$ 2.496	0.080 $\pm$ 0.022	0.004 $\pm$ 0.002	-1.954 $\pm$ 5.074
		LV	*	*	*	*
	$\Phi$	KL	-3.507 $\pm$ 0.635	0.112 $\pm$ 0.032	0.014 $\pm$ 0.008	-2.443 $\pm$ 1.499
		LV	*	*	*	*
$\beta/\Phi$	KL	<b>-1.899<math>\pm</math>0.768</b>	0.075 $\pm$ 0.027	<b>0.023<math>\pm</math>0.019</b>	<b>6.755<math>\pm</math>0.366</b>	
	LV	*	*	*	*	

Table 3.3 Same as Table 3.1 but for the Funnel ( $d = 10$ ) benchmark problem using 32 discretisation steps.

To qualitatively evaluate the samples generated by the trained model, we follow the approach in Midgley et al. (2022), and visualise a selection of two-dimensional marginals. Specifically, we consider two inputs coming from different Double Well factors, while

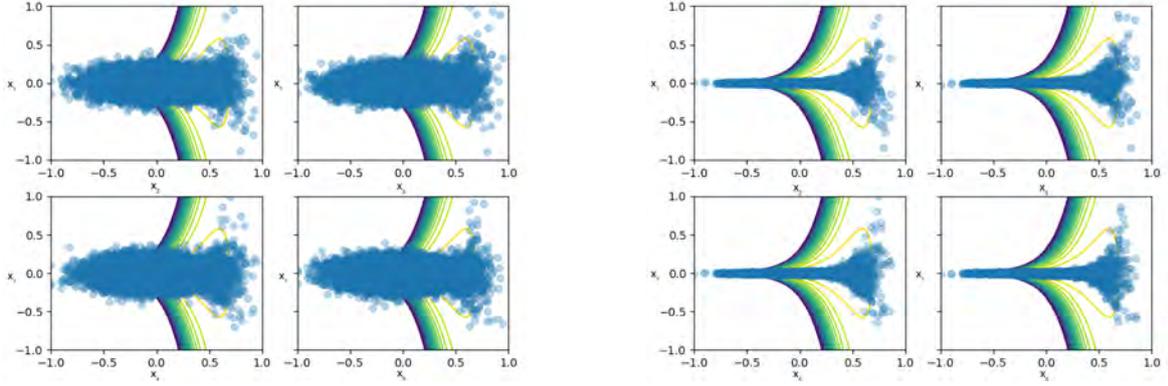


Fig. 3.7 Qualitative comparison of the samples generated by the ancestral sampler, for 32 discretisation steps before (left) and after (right) applying Tweedie's formula. This target is 10-dimensional, and only the marginals  $\{(x_1, x_i)\}_{i=2}^5$  are shown.

fixing the other variables to zero. This is shown in Figure 3.8, which evidences that, before training (left), the variance of the target is significantly overestimated, while after training (right) the samples are grouped more compactly around each mode, thus the higher mean log-likelihoods.

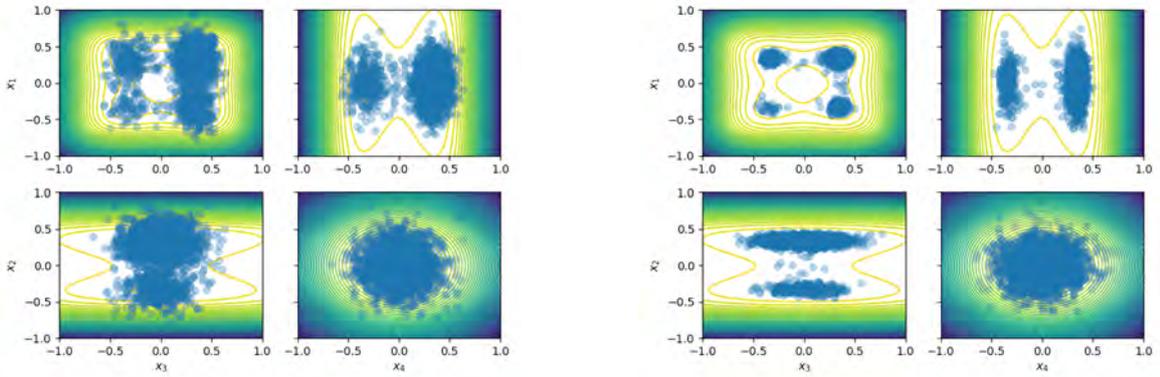


Fig. 3.8 Qualitative comparison of sample quality, in the Many Well benchmark, for the ancestral sampler method, using 32 discretisation steps, before training (left) and after optimisation of the control and discretisation steps (right).

### 3.7 Analysis

Following, we conduct additional analysis to more detailed greater insights into the behaviour of the proposed samplers.

Sampler	Variant	Loss	log Z	$\mathcal{W}_2^\gamma \downarrow$	ESS $\uparrow$	LL $\uparrow$
Target	-	-	41.174	0.037±0.003	-	47.125±0.105
Ancestral	Init	-	40.725±0.491	0.041±0.003	0.028±0.018	39.392±0.313
	$\beta$	KL	<b>41.135±0.279</b>	0.039±0.004	0.052±0.038	46.009±0.123
		LV	41.070±0.736	<b>0.039±0.003</b>	0.031±0.023	44.647±0.149
	$\Phi$	KL	40.876±0.424	0.075±0.007	0.038±0.022	42.406±0.240
		LV	28.920±1.063	0.104±0.008	0.013±0.010	36.964±0.295
	$\beta/\Phi$	KL	41.103±0.187	0.045±0.004	<b>0.089±0.037</b>	<b>46.825±0.108</b>
		LV	40.496±0.627	0.066±0.006	0.025±0.016	45.949±0.141
	Ancestral/ $\sigma$	Init	-	40.632±0.538	0.051±0.003	0.014±0.006
$\beta$		KL	<b>41.157±0.093</b>	<b>0.039±0.003</b>	0.219±0.088	46.434±0.103
		LV	62.432±7.193	0.056±0.004	0.003±0.001	34.689±0.451
$\Phi$		KL	40.926±0.295	0.070±0.006	0.054±0.033	41.623±0.229
		LV	47.161±4.531	0.082±0.006	0.003±0.002	22.143±0.616
$\beta/\Phi$		KL	41.144±0.091	0.043±0.004	<b>0.250±0.112</b>	<b>46.894±0.112</b>
		LV	57.533±4.543	0.058±0.004	0.003±0.001	32.404±0.430
Langevin		Init	-	27.819±6.399	0.174±0.014	0.002±0.000
	$\beta$	KL	36.141±2.541	0.102±0.007	0.005±0.002	-87.670±9.623
		LV	36.187±2.460	0.102±0.007	0.005±0.003	-87.643±9.794
	$\Phi$	KL	35.820±2.647	0.110±0.008	0.004±0.002	-18.134±5.563
		LV	-37945.602±126.600	0.203±0.014	0.002±0.000	-1791.432±75.125
	$\beta/\Phi$	KL	36.897±2.435	0.110±0.008	0.004±0.002	-11.876±4.046
		LV	30.576±3.284	0.111±0.008	0.004±0.002	-48.124±8.107
	$\Phi/\Delta$	KL	39.622±1.282	0.065±0.004	0.008±0.005	28.326±0.635
		LV	20.624±6.826	0.166±0.019	0.002±0.000	-5833.650±419.485
	$\beta/\Phi/\Delta$	KL	<b>40.913±0.538</b>	<b>0.060±0.005</b>	<b>0.023±0.016</b>	<b>40.833±0.237</b>
LV		*	*	*	*	

Table 3.4 Same as Table 3.1 but for the Many Well ( $d = 8$ ) benchmark problem using 32 discretisation steps.

### 3.7.1 Importance of Control Tying

The controls for both the Controlled Ancestral and Controlled Annealed Langevin samplers are defined so that the Problem 2.5 has a unique solution. To do so, either  $u$  or  $v$  is parameterised by a neural network, so it can evolve unconstrained during optimisation, while the other is defined in terms of the free control. For the presented approaches, the specific choice of control tying was motivated from Nelson’s identity (Nelson, 2020)

$$u^* + v^* = g^T \nabla_x \log \overleftarrow{p}_{Y^*}, \quad (3.50)$$

where  $\nabla_x \log \overleftarrow{p}_{Y^*}$  is replaced by an approximation to the ground-truth score. From a high-level perspective, it seems intuitive that relying on independent controls might lead to improved performance, as the inference and generative SDEs would be allowed to evolve unconstrained during optimisation. In this regard, even if the solution to Problem 2.5 is no longer unique, it might still be possible for the optimiser to converge to an optimum whose performance surpasses that of the unique solution when the controls are tied.

However, we empirically verified that untying the controls can result in convergence to pathological solutions. This is illustrated in Figure 3.9, which shows the samples generated when training the controls  $u$  and  $v$  independently for the Controlled Ancestral sampler. Despite the samples collapsing to a single region, and not matching the probability contours at all, the training loss decreased steadily during optimisation, thus indicating that this anomalous behaviour is not due to any divergence during training, but instead, to the ill-defined nature of the problem.

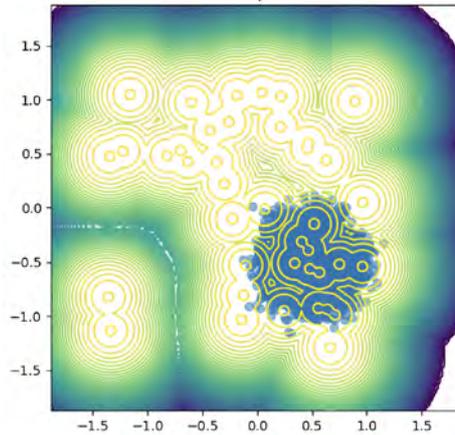


Fig. 3.9 Samples generated, for the 2-D GMM benchmark, after training the Controlled Ancestral Sampler with independent controls.

### 3.7.2 Inaccurate Score Estimation

The score models used in this evaluation were trained with ground-truth samples from the target distribution. However, it is interesting to consider what happens when these are trained, instead, in biased samples from the target distribution. In this scenario, the controls need to be adapted during optimisation to correct the error introduced by the biased score-model.

To this end, we trained the diffusion model for the 1-D GMM benchmark with data from a GMM with the same means and covariances, but with prior probabilities  $(0.9, 0.1)$ , instead of  $(0.5, 0.5)$ . Consequently, the samples generated by the model are highly biased on initialisation, as illustrated in Figure 3.10.

However, the optimisation of the controls can lead to a slight improvement in sample quality, as shown in Figure 3.10 (right), thus indicating that the controls can be adapted to compensate, to some extent, the error in the score estimation. Empirically, we found that training dynamics were less stable, compared to when using a model trained in ground-truth samples, so we hypothesise that sample quality can be further improved by tuning the hyperparameters.

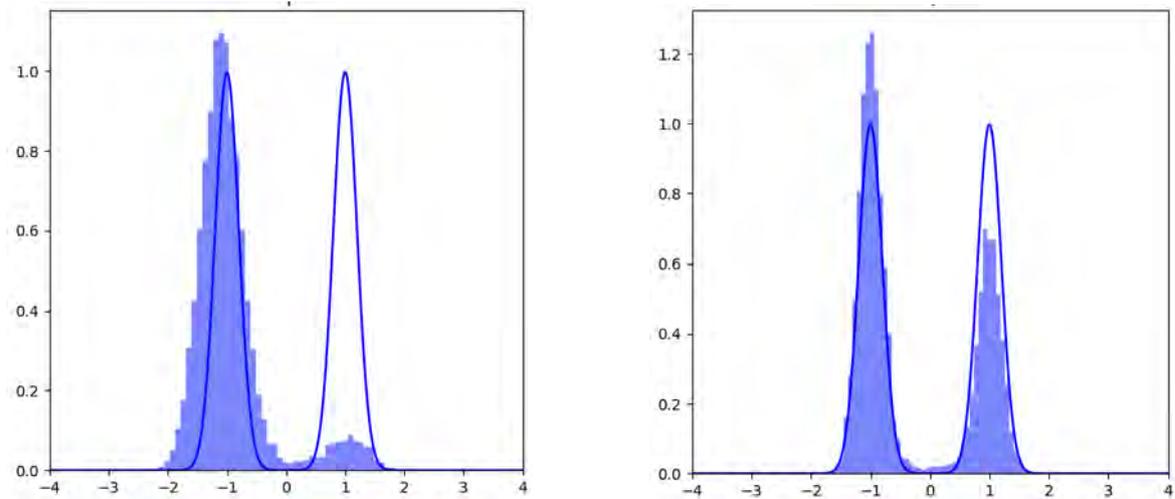


Fig. 3.10 Qualitative comparison of sample quality for the ancestral sampler method, when using an inaccurate score-model, before training (left) and after optimisation of the control and discretisation steps (right).

### 3.7.3 Effective Sample Sizes

The numerical results presented evidence that, despite the improvements in sample quality obtained by optimising the controls, the effective sample sizes are generally low, specially for the 2-GMM and Funnel benchmarks. However, it is important to highlight that our objective is the KL divergence, and, although at optimality the IS weights have zero variance, the optimisation of the KL loss is not equivalent to the minimisation of the importance sampling weights variance (Midgley et al., 2022). Therefore, it can be expected that, if the controls are not optimal, the variance of the IS weights might still be high, thus being reflected in low effective sample sizes.

To reduce the variance of the importance sampling weights after training the model, a non-linear transformation can be applied to them, such as truncation of their values to a given upper bound (Ionides, 2008). Although unbiasedness can no longer be guaranteed, estimators

relying on the truncated IS weights are still consistent. To illustrate the effect of post-training IS variance reduction techniques, we apply the non-linear importance sampling method, presented in Míguez (2017), to the IS weights obtained, after training, for the 2-GMM benchmark, as shown in Figure 3.11. Note that, due to the low effective sample size, the histogram obtained by weighting the samples using the non-transformed IS weights collapses into multiple Dirac distributions (bottom-left), while after transforming the IS weights, the resulting histogram (bottom-right) approximates better the ground-truth histogram (top-left), as the samples in low-probability regions are down-weighted correspondingly.

### 3.7.4 Varying Number of Discretisation Steps

Although the numerical results shown in Tables 3.2 (GMM,  $d=1$ ), 3.1 (GMM,  $d=2$ ), 3.3 (Funnel,  $d=10$ ) and 3.4 (Many Well,  $d=8$ ) consider a fixed number of discretisation steps, we conducted additional experiments, considering varying number of steps sizes.

These experiments are summarised in Figures A.2, A.3, A.4 and A.5. As it was expected, the performance metrics improve as the number of discretisation steps increases, due to reduction of numerical errors in the simulation of the generative SDE. Moreover, it can be seen that, when relying on few steps, optimising the controls and discretisation points is essential to obtain good sample quality.

## 3.8 Summary

In this chapter, the Controlled Ancestral sampler and the Controlled Annealed Langevin sampler were proposed. These approaches aim to improve sample quality, for a pre-trained diffusion model, by optimising relevant controls in the inference and generative SDEs. While the Controlled Ancestral Sampler generally achieves improved sample quality, as compared to the initial performance of the model, the Controlled Annealed Langevin requires careful tuning of its hyperparameters to compete with the Controlled Ancestral sampler.

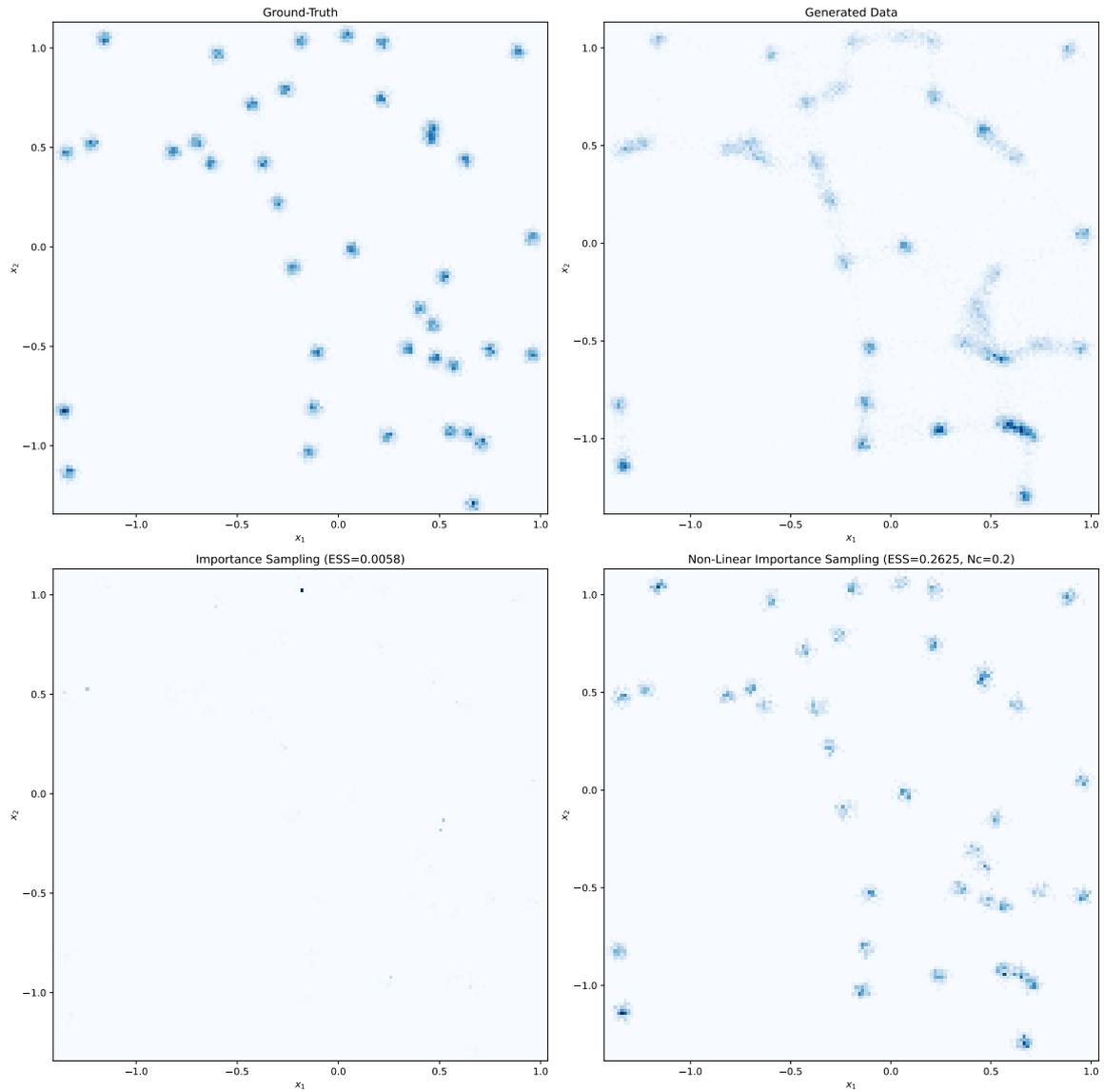


Fig. 3.11 Histograms obtained from ground-truth data of the 2-D GMM benchmark (top-left), non-weighted samples from the trained model (top-right), IS-weighted samples from the model (bottom-left), and non-linear IS-weighted (Míguez, 2017) samples (bottom-right).

<b>Diffusion SDE</b>	
SDE	Variance-Preserving SDE (Song and Ermon, 2019)
log-SNR sched. $\gamma(t)$	$-\log(\exp(1e^{-4} + 10t^2) - 1)$
$\alpha(t)$	$\text{sigmoid}(\gamma(t))^{\frac{1}{2}}$
$\sigma^2(t)$	$\text{sigmoid}(-\gamma(t))$
prior distribution $p_{\text{prior}}$	$\mathcal{N}(0, I)$
<b>Ancestral SDE</b>	
drift $f(x, t)$ , diff. $g(t)$	Variance-Preserving SDE (Song and Ermon, 2019)
inference control $g \cdot u$	$-\Phi_\omega$
generative control	$g^2 s_\theta + \Phi_\omega$
prior distribution $p_{\text{prior}}$	$\mathcal{N}(0, I)$
<b>Annealed SDE</b>	
drift $f(x, t)$ , diff. $g(t)$	$\sigma^2 s_\theta(x, t), \sigma\sqrt{2}$
inference control $g \cdot u$	$-\Phi_\omega$
generative control	$2\sigma^2 s_\theta + \Phi_\omega$
prior distribution $p_{\text{prior}}$	$\mathcal{N}(0, I)$
<b>SDE Solver</b>	
type	Euler-Maruyama (Kloeden et al., 1992)
discretisation steps $T$	[16, 32, 64]
<b>Score Training</b>	
loss	unweighted VLB $\ \varepsilon - \varepsilon_\theta\ $
optimiser	Adam
learning rate	0.0003
batch size	2000
iterations	30000
gradient clipping	5 ( $\ell^2$ -norm)
EMA momentum	0.999
<b>SDE Training</b>	
loss	[Kullback–Leibler divergence $D_{KL}$ , Log-variance divergence $D_{LV}$ ]
optimiser	Adam
learning rate	0.005
batch size	300 (1D GMM), 2000
iterations	3000
gradient clipping	5 ( $\ell^2$ -norm)
<b>Network <math>s_\theta</math></b>	
time encoding	fourier-emb
network input	$X_t$ , fourier-emb, fourier-feat
activation $\rho$	GeLU (Hendrycks and Gimpel, 2016)
num. hiddens $n$	6
embedding dim.	128
architecture	$\text{linear} \circ (\text{res} + \rho \circ \text{linear})^n \circ \text{linear}$
initialization	weights and biases to 0 (last layer), default (rest)
<b>Network <math>\Phi_\omega</math></b>	
network input	$X_t$ , fourier-emb
time encoding	$\text{linear} \circ \rho \circ \text{linear} \circ [\text{fourier-emb}]$
activation $\rho$	GeLU (Hendrycks and Gimpel, 2016)
num. hiddens $n$	2
embedding dim.	[64, 128, 64]
architecture	$\text{linear} \circ (\rho \circ \text{linear})^n$
initialization	weights and biases to 0 (last layer), default (rest)
<b>Evaluation</b>	
num. seeds	30
samples per seed	500

Table 3.5 Hyperparameter configuration for the numerical results presented in Section 3.6.

# Chapter 4

## Conclusions

In this work, two methods are proposed to improve sample quality for a pre-trained diffusion model when an unnormalised density is available: the Controlled Ancestral sampler and the Controlled Annealed Langevin sampler. These are defined in terms of controlled stochastic differential equations, defining transports from a target to a prior distribution, and their corresponding time-reversals. By optimising the controls to minimise the divergence between the path measures induced by these SDEs, it is empirically demonstrated that sample quality can be improved for several benchmark targets.

### 4.1 Limitations

An important limitation of this work is that the evaluation is conducted in relatively low-dimensional targets, and, although theoretically the generative SDE inherits the fast-mixing properties of the inference SDE (Vargas et al., 2023), evaluation in higher-dimensional targets should be conducted to further prove the ability of the proposed methods to improve sample quality. Note that this requires training a diffusion model with high-capacity, and simulating the generative SDE using a sufficiently high number of discretisation steps, so tracking the computational graph can become prohibitively expensive, and adjoint SDE solvers are required. As the Euler-Maruyama scheme is used in this work for the SDE simulation, the experimental infrastructure needs to be adapted to handle more challenging targets, although the theoretical framework still holds its validity.

Another challenge is the discovery of high-probability regions that are missed by the pre-trained diffusion model. Although the stochasticity in the SDE simulation can enable the discovery of these regions, this exploration is likely to become inefficient in high-dimensions. In this setting, on-policy optimisation of the control, using an objective such as the Kullback-Leibler divergence, is likely to result in convergence to a local minima which possibly misses

undiscovered modes. Consequently, relying on the target score is fundamental to efficiently guide the exploration and facilitate the discovery of missing modes. However, in the present work, gradient information of the target is not exploited.

## 4.2 Future Works

Besides addressing the limitations outlined previously, potential research directions involve enhancing exploration using off-policy divergences, incorporating learning signals at intermediate steps in the simulation trajectory and explore alternative divergences.

Regarding the first point, it was highlighted previously that, when relying on on-policy training, exploration is only encouraged by the stochasticity of the SDE. Alternatively, an exploratory reference process could be considered, e.g. a noisier variant of the generative SDE, and optimisation of the control done in an off-policy fashion, thus facilitating the discovery of new high-density regions. To this end, the KL divergence is not readily applicable, and an alternative optimisation objective needs to be considered, such as the log-variance divergence. On the other hand, the reference process could leverage gradient guidance to enable efficient exploration, even in high dimensions.

Moreover, for the proposed methods, the learning signal is provided after simulating the complete trajectory, when the unnormalised density is evaluated. Alternatively, intermediate learning signals could be incorporated, which can be beneficial to reduce gradient variance. Zhang et al. (2023a) explore this possibility, and rely on a variance-exploding stochastic process with a Dirac distribution as initial boundary condition. Consequently, it can be interesting to adapt this method to reduce gradient variance for the sampling approaches presented in this dissertation.

Finally, alternative divergences could be used to define the optimisation objective. For instance, the 2-divergence in Midgley et al. (2022) enables minimisation of the importance weights variance. Consequently, this divergence could be incorporated to possibly improve the effective sample sizes.

# References

- Akhound-Sadegh, T., Rector-Brooks, J., Bose, A. J., Mittal, S., Lemos, P., Liu, C.-H., Sendera, M., Ravanbakhsh, S., Gidel, G., Bengio, Y., Malkin, N., and Tong, A. (2024). Iterated denoising energy matching for sampling from boltzmann densities.
- Anderson, B. D. (1982). Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12(3):313–326.
- Bao, F., Li, C., Zhu, J., and Zhang, B. (2022). Analytic-dpm: an analytic estimate of the optimal reverse variance in diffusion probabilistic models. *arXiv preprint arXiv:2201.06503*.
- Berner, J., Richter, L., and Ullrich, K. (2022). An optimal control perspective on diffusion-based generative modeling. *arXiv preprint arXiv:2211.01364*.
- Cao, Y., Chen, J., Luo, Y., and Zhou, X. (2024). Exploring the optimal choice for generative processes in diffusion models: Ordinary vs stochastic differential equations. *Advances in Neural Information Processing Systems*, 36.
- Chen, H., Lee, H., and Lu, J. (2023). Improved analysis of score-based generative modeling: User-friendly bounds under minimal smoothness assumptions. In *International Conference on Machine Learning*, pages 4735–4763. PMLR.
- Chen, N., Zhang, Y., Zen, H., Weiss, R. J., Norouzi, M., and Chan, W. (2020). Wavegrad: Estimating gradients for waveform generation. *arXiv preprint arXiv:2009.00713*.
- Chen, T., Liu, G.-H., and Theodorou, E. A. (2021). Likelihood training of schrödinger bridge using forward-backward sdes theory. *arXiv preprint arXiv:2110.11291*.
- Cuturi, M. (2013). Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26.
- De Bortoli, V. (2022). Convergence of denoising diffusion models under the manifold hypothesis. *arXiv preprint arXiv:2208.05314*.
- De Bortoli, V., Hutchinson, M., Wirnsberger, P., and Doucet, A. (2024). Target score matching. *arXiv preprint arXiv:2402.08667*.
- De Bortoli, V., Thornton, J., Heng, J., and Doucet, A. (2021). Diffusion schrödinger bridge with applications to score-based generative modeling. *Advances in Neural Information Processing Systems*, 34:17695–17709.
- Del Moral, P., Doucet, A., and Jasra, A. (2006). Sequential monte carlo samplers. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 68(3):411–436.

- Desjardins, G., Courville, A., Bengio, Y., Vincent, P., and Delalleau, O. (2010). Tempered markov chain monte carlo for training of restricted boltzmann machines. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 145–152. JMLR Workshop and Conference Proceedings.
- Doucet, A., Grathwohl, W., Matthews, A. G., and Strathmann, H. (2022). Score-based diffusion meets annealed importance sampling. *Advances in Neural Information Processing Systems*, 35:21482–21494.
- Efron, B. (2011). Tweedie’s formula and selection bias. *Journal of the American Statistical Association*, 106(496):1602–1614.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2020). Generative adversarial networks. *Communications of the ACM*, 63(11):139–144.
- Hartmann, C. and Richter, L. (2024). Nonasymptotic bounds for suboptimal importance sampling. *SIAM/ASA Journal on Uncertainty Quantification*, 12(2):309–346.
- Hendrycks, D. and Gimpel, K. (2016). Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*.
- Hess, T., Tuytelaars, T., and van de Ven, G. M. (2023). Two complementary perspectives to continual learning: Ask not only what to optimize, but also how. *arXiv preprint arXiv:2311.04898*.
- Ho, J., Jain, A., and Abbeel, P. (2020). Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851.
- Ho, J., Salimans, T., Gritsenko, A., Chan, W., Norouzi, M., and Fleet, D. J. (2022). Video diffusion models. *Advances in Neural Information Processing Systems*, 35:8633–8646.
- Hoogeboom, E., Satorras, V. G., Vignac, C., and Welling, M. (2022). Equivariant diffusion for molecule generation in 3d. In *International conference on machine learning*, pages 8867–8887. PMLR.
- Ionides, E. L. (2008). Truncated importance sampling. *Journal of Computational and Graphical Statistics*, 17(2):295–311.
- Karras, T., Aittala, M., Aila, T., and Laine, S. (2022). Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577.
- Kidger, P., Foster, J., Li, X. C., and Lyons, T. (2021). Efficient and accurate gradients for neural sdes. *Advances in Neural Information Processing Systems*, 34:18747–18761.
- Kingma, D., Salimans, T., Poole, B., and Ho, J. (2021). Variational diffusion models. *Advances in neural information processing systems*, 34:21696–21707.
- Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

- Kingma, D. P. and Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.
- Kloeden, P. E., Platen, E., Kloeden, P. E., and Platen, E. (1992). *Stochastic differential equations*. Springer.
- Léonard, C. (2013). A survey of the schrödinger problem and some of its connections with optimal transport. *arXiv preprint arXiv:1308.0215*.
- Midgley, L. I., Stimper, V., Simm, G. N., Schölkopf, B., and Hernández-Lobato, J. M. (2022). Flow annealed importance sampling bootstrap. *arXiv preprint arXiv:2208.01893*.
- Míguez, J. (2017). On the performance of nonlinear importance samplers and population monte carlo schemes. In *2017 22nd International Conference on Digital Signal Processing (DSP)*, pages 1–5. IEEE.
- Minka, T. et al. (2005). Divergence measures and message passing. Technical report, Technical report, Microsoft Research.
- Neal, R. M. (1993). Probabilistic inference using markov chain monte carlo methods.
- Neal, R. M. (2001). Annealed importance sampling. *Statistics and computing*, 11:125–139.
- Neal, R. M. (2003). Slice sampling. *The annals of statistics*, 31(3):705–767.
- Nelson, E. (2020). *Dynamical theories of Brownian motion*, volume 101. Princeton university press.
- Nichol, A. Q. and Dhariwal, P. (2021). Improved denoising diffusion probabilistic models. In *International conference on machine learning*, pages 8162–8171. PMLR.
- Noé, F., Olsson, S., Köhler, J., and Wu, H. (2019). Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning. *Science*, 365(6457):eaaw1147.
- Ott, M., Edunov, S., Baevski, A., Fan, A., Gross, S., Ng, N., Grangier, D., and Auli, M. (2019). fairseq: A fast, extensible toolkit for sequence modeling. *arXiv preprint arXiv:1904.01038*.
- Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., and Sutskever, I. (2021). Zero-shot text-to-image generation. In *International conference on machine learning*, pages 8821–8831. Pmlr.
- Rezende, D. and Mohamed, S. (2015). Variational inference with normalizing flows. In *International conference on machine learning*, pages 1530–1538. PMLR.
- Richter, L. and Berner, J. (2023). Improved sampling via learned diffusions. *arXiv preprint arXiv:2307.01198*.
- Roberts, G. O. and Tweedie, R. L. (1996). Exponential convergence of langevin distributions and their discrete approximations.
- Roeder, G., Wu, Y., and Duvenaud, D. K. (2017). Sticking the landing: Simple, lower-variance gradient estimators for variational inference. *Advances in Neural Information Processing Systems*, 30.

- Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695.
- Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E. L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al. (2022). Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494.
- Song, Y. and Ermon, S. (2019). Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32.
- Song, Y. and Ermon, S. (2020). Improved techniques for training score-based generative models. *Advances in neural information processing systems*, 33:12438–12448.
- Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. (2020). Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*.
- Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. (2021). Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*.
- Thin, A., Kotelevskii, N., Doucet, A., Durmus, A., Moulines, E., and Panov, M. (2021). Monte carlo variational auto-encoders. In *International Conference on Machine Learning*, pages 10247–10257. PMLR.
- Turner, R. E., Diaconu, C.-D., Markou, S., Shysheya, A., Foong, A. Y., and Mlodozieniec, B. (2024). Denoising diffusion probabilistic models in six simple steps. *arXiv preprint arXiv:2402.04384*.
- Üstünel, A. S. and Zakai, M. (2013). *Transformation of measure on Wiener space*. Springer Science & Business Media.
- Vargas, F., Grathwohl, W., and Doucet, A. (2023). Denoising diffusion samplers. *arXiv preprint arXiv:2302.13834*.
- Vargas, F., Padhy, S., Blessing, D., and Nüsken, N. (2024). Transport meets variational inference: Controlled monte carlo diffusions. In *The Twelfth International Conference on Learning Representations*.
- Villani, C. et al. (2009). *Optimal transport: old and new*, volume 338. Springer.
- Vincent, P. (2011). A connection between score matching and denoising autoencoders. *Neural computation*, 23(7):1661–1674.
- Watson, D., Ho, J., Norouzi, M., and Chan, W. (2021). Learning to efficiently sample from diffusion probabilistic models. *arXiv preprint arXiv:2106.03802*.
- Watson, J. L., Juergens, D., Bennett, N. R., Trippe, B. L., Yim, J., Eisenach, H. E., Ahern, W., Borst, A. J., Ragotte, R. J., Milles, L. F., et al. (2022). Broadly applicable and accurate protein design by integrating structure prediction networks and diffusion generative models. *BioRxiv*, pages 2022–12.

- Wu, H., Köhler, J., and Noé, F. (2020). Stochastic normalizing flows. *Advances in Neural Information Processing Systems*, 33:5933–5944.
- Wu, K. E., Yang, K. K., van den Berg, R., Alamdari, S., Zou, J. Y., Lu, A. X., and Amini, A. P. (2024). Protein structure generation via folding diffusion. *Nature communications*, 15(1):1059.
- Yim, J., Trippe, B. L., De Bortoli, V., Mathieu, E., Doucet, A., Barzilay, R., and Jaakkola, T. (2023). Se (3) diffusion model with application to protein backbone generation. *arXiv preprint arXiv:2302.02277*.
- Zhang, D., Chen, R. T. Q., Liu, C.-H., Courville, A., and Bengio, Y. (2023a). Diffusion generative flow samplers: Improving learning signals through partial trajectory optimization. *arXiv preprint arXiv:2310.02679*.
- Zhang, H., Li, S., Zhang, J., Wang, Z., Wang, J., Jiang, D., Bian, Z., Zhang, Y., Deng, Y., Song, J., et al. (2023b). Sdegen: learning to evolve molecular conformations from thermodynamic noise for conformation generation. *Chemical Science*, 14(6):1557–1568.
- Zhang, Q. and Chen, Y. (2021). Path integral sampler: a stochastic control approach for sampling. *arXiv preprint arXiv:2111.15141*.
- Zhang, Q. and Chen, Y. (2022). Fast sampling of diffusion models with exponential integrator. *arXiv preprint arXiv:2204.13902*.



# **Appendix A**

## **Additional Figures**

In this Appendix, the figures that were omitted from the main text, for the sake of conciseness, are included.

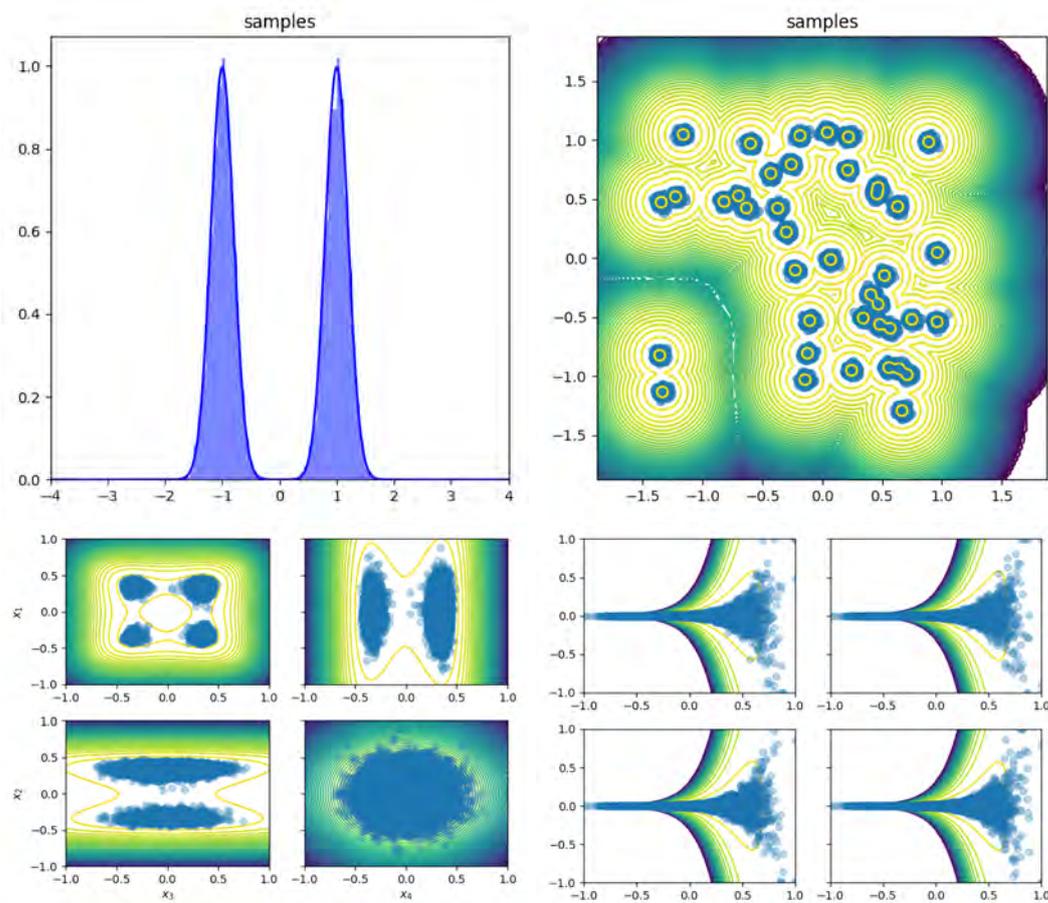


Fig. A.1 Densities, along with ground-truth samples for the benchmarks 1-D GMM (top-left), 2-D GMM (top-right), Many Well (bottom-left) and Funnel (bottom-right). For targets with  $d > 2$ , a subset of marginals are shown.

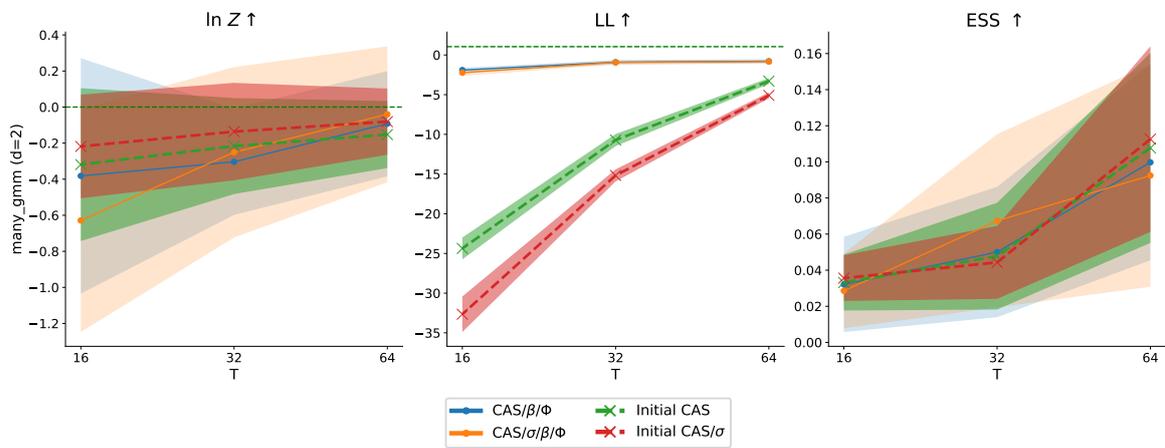


Fig. A.2 Evolution of the performance metrics with increasing number of discretisation steps, for the 2-D GMM benchmark and using the Controlled Ancestral sampler.

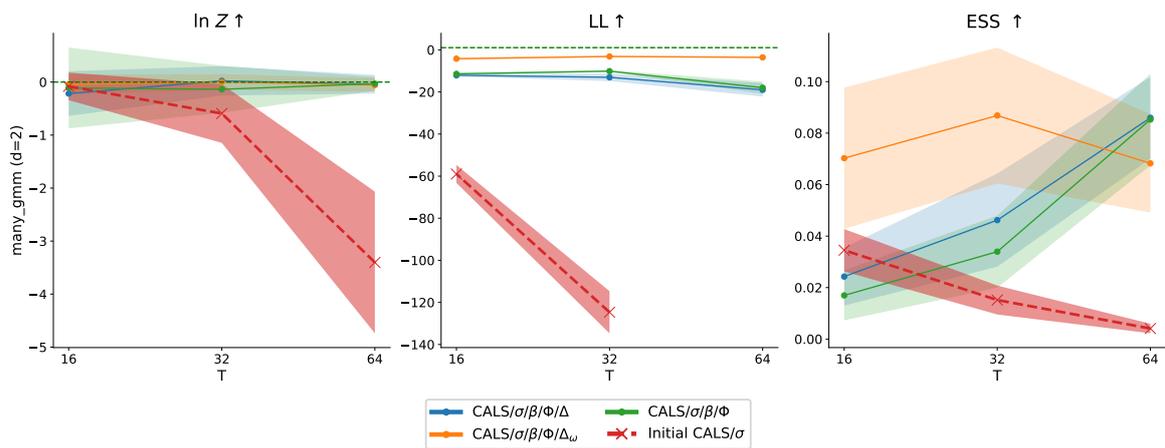


Fig. A.3 Evolution of the performance metrics with increasing number of discretisation steps, for the 2-D GMM benchmark and using the Controlled Annealed Langevin sampler.

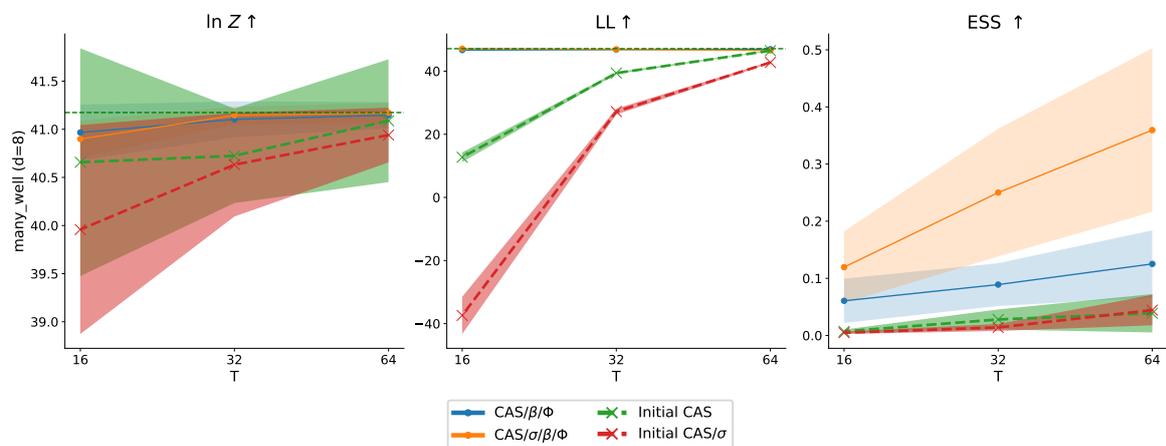


Fig. A.4 Evolution of the performance metrics with increasing number of discretisation steps, for the Many Well benchmark and using the Controlled Ancestral sampler.

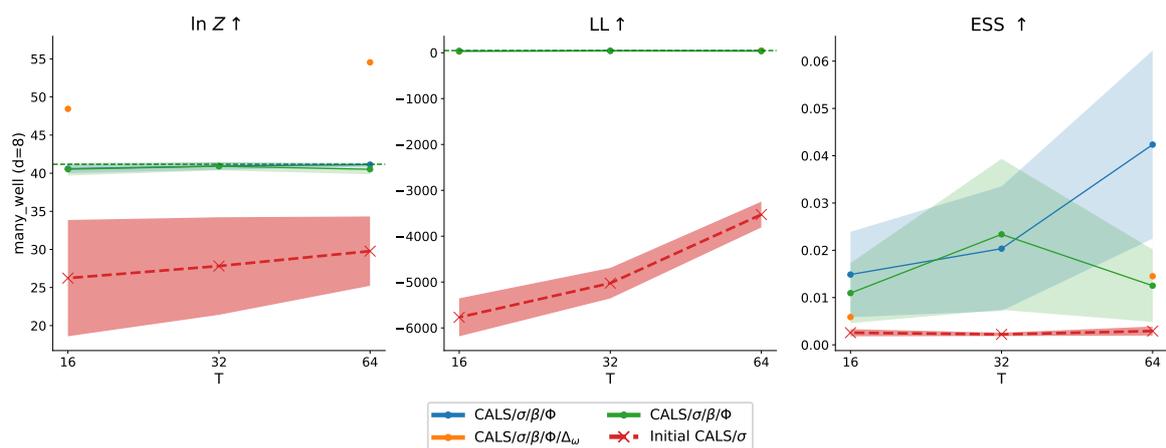


Fig. A.5 Evolution of the performance metrics with increasing number of discretisation steps, for the Many Well benchmark and using the Controlled Annealed Langevin sampler.