LEARNING DIFFUSION BRIDGES ON CONSTRAINED DOMAINS

Xingchao Liu, Lemeng Wu, Mao Ye, Qiang Liu
Department of Computer Science
University of Texas at Austin
{xcliu, lmwu, my21, lqiang}@cs.utexas.edu

ABSTRACT

Diffusion models have achieved promising results on generative learning recently. However, because diffusion processes are most naturally applied on the unconstrained Euclidean space $\mathbb{R}^d$, key challenges arise for developing diffusion based models for learning data on constrained and structured domains. We present a simple and unified framework to achieve this that can be easily adopted to various types of domains, including product spaces of any type (be it bounded/unbounded, continuous/discrete, categorical/ordinal, or their mix). In our model, the diffusion process is driven by a drift force that is a sum of two terms: one singular force designed by Doob’s $h$-transform that ensures all outcomes of the process to belong to the desirable domain, and one non-singular neural force field that is trained to make sure the outcome follows the data distribution statistically. Experiments show that our methods perform superbly on generating tabular data, images, semantic segments and 3D point clouds. Code is available at https://github.com/gnobitab/ConstrainedDiffusionBridge.

1 INTRODUCTION

Diffusion-based deep generative models, notably score matching with Langevin dynamics (SMLD) (Song & Ermon, 2019, 2020), denoising diffusion probabilistic models (DDPM) (Ho et al., 2020), and their variants (e.g., Song et al., 2020b,a; Kong & Ping, 2021; Song et al., 2021; Nichol & Dhariwal, 2021), have shown to achieve new state of the art results for image synthesis (Dhariwal & Nichol, 2021; Ramesh et al., 2022; Ho et al., 2022; Liu et al., 2021), audio synthesis (Chen et al., 2020; Kong et al., 2020), point cloud synthesis (Luo & Hu, 2021a,b; Zhou et al., 2021), and many other AI tasks. These methods train a deep neural network to drive as drift force a diffusion process to generate data, and are shown to outperform competitors, mainly GANs and VAEs, on stability and sample diversity (Xiao et al., 2021; Ho et al., 2020; Song et al., 2020b).

However, due to the continuous nature of diffusion processes, the standard approaches are restricted to generating unconstrained continuous data in $\mathbb{R}^d$. For generating data constrained on special structured domains, such as discrete, bound data or mixes of them, special techniques, e.g., dequantization (Uria et al., 2013; Ho et al., 2019) and multinomial diffusion (Hoogeboom et al., 2021; Austin et al., 2021), need to be developed case by case and the results still tend to be unsatisfying despite promising recent advances (Hoogeboom et al., 2021; Austin et al., 2021).

This work proposes a simple and unified framework for learning diffusion models on general constrained domains $\Omega$ embedded in the Euclidean space $\mathbb{R}^d$. The idea is to learn a continuous $\mathbb{R}^d$-valued diffusion process $Z_t$ on time interval $t \in [0, T]$, with a carefully designed force field, such that the final state $Z_T$ guarantees to 1) fall into the desirable domain $\Omega$, and 2) follows the data distribution asymptotically. We achieve both steps by leveraging a key tool in stochastic calculus called Doob’s $h$-transform (Doob, 1984), which provides formula for deriving the diffusion processes whose final states are guaranteed to fall into a specific set or equal a specific value.

Figure 1: An $\Omega$-Bridge on discrete domain $\Omega = \{1, 2, 3, 4\}$. 
Algorithm 1: Learning Diffusion Models on Constrained Domains (a Simple Example)

**Input:** A dataset $\mathcal{D} := \{x^{(i)}\}$ drawn from distribution $\Pi^*$ on a domain $\Omega = \{e_1, e_2, \ldots, e_K\}$.

**Goal:** Learn a diffusion model that terminates at time $T$ to generate samples from $\Pi^*$.

**Learning:** Solve the optimization below with stochastic gradient descent (or other optimizers)

$$ \theta^* = \arg\min_\theta \int_0^T \mathbb{E}_{x \sim \mathcal{D}} \left[ \| f^\theta(Z_t, t) - \nabla Z_t \log \omega^\Omega(x \mid Z_t, t) \|^2 \right] \, dt, \quad (1) $$

where

$$ \omega^\Omega(x \mid z, t) = \frac{\exp \left( - \frac{\| z - z \|^2}{2(T - t)} \right)}{\sum_{e \in \Omega} \exp \left( - \frac{\| z - e \|^2}{2(T - t)} \right)}, \quad Z_t = t \frac{T^2}{T^2 + (1 - t) x_0 + \sqrt{\frac{t(1 - t)}{T}} \xi}, \quad (2) $$

with $x$ drawn from the dataset $\mathcal{D}$, $\xi \sim \mathcal{N}(0, I)$, and $x_0$ any initial point.

**Sampling:** Generate sample $Z_T$ from

$$ dZ_t = \left[ f^\theta(Z_t, t) + \nabla Z_t \log \omega^\Omega(Z_t, t) \right] \, dt + dW_t, \quad Z_0 = x_0. $$

Remark: When the domain $\Omega$ is a manifold (e.g., line or surface) in $\mathbb{R}^d$, simply replace the sum $\sum_{e \in \Omega}$ with the corresponding line or surface (in general Hausdorff) integration $\int_{\Omega}$ on $\Omega$.

Our simple procedure can be applied to any domain $\Omega$ once a properly defined summation (for discrete sets) or integration (for continuous domains) can be evaluated. To give a quick overview on the practical intuition without invoking the mathematical theory, we show in Algorithm 1 a simple instance of the framework when the domain is a discrete set $\Omega = \{e_1, \ldots, e_K\}$. The idea is to set up the diffusion model to have a form of

$$ dZ_t = \left[ f^\theta(Z_t, t) + \nabla_z \psi^\Omega(Z_t, t) \right] \, dt + dW_t, \quad \psi^\Omega(z, t) := \log \sum_{e \in \Omega} \exp \left( - \frac{\| z - e \|^2}{2(T - t)} \right), \quad (3) $$

where the drift is a sum of a non-singular (e.g., bounded) term $f^\theta(z, t)$ which is a trainable neural force field with parameter $\theta$, and a singular term $\nabla_z \psi^\Omega(z, t)$, which drives $Z_t$ towards set $\Omega$ as a gradient ascent on $\psi^\Omega(z, t)$. The $\psi^\Omega(z, t)$ measures the closeness of $z$ to set $\Omega$, as the log-likelihood of a Gaussian mixture model (GMM) centered on the elements in $\Omega$ with variance $T - t$. When $t$ approaches to the terminal time $T$, the variance $T - t$ of the GMM goes to zero, and the magnitude of $\nabla_z \psi^\Omega(z, t)$ grows to infinity, hence ensuring that $Z_T$ must belong to $\Omega$. In particular, note that

$$ \nabla_z \psi^\Omega(z, t) = \sum_{e \in \Omega} \omega^\Omega(e \mid z, t) \frac{e - z}{T - t}, \quad \omega^\Omega(e \mid z, t) = \frac{\exp \left( - \frac{\| z - e \|^2}{2(T - t)} \right)}{\sum_{e \in \Omega} \exp \left( - \frac{\| z - e \|^2}{2(T - t)} \right)}, $$

which increases with an $O(1/(T - t))$ rate as $t \to T$; here $\omega^\Omega(e \mid z, t)$ is the softmax probability measuring the relative closeness of $z$ to the elements $e$ in $\Omega$ (see also Eq (2)).

As we show in Section 2.3, once $f^\theta$ is non-singular in the sense of the mild condition of $\int_0^T \mathbb{E}[\| f^\theta(Z_t, t) \|^2] \, dt < +\infty$, the diffusion model in (3) guarantees to yield a final state $Z_T$ that belongs to $\Omega$, and hence provides a flexible model family on $\Omega$. Moreover, as shown in Eq 1 in Algorithm 1, the neural field $f^\theta$ can be simply trained to approximate $\nabla \log \omega^\Omega(e \mid z, t)$ with $e$ plugged as the data point that we expect to achieve when starting from $z$ at time $t$. Intuitively, such fitted $f^\theta$ increases the relative probability of the observed data points and hence allows us to fit the data distribution. Empirically, diffusion models learned through $\Omega$-bridge achieves favorable results in generating mixed discrete/continuous tabular data, point clouds on grids, categorical semantic segments and discrete CIFAR10 images.
Outline The rest of the paper is organized as follows. Section 2 introduces \( h \)-transform, which allows us to derive bridge processes that are guaranteed to enter specific sets at the terminal time, and Section 2.3 specifies the parametric diffusion models for \( \Omega \)-bridges. Then, with the learnable diffusion models, Section 3 introduces the general learning framework along with the loss function.

2 BACKGROUND: DIFFUSION PROCESSES AND \( h \)-TRANSFORM

A diffusion process \( Z = \{ Z_t : t \in [0, T] \} \) on \( \mathbb{R}^d \) follows a stochastic differential equation of form
\[
Q : \quad dZ_t = b(Z_t, t)dt + \sigma(Z_t, t)dW_t,
\]
where \( W_t \) is a Wiener process, and \( \sigma : [0, T] \times \mathbb{R}^d \to \mathbb{R} \) is a positive diffusion coefficient, and \( b \in [0, T] \times \mathbb{R}^d \to \mathbb{R}^d \) is a drift function. We use \( Q \) (or \( \mathbb{P} \)) to denote the path measure of stochastic processes \( Z \), which are probability measures on the space of continuous paths. Let \( \mathbb{Q}_t \) be the marginal distribution of \( Z_t \) at time \( t \) under \( Q \).

Our framework heavily relies on the bridge processes, special stochastic processes that guarantee to achieve a deterministic value or fall into a given set at the final state \( T \).

\[\text{For a set } \Omega \subseteq \mathbb{R}^d, \text{ a process } Z \text{ in } \mathbb{R}^d \text{ with law } Q \text{ is called an } \Omega \text{-bridge if } Q(Z_T \in \Omega) = 1.\]

One natural approach to constructing bridge processes is to derive the conditioned process of a general unconstrained process given that the desirable bridge constraint happens. Specifically, assume that \( Q \) is the law of a general unconstrained diffusion process of form (4), and denote by \( Q^\Omega(\cdot) = Q(\cdot \mid Z_T \in \Omega) \) the conditioned distribution given that the event of \( Z_T \in \Omega \) happens. Then \( Q^\Omega \) is guaranteed to be an \( \Omega \)-bridge by definition. Importantly, a remarkable result from Doob (1984), now known as \( h \)-transform, shows that \( Q^\Omega \) is the law of a diffusion process with a properly modified drift term. Below, we introduce this results, first for the case \( x \)-bridge when \( \Omega = \{ x \} \) includes a single point, and then for more general sets \( \Omega \).

2.1 \( x \)-BRIDGES

Let us first consider the \( x \)-bridge \( Q^x(\cdot) := Q(\cdot \mid Z_T = x) \), the process \( Q \) pinned at a deterministic terminal point \( Z_T = x \). By the result from \( h \)-transform (see e.g., Oksendal (2013)), the conditioned process \( Q^x(\cdot) := Q(\cdot \mid Z_T = x) \), if it exists, can be shown to be the law of
\[
dZ_t = \left( b(Z_t, t) + \sigma^2(Z_t, t) \nabla_x \log q_{T\mid t}(x \mid Z_t) \right) dt + \sigma(Z_t, t)dW_t,
\]
where \( q_{T\mid t}(x \mid z) \) is the density function of the transition probability
\[
Q_{T\mid t}(dz \mid z) := Q(Z_T \in dz \mid Z_t = z),
\]
where \( dz \) denotes an infinitesimal volume centering around \( x \). Compared with the diffusion process (4) of \( Q \), the main difference is that the conditioned process has an additional drift force \( \sigma^2(z, t) \nabla_x \log q_{T\mid t}(x \mid z) \) which plays the role of steering \( Z_t \) towards the target \( Z_T = x \); this is a singular force whose magnitude increases to infinity as \( t \to T \), because \( q_{T\mid t}(\cdot \mid z) \) is a delta measure centered at \( z \) when \( t = T \).

In addition, by Bayes rule, the distribution of the initial state \( Z_0 \) should be given by
\[
Z_0 \sim Q_{0\mid T}(\cdot \mid x), \quad Q_{0\mid T}(dz \mid x) \propto Q_0(dz)q_{0\mid T}(x \mid z).
\]

Example 2.1. If \( Q \) is the law of \( dZ_t = \sigma_t dW_t \), we have \( Q_{T\mid t}(\cdot \mid z) = N(z, \beta_T - \beta_t) \), where \( \beta_t = \int_0^t \sigma_s^2 ds \). Hence, following the formula in (5), \( Q^x := Q(\cdot \mid Z_T = x) \) is the law of
\[
dZ_t = \sigma_t^2 \frac{x - Z_t}{\beta_T - \beta_t} dt + \sigma_t dW_t,
\]
and \( Z_0 \sim Q_{0\mid T}(dz) \propto Q_0(dz)\phi(x \mid z, \beta_T - \beta_0) \), and \( \phi(\cdot \mid \mu, \sigma^2) \) is the density function of \( N(\mu, \sigma^2) \). The process in (7) is known as a (time-scaled) Brownian bridge. Note that the drift in (7) grows to infinity in magnitude with a rate of \( O(1/|\beta_T - \beta_t|) \) as \( t \to T \), which ensures that \( Z_t = x \) with probability one.
Arbitrary initialization. To make (5) the conditioned process of (4), the initial distribution must follow the Bayes rule in (6). However, thanks to the singular force $\nabla_z \log q_{T|t}(x|z)$, the process (5) can guarantee $Z^*_T = x$ from an arbitrary initialization once the process is well defined. When the initialization is different from (6), the process in (5) is no longer the conditioned process of (4), but it remains to be an $x$-bridge in that $Z_T = x$ is still guaranteed. To see why this is the case, assume that $\mathbb{Q}$ is initialized from a deterministic point $Z_0 = x_0$. Then we would still have $Z_0 = x_0$ when conditioned on $Z_T \in \Omega$ by Bayes rule. This suggests that (5) starting from any deterministic initialization is the conditioned process of $\mathbb{Q}$ with the same deterministic initialization, and is hence an $x$-bridge. As a result, (5) from any stochastic initialization is also an $x$-bridge because it can be viewed as the mixture of the processes with different deterministic initialization, all of which are $x$-bridges. See Appendix A.4 for a detailed analysis, in which it is shown that (5) with an arbitrary initialization can be viewed as the conditioned process of a special class of non-Markov processes called reciprocal process.

2.2 \(\Omega\)-bridges

More generally, for the law \(\mathbb{Q}\) of (4) and a set $\Omega \in \mathbb{R}^d$, the \(\Omega\)-bridge $\mathbb{Q}^{\Omega} := \mathbb{Q}(\cdot | Z_T \in \Omega)$ follows

$$
\mathbb{Q}^{\Omega} : \quad dZ_t = \eta^\Omega(Z_t, t)dt + \sigma(Z_t, t)dW_t,
$$

with

$$
\eta^\Omega(z, t) = \beta_t
$$

and

$$
\eta^\Omega(z, t) = \beta_t \mathbb{E}_{x \sim N_0(z, \theta_T - \beta_t)} [\nabla_z \log q_{T|t}(x|z)], \quad Z_0 \sim \mathbb{Q}_{0|T}(\cdot | Z_T \in \Omega),
$$

where drift force $\eta^\Omega$ is similar to that of the $x$-bridge in (5), except that the final state $x$ is now randomly drawn from an $\Omega$-truncated (or \(\Omega\)-conditioned) transition probability:

$\mathbb{Q}_{T|t}(d|x | z) := \mathbb{Q}_T(d|x | Z_t = z, Z_T \in \Omega),$

which is the transition probability from $Z_t$ to $Z_T$, conditioned on that $Z_T \in \Omega$. In practice, its form can be derived using Bayes rule.

Example 2.2. Assume \(\mathbb{Q}\) follows $dZ_t = \sigma_t dW_t$. Then $\mathbb{Q}^{\Omega}$ yields the following $\Omega$-bridge:

$$
\mathbb{Q}^{\Omega} : \quad dZ_t = \eta^\Omega(Z_t, t)dt + \sigma_t dW_t,
$$

with

$$
\eta^\Omega(z, t) = \beta_t \mathbb{E}_{x \sim N_0(z, \theta_T - \beta_t)} [\nabla_z \log q_{T|t}(x|z)], \quad Z_0 \sim \mathbb{Q}_{0|T}(\cdot | Z_T \in \Omega),
$$

where $N_0(z, \sigma^2)$ is a neural network with input $(z, t)$ and parameter $\theta$, which can be a trainable neural network, on top of the $\Omega$-bridge in (5). Specifically, we construct the following parametric diffusion model $\mathbb{P}^\theta$:

$$
\mathbb{P}^\theta : \quad dZ_t = (\sigma(Z_t, t)f^\theta(Z_t, t) + \eta^\Omega(Z_t, t))dt + \sigma(Z_t, t)dW_t, \quad Z_0 \sim \mathbb{P}^\theta_0,
$$

where $f^\theta(z, t)$ is a neural network with input $(z, t)$ and parameter $\theta$, which will be trained based on the empirical observations. Adding the neural drift $\sigma(Z_t, t)f^\theta(Z_t, t)$ term does not break the $\Omega$-bridge condition, once it satisfies a very mild regularization condition:

Proposition 2.3. For any $\mathbb{Q}^{\Omega}$ following $dZ_t = \eta^\Omega(Z_t, t)dt + \sigma(Z_t, t)dW_t$ that is an $\Omega$-bridge, the $\mathbb{P}^\theta$ in (10) is also an $\Omega$-bridge if $\mathbb{E}_{Z \sim \mathbb{Q}^{\Omega}} [\int_0^T \|f^\theta(Z_t, t)\|^2 dt] < +\infty$ and $K\mathcal{L}(\mathbb{Q}^{\Omega}) \| \mathbb{P}^\theta \| < +\infty$. 

The condition on $f^\theta$ is very mild, and it is satisfied if $f^\theta$ is bounded. Moreover, it can easily hold even when $f_\theta$ is not bounded. For example, assuming that $f_\theta(x) \leq a \|x\|^2 + b$, which holds for ReLU network with $\beta = 1$, we just need to require that the underlying process has a bounded moment $E_{Z \sim Q^\theta} \left[ \int_0^T \|Z_t\|^{2\beta} \, dt \right] < +\infty$, which is a typical regularity condition to expect.

3 Learning $\Omega$-Bridge Models

Let $\{x^{(i)}\}_{i=1}^n$ be an i.i.d. sample from an unknown distribution $\Pi^*$ on a domain $\Omega \subseteq \mathbb{R}^d$. Our goal is to learn the parameter $\theta$ for the $\Omega$-bridge model $\mathbb{P}^\theta$ in (10) such that the terminal distribution $Z_T \sim \mathbb{P}^\theta_T$ matches the data $X \sim \Pi^*$. We should distinguish $\mathbb{P}^\theta$, which is the trainable generative model, and $Q$, which is a fixed “baseline process” that helps us to derive methods for constructing and learning the model. $Q$ can be the simple Brownian motion in Example 2.1 and 2.2.

As the case of other diffusion models, $\mathbb{P}^\theta$ can be viewed as a model with an infinite dimensional latent variable of the intermediate trajectories of $Z$. Hence, a canonical learning approach is expectation maximization (EM), which alternates between

1) E-step: estimating the posterior $\mathbb{P}^\theta|x := \mathbb{P}^\theta(Z | Z_T = x)$ of the latent trajectories $Z$ given the observation $Z_T = x$.

2) M-step: estimating the parameter $\theta$ with $Z$ imputed from $\mathbb{P}^\theta|x$.

A key challenge, however, is that the posterior distribution $\mathbb{P}^\theta|x$ is difficult to calculate due to the presence of neural force field in $\mathbb{P}^\theta$ (as the $h$-transform formula would have no closed form), and it need to be iteratively updated as $\theta$ changes. Following DDPM (Ho et al., 2020), we consider a simpler approach that replaces the posterior $\mathbb{P}^\theta|x$ with an arbitrary $x$-bridge, denoted by $Q^x$. This yields a simplified EM algorithm without the expensive posterior inference in the E-step. A natural choice is the conditioned process $Q^x := Q(Z | Z_T = x)$, but the method works for a general $x$-bridge.

Specifically, let $Q^{\Pi^*}(\cdot) = \int Q^x(\cdot)\Pi^*(dx)$ be the mixture of the $x$-bridges whose end point $x$ is randomly drawn from the data distribution $x \sim \Pi^*$. The trajectories from $Q^{\Pi^*}$ can be generated in the following “backward” way: first drawing a data point $x \sim \Pi^*$, and then $Z \sim Q^x$ conditioned on the end point $x$. Obviously, by construction, the terminal distribution of $Q^{\Pi^*}$ equals $\Pi^*$, that is, $Q^{\Pi^*}_T = \Pi^*$. Then, the model $\mathbb{P}^\theta$ can be estimated by fitting data drawn from $Q^{\Pi^*}$ using maximum likelihood estimation:

$$
\min_{\theta} \left\{ \mathcal{L}(\theta) := \mathcal{KL}(Q^{\Pi^*} \| \mathbb{P}^\theta) \right\},
$$

(11)

The classical (variational) EM would alternatively update $\theta$ (M-step) and $Q^x$ (E-step) to make $Q^x \approx \mathbb{P}^\theta|x$. Why is it OK to simply drop the E-step? At the high level, it is the benefit from using universal approximators like deep neural networks: if the model space of $\mathbb{P}^\theta$ is sufficiently rich, by minimizing the KL divergence in (11), $\mathbb{P}^\theta$ can approximate the given $Q^{\Pi^*}$ well enough (in a way that is made precise in the Appendix A) such that their terminal distributions are close: $\mathbb{P}^\theta_T \approx Q^{\Pi^*}_T = \Pi^*$.

**Learning latent variable models require no E-step if the model space is sufficiently rich.**

We should see that in this case the latent variables $Z$ in the learned model $\mathbb{P}^\theta$ is dictated by the choice of the imputation distribution $Q$ since we have $\mathbb{P}^\theta|x = Q^x$ when the KL divergence in (11) is fully minimized to zero; EM also achieves $\mathbb{P}^\theta|x = Q^x$ but has the imputation distribution $Q^x$ determined by the model $\mathbb{P}^\theta$, not the other way.

Loss Function In its general form, the $x$-bridge $Q^x$ that we use can be a non-Markov diffusion process

$$
Q^x : \quad dZ_t = \eta^x(Z, t) \, dt + \sigma(Z, t) \, dW_t, \quad Z_0 \sim \mu^x,
$$

(12)

which has the same diffusion coefficient $\sigma(Z, t)$ as $\mathbb{P}^\theta$ in (10), and any $x$-dependent $\eta^x$ and initialization $\mu^x$ once the $x$-bridge condition is ensured. As the general framework, we assume that $\eta^x$ can depend on the whole trajectory $Z$.
By default, we set the initialization $Q^x$ and an $\Omega$-bridge $Q^\Omega$.

$$dZ_t = \eta^x(Z_t, t)dt + \sigma(Z_t, t)dW_t,$$

Specify the generative model $P^\theta$ based on $Q^\Omega$ and a neural network $f^\theta$:

$$dZ_t = \sigma(Z_t, t)f^\theta(Z_t, t) + \eta^\Omega(Z_t, t)dt + \sigma(Z_t, t)dW_t, \quad Z_0 \sim P^\theta.$$

**Default:** let $Q$ be the law of $dZ_t = \sigma dW_t$ and derive the bridges by $h$-transform as $Q^x = Q(|Z_T = x)$ in Eq (7) and $Q^\Omega = Q(|Z_T \in \Omega)$ in Eq (9).

**Training:** Estimating $\theta$ by minimizing the loss function (13) using any off-the-shelf optimizer.

**Sampling:** Generate sample $Z_T$ from $P^\theta$ with the trained parameter $\theta$.

Using Girsanov theorem (e.g., Oksendal, 2013), with $P^\theta$ in (10) and $Q^x$ in (12), the KL divergence in (11) can be shown to equal to

$$\mathcal{L} = \mathbb{E}_{Z\sim Q^x}\left[ -\log p_0^\theta(Z_0) + \frac{1}{2} \int_0^T \left\| \frac{1}{\sigma}(Z_t, t)(s^\theta(Z_t, t) - \eta^x(Z_t, t)) \right\|^2 dt \right] + \text{const}$$

where we write $s^\theta$ as the overall drift force of $P^\theta$ in (10), that is,

$$s^\theta(z, t) = \sigma(z, t)f^\theta(z, t) + \eta^\Omega(z, t),$$

and $p_0^\theta$ is the probability density function (PDF) of the initial distribution $P_0^\theta$. Therefore, $\mathcal{L}(\theta)$ is a sum of the negative log-likelihood of the initial distribution that encourages $P_0^\theta \approx Q^x$ and a least squares loss between $s^\theta$ and $\eta^x$. In practice, we simply fix the initial distribution $P_0^\theta$ to be a delta measure on a fixed point (say, $x_0 = 0$), so we only need to train the drift function $f^\theta$. Algorithm 1 shows a simple instance of the framework when the baseline process $Q$ is the standard Brownian motion $dZ_t = dW_t$, $Q^x = Q(\cdot | Z_T = x)$ and $\sigma(z, t) = 1$. Note that the least squares term in (13) can be viewed as enforcing $f^\theta \approx \sigma^{-1}(\eta^x - \eta^\Omega)$, which reduces to $f^\theta \approx \nabla \log \omega^\Omega$ in the case of Algorithm 1.

**Related Works** Bridge processes provide a simple and flexible approach to learning diffusion generative models, which was explored in Peluchetti (2021); Ye et al. (2022); Wu et al. (2022); De Bortoli et al. (2021). Heng et al. (2021) investigates the orthogonal problem of simulating from the bridge $Q^x$ for a given $Q$. In comparison, our method learns diffusion models on general domains $\Omega$ on which an $\Omega$-bridge can be derived (using $h$-transform or any other method), and hence provides a highly flexible framework for learning with structured data (including discrete, continuous, and their mixes). This distinguishes it with existing approaches that are designed for special types of data (e.g., Ho et al. (2020); Hoogeboom et al. (2021); Austin et al. (2021); Li et al. (2022); Dieleman et al. (2022) for discrete data). De Bortoli et al. (2022) discusses how to learn score-based generative models on general Riemannian manifolds. Another highly related work is Ye et al. (2022), which proposes to learn first hitting diffusion models for generating data on both discrete sets and spheres. The advantage of our approach is that it is simpler and easier to derive for more complex types of domains.

## 4 EXPERIMENTS

We evaluate our algorithms for generating mixed-typed tabular data, grid-valued point clouds, categorical semantic segmentation maps, discrete CIFAR10 images. We observe that $\Omega$-bridge provides a particularly attractive and superb approach to generating data from various constrained domains.

**Algorithm Overview** For all experiments, we use Algorithm 2 with the default choice of $Q^x$ in (7) and $Q^\Omega$ in (9). The specific form of $\eta^\Omega$ is derived based on the specific choice of the domain $\Omega$. By default, we set the initialization $Z_0 = 0$ and the optimizer Adam.
where \( \eta \) is the standard Gaussian CDF. Finally, we have \( \eta^t(z, t) = \sigma_t^2 \nabla_z \log \sum_{e \in I} \exp \left( -\frac{||z-e||^2}{2(\beta_t - \beta_1)} \right) \); for non-negative continuous domains, \( I = [0, +\infty) \), derivation shows \( \eta^t(z, t) = \sigma_t^2 \nabla_z \log \left( \frac{F(z)}{\sqrt{\beta_t - \beta_1}} \right) \), where \( F \) is the standard Gaussian CDF. Finally, we have \( \eta^\Omega(z, t) = \sum_{i=1}^{15} \eta^t(z_i, t) \) for the whole domain \( \Omega \). We set the number of diffusion steps to \( K = 2000 \). Results are shown in Table 1.

**Result** All the three different classifiers yield the highest accuracy when trained on the data generated by our method, referred to as Mixed-Bridge in this case. The result reflects that the data generated by Mixed-Bridge is closer to the real distribution than the baseline methods.

<table>
<thead>
<tr>
<th>Method</th>
<th>MMD ↓</th>
<th>COV ↑</th>
<th>1-NNA ↓</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCD (Luo &amp; Hu, 2021a)</td>
<td>13.37</td>
<td>46.60</td>
<td>58.94</td>
</tr>
<tr>
<td>( \mathbb{R}^d )-Bridge</td>
<td>13.30</td>
<td>46.52</td>
<td>59.32</td>
</tr>
<tr>
<td>Grid-Bridge</td>
<td><strong>12.85</strong></td>
<td><strong>47.78</strong></td>
<td><strong>56.25</strong></td>
</tr>
</tbody>
</table>

**Figure 2 & Table 2:** The point clouds (upper row) generated by different methods and meshes reconstructed from them (lower row). Grid-Bridge obtains more uniform points and hence better mesh thanks to the integer constraints. Numbers in the table are multiplied by \( 10^2 \).

### 4.1 Generating Mixed-Type Tabular Data

Learning to generate tabular data is challenging, because tabular data usually contains a mixture of discrete and continuous attributes (Xu et al., 2019; Park et al., 2018). Unlike carefully designing special GANs as in previous works (Xu et al., 2019; Srivastava et al., 2017), \( \Omega \)-bridge can be seamlessly applied to mixed-typed tabular data generation without any further modification. In contrast, diffusion processes that solely work on discrete domain (Austin et al., 2021; Hoogeboom et al., 2021) cannot be applied to this task.

In this experiment, we use the Adult Income dataset (Kohavi, 1996), which contains 30,162 training samples. The data points are described by a series of attributes, including continuous (age, capital-gain, etc.) and discrete (sex, race, etc.). We compare with conditional tabular GAN (Xu et al., 2019) (CTGAN), CopulaGAN (Patki et al., 2016), and Table VAE (Xu et al., 2019) (TVAE), which are state-of-the-art GAN-based and VAE-based generative models for mixed-typed tabular data. Following previous works (Xu et al., 2019; Patki et al., 2016), we measure the classification accuracy on the real data of logistic regression, AdaBoost classifier and MLP classifier when trained on the generated data.

### 4.2 Generating Integer-valued Point Clouds

A feature of point clouds in 3D objects in graphics is that they tend to distribute evenly, especially if they are discretized from a mesh. This aspect is omitted in most existing works on point cloud generation. As a result they tend to generate non-uniform points that are unsuitable for real applications, which often involve converting back to meshes with procedures like Ball-Pivoting (Bernardini...
We apply our method to generate point clouds that constrained on an integer grid which we show yields much more uniformly distributed points. To the best of our knowledge, we are the first work on integer-valued 3D point cloud generation.

A point cloud is a set of points \( \{ x_i \}_{i=1}^m \), \( x_i \in \mathbb{R}^3 \) in the 3D space, where \( m \) refers to the number of points. We apply two variants of our method: \( \mathbb{R}^d \)-Bridge and Grid-Bridge. \( \mathbb{R}^d \)-Bridge generates points in the continuous 3D space, i.e., \( \Omega = \mathbb{R}^3 \). Grid-Bridge generate points on integer grids, \( \Omega_{\text{Grid}} = \{1, \ldots, 128\}^3 \). We fix the diffusion coefficient \( \sigma_t = 1 \). The number of diffusion steps \( K \) is set to 1000. We test our method on ShapeNet (Chang et al., 2015) chair models, and compare it with Point Cloud Diffusion (PCD) (Luo & Hu, 2021a), a state-of-the-art continuous diffusion-based generative model for point clouds. The neural network \( f^\theta \) in our methods are the same as that of PCD for fair comparison. Qualitative results and quantitative results are shown in Figure 2 and Table 2. As common practice (Luo & Hu, 2021a,b), we measure minimum matching distance (MMD), coverage score (COV) and 1-NN accuracy (1-NNA) using Chamfer Distance (CD) with the test dataset.

**Result** Both \( \mathbb{R}^d \)-Bridge and Grid-Bridge get better MMD, COV, and 1-NNA than PCD. Moreover, by constraining the domain of interest to the integer grids, Grid-Bridge yields even better performance than \( \mathbb{R}^d \)-Bridge. In Figure 2, since the point clouds generated by Grid-Bridge are limited to integer grids, the reconstructed meshes from Ball-Pivoting clearly have higher quality than \( \mathbb{R}^d \)-Bridge and PCD.

### 4.3 Generating Semantic Segmentation Maps on CityScapes

We consider unconditionally generating categorical semantic segmentation maps. We represent each pixel as a one-hot categorical vector. Hence the data domain is \( \Omega = \{e_1, \ldots, e_c\}^{h \times w} \), where \( c \) is the number of classes and \( e_i \) is the \( i \)-th \( c \)-dimensional one-hot vector, and \( h, w \) represent the height and width of the image. In CityScapes (Cordts et al., 2016), \( h = 32, w = 64, c = 8 \). In this experiment, we test different schedule of the diffusion coefficient \( \sigma_t \), including (Constant Noise): \( \sigma_t = 1 \); (Noise Decay A): \( \sigma_t = a \exp(-bt) \); (Noise Decay B): \( \sigma_t = a(1-t) \); (Noise Decay C) \( \sigma_t = a(1 - \exp(-b(1-t))) \). Here \( a \) and \( b \) are hyper-parameters. The number of diffusion steps \( K \) is set to 500. We measure the negative log-likelihood (NLL) of the test set using the learned models. The NLL (bits-per-dimension) is estimated with evidence lower bound (ELBO) and importance weighted bound (IWBO) (Burda et al., 2016), respectively, as in (Hoogeboom et al., 2021). We compare \( \Omega \)-Bridge with a state-of-the-art categorical diffusion algorithm, Argmax Flow (and Multinomial Diffusion) (Hoogeboom et al., 2021), and the traditional methods, uniform dequantization (Uria et al., 1999). We apply our method to generate point clouds that constrained on an integer grid which we show yields much more uniformly distributed points. To the best of our knowledge, we are the first work on integer-valued 3D point cloud generation.

<table>
<thead>
<tr>
<th>Methods</th>
<th>ELBO (↓)</th>
<th>IWBO (↓)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniform Dequantization (Uria et al., 2013)</td>
<td>1.010</td>
<td>0.930</td>
</tr>
<tr>
<td>Variational Dequantization (Ho et al., 2019)</td>
<td>0.334</td>
<td>0.315</td>
</tr>
<tr>
<td>Argmax Flow (Softplus thres.) (Hoogeboom et al., 2021)</td>
<td>0.303</td>
<td>0.290</td>
</tr>
<tr>
<td>Argmax Flow (Gumbel distr.) (Hoogeboom et al., 2021)</td>
<td>0.365</td>
<td>0.341</td>
</tr>
<tr>
<td>Argmax Flow (Gumbel thres.) (Hoogeboom et al., 2021)</td>
<td>0.307</td>
<td>0.287</td>
</tr>
<tr>
<td>Multinomial Diffusion (Hoogeboom et al., 2021)</td>
<td>0.305</td>
<td>-</td>
</tr>
<tr>
<td>Cat.-Bridge (Constant Noise)</td>
<td>0.844</td>
<td>0.707</td>
</tr>
<tr>
<td>Cat.-Bridge (Noise Decay A)</td>
<td>0.276</td>
<td>0.232</td>
</tr>
<tr>
<td>Cat.-Bridge (Noise Decay B)</td>
<td>0.301</td>
<td>0.285</td>
</tr>
<tr>
<td>Cat.-Bridge (Noise Decay C)</td>
<td>0.363</td>
<td>0.302</td>
</tr>
</tbody>
</table>

Table 3: Results on the CityScapes dataset. Cat. refers to "Categorical".
Methods | IS (↑) | FID (↓) | NLL (↓)
--- | --- | --- | ---
**Discrete**
D3PM uniform $L_{vb}$ (Austin et al., 2021) | 5.99 | 51.27 | 5.08
D3PM absorbing $L_{vb}$ (Austin et al., 2021) | 6.26 | 41.28 | 4.83
D3PM Gauss $L_{vb}$ (Austin et al., 2021) | 7.75 | 15.30 | 3.96
D3PM Gauss $L_{\lambda=0.001}$ (Austin et al., 2021) | 8.54 | 8.34 | 3.97
D3PM Gauss + logistic $L_{\lambda=0.001}$ | 8.56 | 7.34 | 3.43
Integer-Bridge (Init. A) | 8.77 | 6.77 | 3.46
Integer-Bridge (Init. B) | 8.68 | 6.91 | **3.35**
Integer-Bridge (Init. C) | 8.72 | 6.94 | 3.40

Table 4: Discrete CIFAR10 Image Generation

et al., 2013) and variational dequantization (Ho et al., 2019). The numerical results of the baselines are directly adopted from (Hoogeboom et al., 2021), and experiment configuration is kept the same for fair comparison. The neural network $f^\theta$ is the same as (Hoogeboom et al., 2021). The results are shown in Figure 3 and Table 3. Our $\Omega$-bridge is named *Categorical-Bridge (Cat.-Bridge)* in this experiment.

**Result** We observe that all the four kinds of Cat.-Bridge can successfully generate categorical semantic segments, and different noise schedules result in different empirical performance. Among the four variants of Cat.-Bridge, Cat.-Bridge with Noise Decay A yields the best ELBO and IWBO, surpassing all the other algorithms in comparison.

### 4.4 Generating Discrete CIFAR10 Images

In this experiment, we apply three types of bridges. All of these bridges use the same output domain $\Omega = \{0, \ldots, 255\}^{h \times w \times c}$, where $h, w, c$ are the height, width and number of channels of the images, respectively. We set $\sigma_t = 3 \exp(-3t)$. We consider different initial distributions: *(Init. A)* $Z_0 = 128$; *(Init. B)* $Z_0 = \hat{\mu}_0, \hat{\sigma}_0$, *(Init. C)* $Z_0 \sim \mathcal{N}(\hat{\mu}_0, \hat{\sigma}_0)$, where $\hat{\mu}_0$ and $\hat{\sigma}_0$ are the empirical mean and variance of pixels in the CIFAR10 training set. The number of diffusion steps $K$ is set to 1000. We compare with the variants of a state-of-the-art discrete diffusion model, D3PM (Austin et al., 2021). For fair comparison, we use the DDPM backbone (Ho et al., 2020) as the neural drift $f^\theta$ in our method, similar to D3PM. We report the Inception Score (IS) Salimans et al. (2016), Fréchet Inception Distance (FID) Heusel et al. (2017) and negative log-likelihood (NLL) of the test dataset. We call our method *Integer-Bridge* in this case. The results are shown in Table 4 and Figure 4.

**Result** In Table 4, Integer-Bridge with Initialization A,B,C can all get lower FIDs ($\leq 7$) than the variants of D3PM. Among the three kinds of Integer-bridges, Integer-Bridge (Init. B) obtains the lowest NLL (3.35). It also beats D3PM Gauss + logistic (3.435) on NLL, which has the best NLL in the variants of D3PM.

### 5 Conclusion and Limitations

We present a framework for learning diffusion generative models on constrained data domains. It leaves a number of directions for further explorations and improvement. For example, the practical impact of the choices of the bridges $Q$, in terms of initialization, dynamics, and noise schedule, are still not well understood and need more systematical studies. Besides, our current method is limited to $\Omega$ that are factorizable and integrable. Moreover, application of $\Omega$-bridge to many other practical fields also needs investigation in the future.
ACKNOWLEDGEMENTS

This research is supported by NSF CAREER1846421, SenSE2037267, EAGER-2041327, Office of Navy Research, and NSF AI Institute for Foundations of Machine Learning (IFML).

REFERENCES


The appendix is structured as follows:

- Appendix A provides the theoretical analysis and derivation of diffusion bridges. In particular, Appendix A.1 shows the derivation of the main training loss; Appendix A.2 derives the drift term \( \eta_{II^*} \) of \( Q_{II^*} \); Appendix A.3 proves that we can actually use a Markov model \( \mathbb{P}^0 \) to match all time-marginals with \( Q_{II^*} \); Appendix A.4 explains why we can use arbitrary initialization when constructing bridge processes and discusses the reciprocal structure of \( Q_{II^*} \); Appendix A.5 provides analysis on the time-discretization error and statistical error of the practical discretized algorithm. Appendix A.6 and A.7 presents details on the condition and examples of \( \Omega \)-bridge discretization.

- Appendix B shows additional experiment details and results.

A  THEORETICAL ANALYSIS ON BRIDGES

A.1 DERIVATION OF THE MAIN LOSS IN EQUATION (13)

[Proof of Equation (13)] Denote by \( Q^x = Q(\cdot | Z_T = x) \). Note that

\[
\mathcal{KL}(Q_{II^*}^x \mid \mathbb{P}^0) = \mathbb{E}_{x \sim \Pi^*, Z \sim Q^x} \left[ \log \frac{dQ_{II^*}^x}{d\mathbb{P}^0}(Z) \right]
= \mathbb{E}_{x \sim \Pi^*, Z \sim Q^x} \left[ \log \frac{dQ_{II^*}^x}{d\mathbb{P}^0}(Z) + \log \frac{dQ_{II^*}^x}{dQ_{II^*}^x}(Z) \right]
= \mathbb{E}_{x \sim \Pi^*} \left[ \mathcal{KL}(Q^x \mid \mathbb{P}^0) \right] + \text{const},
\]

where \( \text{const} \) denotes a constant that is independent of \( \theta \). Recall that \( Q^x \) follows \( dZ_t = \eta_t^x(Z_{0,t}, t)dt + \sigma(Z_t, t)dW_t \), and \( \mathbb{P}^0 \) follows \( dZ_t = s^\theta(Z_t, t)dt + \sigma(Z_t, t)dW_t \). By Girsanov theorem (e.g., Lejay, 2018),

\[
\mathcal{KL}(Q^x \mid \mathbb{P}^0) = \mathcal{KL}(Q_0^x \mid \mathbb{P}_0^\theta) + \frac{1}{2} \mathbb{E}_{Z \sim Q^x} \left[ \int_0^T \| s^\theta(Z_t, t) - \eta_t^x(Z_{0,t}, t) \|^2 dt \right]
= \mathbb{E}_{Z \sim Q^x} \left[ -\log p_0^\theta(Z_0) + \frac{1}{2} \int_0^T \| s^\theta(Z_t, t) - \eta_t^x(Z_{0,t}, t) \|^2 dt \right] + \text{const}.
\]

Hence

\[
L(\theta) = \mathbb{E}_{x \sim \Pi^*, Z \sim Q^x} \left[ -\log p_0^\theta(Z_0) + \frac{1}{2} \int_0^T \| s^\theta(Z_t, t) - \eta_t^x(Z_{0,t}, t) \|^2 dt \right] + \text{const}
= \mathbb{E}_{Z \sim Q^x} \left[ -\log p_0^\theta(Z_0) + \frac{1}{2} \int_0^T \| s^\theta(Z_t, t) - \eta_t^x(Z_{0,t}, t) \|^2 dt \right] + \text{const}.
\]

A.2 DERIVATION OF THE DRIFT \( \eta_{II^*} \) OF \( Q_{II^*} \)

Lemma A.1. Let \( Q^x \) is the law of

\[
dZ_t^x = \eta_t^x(Z_{0,t}, t)dt + \sigma(Z_t, t)dW_t, \quad Z_0 \sim Q_0^x,
\]

and \( Q_{II^*} := \int Q^x(Z) \Pi^*(dx) \) for a distribution \( \Pi^* \) on \( \mathbb{R}^d \). Then \( Q_{II^*} \) is the law of

\[
dZ_t = \eta_{II^*}(Z_{0,t}, t)dt + \sigma(Z_t, t)dW_t, \quad Z_0 \sim Q_0^{II^*},
\]

where

\[
\eta_{II^*}(z_{0,t}, t) = \mathbb{E}_{x \sim \Pi^*, Z \sim Q^x}[\eta_t^x(Z_{0,t}) dZ_t = z_{0,t}], \quad Q_0^{II^*}(dz_0) = \mathbb{E}_{x \sim \Pi^*}[Q_0^x(dz_0)].
\]

[Proof] \( Q_{II^*} \) is the solution of the following optimization problem:

\[
Q_{II^*} = \arg \min_{\mathbb{P}} \left\{ \mathcal{KL}(Q_{II^*} \mid \mathbb{P}) = \mathbb{E}_{x \sim \Pi^*}[\mathcal{KL}(Q^x \mid \mathbb{P})] + \text{const} \right\}.
\]
By Girsanov’s Theorem (e.g., Lejay, 2018), any stochastic process \( \mathbb{P} \) that has \( \mathcal{KL}(\mathbb{Q}^\tau || \mathbb{P}) < +\infty \) (and hence is equivalent to \( \mathbb{Q}^\tau \)) has a form of \( dZ_t = \eta^\Pi(t) \sigma(Z_t, t) dt + \sigma(Z_t, t) dW_t \) for some measurable function \( \eta^\Pi \), and
\[
\mathbb{E}_{x \sim \Pi^*} \mathcal{KL}(\mathbb{Q}^\tau || \mathbb{P}) = \mathbb{E}_{x \sim \Pi^*} \mathcal{KL}(\mathbb{Q}_0^\tau || \mathbb{P}_0) + \mathbb{E}_{x \sim \Pi^*, Z \sim \mathbb{Q}^\tau} \left[ \frac{1}{2} \int_0^T \| \sigma(Z_t, t)^{-1}(\eta^\Pi(t) - \eta^\tau(t, 0)) \|^2 \right].
\]

It is clear that to achieve the minimum, we need to take \( \mathbb{P}_0(\cdot) = \mathbb{E}_{x \sim \Pi^*}[\mathbb{Q}_0^\tau(\cdot)] \) and \( \eta^\Pi(z_{0,t}, t) = \mathbb{E}_{x \sim \Pi^*, Z \sim \mathbb{Q}^\tau}[\eta^\tau(Z_{0,t}, t) \mid Z_{0,t} = z_{0,t}] \), which yields the desirable form of \( \mathbb{Q}^\Pi^* \).

### A.3 \( \mathbb{P}^\theta^* \) YIELDS A MARKOVIZATION OF \( \mathbb{Q}^\Pi^* \)

As \( \mathbb{P}^\theta \) is Markov by the model assumption, it can not perfectly fit \( \mathbb{Q}^\Pi^* \) which is non-Markov in general. This is a substantial problem because \( \mathbb{Q}^\Pi^* \) can be non-Markov even if \( \mathbb{Q}^\tau \) is Markov for all \( x \in \Omega \) (see Section A.4). In fact, using Doob’s h-transform method (Doob, 1984), \( \mathbb{Q}^\Pi^* \) can be shown to be the law of a diffusion process
\[
dZ_t = \eta^\Pi(t) \sigma(Z_t, t) dt + \sigma(Z_t, t) dW_t, \quad \eta^\Pi(z_{0,t}, t) = \mathbb{E}_{Z \sim \mathbb{Q}^\tau} [\eta^\tau(z_{0,t}, t) \mid Z_{0,t} = z_{0,t}],
\]
where \( \eta^\Pi \) is the expectation of \( \eta^\tau \) when \( x = Z_T \) is drawn from \( \mathbb{Q} \) conditioned on \( Z_{0,t} \).

We resolve this by observing that it is not necessary to match the whole path measure \( \mathbb{P}^\theta \approx \mathbb{Q}^\Pi^* \) to match the terminal \( \mathbb{P}^\theta_T \approx \mathbb{Q}^\Pi^* \). It is enough for \( \mathbb{P}^\theta \) to be the best Markov approximation (a.k.a. Markovization) of \( \mathbb{Q}^\Pi^* \), which matches all (hence terminal) fixed-time marginals with \( \mathbb{Q}^\Pi^* \):
\[
\text{Proj}(\mathbb{Q}^\Pi^*, \mathcal{M}) := \arg\min_{\mathbb{P} \in \mathcal{M}} \mathcal{KL}(\mathbb{Q}^\Pi^* || \mathbb{P}), \quad \mathcal{M} = \text{the set of all Markov processes on } [0, T].
\]

**Proposition A.2.** The global optimum of \( \mathcal{L}(\theta) \) in (11) and (13) is achieved by \( \theta^* \) if
\[
s^\theta^*(z, t) = \mathbb{E}_{Z \sim \mathbb{Q}^\Pi^*} [\eta^\tau(Z_{0,t}, t) \mid Z_t = z], \quad \mathbb{Q}^\Pi_t = \mathbb{Q}^\Pi^* \text{ for all time } t \in [0, T]. \quad (14)
\]
In this case, \( \mathbb{P}^\theta^* = \text{Proj}(\mathbb{Q}^\Pi^*, \mathcal{M}) \) is the Markovization of \( \mathbb{Q}^\Pi^* \), with which it matches all time-marginals: \( \mathbb{P}^\theta_T = \mathbb{Q}^\Pi_T^* \) for all time \( t \in [0, T] \). In addition,
\[
\mathcal{KL}(\Pi^* || \mathbb{P}^\theta_T) \leq \mathcal{KL}(\Pi^* || \mathbb{P}^\theta) = \mathcal{KL}(\mathbb{Q}^\Pi^* || \mathbb{P}^\theta) - \mathcal{KL}(\mathbb{Q}^\Pi^* || \mathbb{P}) = \mathcal{L}(\theta) - \mathcal{L}(\theta^*). \quad (15)
\]
Note that \( s^\theta^* \) is a conditional expectation of \( \eta^\Pi^* \): \( s^\theta^*(z, t) = \mathbb{E}_{Z \sim \mathbb{Q}^\Pi^*} [\eta^\Pi^*(Z_{0,t}, t) \mid Z_t = z] \).

Theorem 1 of Peluchetti (2021) gives a related result that the marginals of mixtures of Markov diffusion processes can be matched by another Markov diffusion process, but does not discuss the issue of Markovization nor connect to KL divergence. Theorem 1 of Song et al. (2021) is the special case of (15) when \( \mathbb{Q}^\Pi^* \) is Markov.

**Proof of Proposition A.2** It is the combined result of Lemma A.3 and Lemma A.4 below.

**Lemma A.3.** Let \( \mathbb{Q} \) be a non-Markov diffusion process on \([0, T]\) of form
\[
\mathbb{Q} : \quad dZ_t = \eta(Z_{0,t}, t) dt + \sigma(Z_t, t) dW_t, \quad Z_0 \sim \mathbb{Q}_0,
\]
and \( \mathcal{M} = \arg\min_{\mathbb{Q} \in \mathcal{M}} \mathcal{KL}(\mathbb{Q} || \mathbb{P}) \) be the Markovization of \( \mathbb{Q} \), where \( \mathcal{M} \) is the set of all Markov processes on \([0, T]\). Then \( \mathbb{Q} \) is the law of
\[
\mathbb{M} : \quad dZ_t = m(Z_t, t) dt + \sigma(Z_t, t) dW_t, \quad Z_0 \sim \mathbb{Q}_0,
\]
where
\[
m(z, t) = \mathbb{E}_{Z \sim \mathbb{Q}} [\eta(Z_{0,t}, t) \mid Z_t = z].
\]
In addition, we have \( \mathbb{Q}_t = \mathbb{M}_t \) for all time \( t \in [0, T] \).
Lemma A.4. Let

\[ \begin{align*}
\mathcal{Q} : & \quad dZ_t = \eta(Z_{[0,t]}, t)dt + \sigma(Z_t, t)dw_t, \quad Z_0 \sim Q_0 \\
\mathcal{M} : & \quad dZ_t = m(Z_t, t)dt + \sigma(Z_t, t)dw_t, \quad Z_0 \sim Q_0, \\
\mathcal{P}^\theta : & \quad dZ_t = s^\theta(Z_t, t)dt + \sigma(Z_t, t)dw_t, \quad Z_0 \sim \mathcal{P}_0^\theta,
\end{align*} \]

where \( \mathcal{M} \) is the Markovization of \( \mathcal{Q} \) (see Lemma A.3). Then

\[ \mathcal{KL}(\mathcal{Q} \| \mathcal{P}^\theta) = \mathcal{KL}(\mathcal{Q} \| \mathcal{M}) + \mathcal{KL}(\mathcal{M} \| \mathcal{P}^\theta). \]

Hence, assume there exists \( \theta^* \) such that \( \mathcal{P}^{\theta^*} = \mathcal{M} \) and write \( \mathcal{L}(\theta) := \mathcal{KL}(\mathcal{Q} \| \mathcal{P}^\theta) \). We have

\[ \mathcal{KL}(\mathcal{Q}^T \| \mathcal{P}_{T^*}^\theta) = \mathcal{KL}(\mathcal{M}^T \| \mathcal{P}_{T^*}^\theta) \leq \mathcal{KL}(\mathcal{M} \| \mathcal{P}^\theta) = \mathcal{L}(\theta) - \mathcal{L}(\theta^*). \]

[Proof] Note that

\[ \begin{align*}
\mathcal{KL}(\mathcal{M} \| \mathcal{P}^\theta) &= \mathcal{KL}(\mathcal{M}_0 \| \mathcal{P}_0^\theta) + \frac{1}{2} \mathbb{E}_{Z_t \sim \mathcal{M}_t} \left[ \int_0^T \left| \sigma(Z_t, t)^{-1}(s^\theta(Z_t, t) - m(Z_t, t)) \right|^2 dt \right] \\
&= \mathcal{KL}(\mathcal{Q}_0 \| \mathcal{P}_0^\theta) + \frac{1}{2} \int_0^T \mathbb{E}_{Z_t \sim \mathcal{M}_t} \left[ \left| \sigma(Z_t, t)^{-1}(s^\theta(Z_t, t) - m(Z_t, t)) \right|^2 \right] dt \\
&= \mathcal{KL}(\mathcal{Q}_0 \| \mathcal{P}_0^\theta) + \frac{1}{2} \int_0^T \mathbb{E}_{Z_t \sim \mathcal{Q}_t} \left[ \left| \sigma(Z_t, t)^{-1}(s^\theta(Z_t, t) - m(Z_t, t)) \right|^2 \right] dt \\
&= \mathcal{KL}(\mathcal{Q}_0 \| \mathcal{P}_0^\theta) + \frac{1}{2} \int_0^T \mathbb{E}_{Z_t \sim \mathcal{Q}_t} \left[ \left| \sigma(Z_t, t)^{-1}(s^\theta(Z_t, t) - m(Z_t, t)) \right|^2 \right] dt \\
&= \mathcal{KL}(\mathcal{Q}_0 \| \mathcal{P}_0^\theta) + \frac{1}{2} \| s^\theta - m \|_{Q, \sigma}^2,
\end{align*} \]

where we define \( \| f \|_{Q, \sigma}^2 = \mathbb{E}_{Z_t \sim \mathcal{Q}_t} \left[ \frac{1}{2} \int_0^T \left| \sigma(Z_t, t)^{-1}f(Z_t, t) \right|^2 dt \right]. \)

On the other hand,

\[ \begin{align*}
\mathcal{KL}(\mathcal{Q} \| \mathcal{P}^\theta) &= \mathcal{KL}(\mathcal{Q}_0 \| \mathcal{P}_0^\theta) + \frac{1}{2} \mathbb{E}_{Z_t \sim \mathcal{Q}_t} \left[ \int_0^T \left| \sigma(Z_t, t)^{-1}(s^\theta(Z_t, 0) - \eta(Z_{[0,t]}, t)) \right|^2 dt \right] \\
&= \mathcal{KL}(\mathcal{Q}_0 \| \mathcal{P}_0^\theta) + \frac{1}{2} \| s^\theta - \eta \|_{Q, \sigma}^2 \\
\mathcal{KL}(\mathcal{Q} \| \mathcal{M}) &= \mathbb{E}_{Z_t \sim \mathcal{Q}_t} \left[ \frac{1}{2} \int_0^T \left| \sigma(Z_t, t)^{-1}(\eta(Z_{[0,t]}, t) - m(Z_t, 0)) \right|^2 dt \right] \\
&= \frac{1}{2} \| \eta - m \|_{Q, \sigma}^2.
\end{align*} \]
Using Lemma A.5 with \( a(z) = \sigma(z, t)^{-1}s^\theta(z, t) \) and \( b(z_{[0, t]}) = \sigma(z, t)^{-1}\eta(z_{[0, t]}, t) \), we have the following bias-variance decomposition:

\[
\|\eta - s^\theta\|_{Q, \sigma}^2 = \|s^\theta - m\|_{Q, \sigma}^2 + \|\eta - m\|_{Q, \sigma}^2.
\]

Hence, \( KL(Q \| P^\theta) = KL(M \| P^\theta) + KL(Q \| M) \).

Finally, \( KL(M_T \| P^\theta_T) \leq KL(M \| P^\theta) \) is the direct result of the following factorization of KL divergence:

\[
KL(M \| P^\theta) = KL(M_T \| P^\theta_T) + E_{x \sim M_T} [KL(M_T(\cdot|Z_T = x) \| P^\theta_T(\cdot|Z_T = x))] .
\]

Lemma A.5. Let \((X, Y)\) be a random variable and \(a(x), b(x, y)\) are square integral functions. Let \(m(x) = E[b(X, Y) \mid X = x]\). We have

\[
E[\|a(X) - b(X, Y)\|_2^2] = E[\|a(X) - m(X)\|_2^2] + E[\|b(X, Y) - m(X)\|_2^2].
\]

[Proof] \(E[\|a(X) - b(X, Y)\|_2^2] = E[\|a(X) - m(X) + m(X) - b(X, Y)\|_2^2]

= E[\|a(X) - m(X)\|_2^2] + E[\|m(X) - b(X, Y)\|_2^2] + 2\Delta,
\]

where \(\Delta = E[(a(X) - m(X))^\top (m(X) - b(X, Y))]

= E[(a(X) - m(X))^\top E[(m(X) - b(X, Y))|X]]

= E[(a(X) - m(X))^\top (m(X) - m(X))] = 0.\]

A.4 Markov and Reciprocal Properties of \(Q^{\Pi^*}\)

Mixture of Bridges and Initialization It is an immediate observation that the mixtures of a set of bridges are also bridges: let \(Q^{z:A}\) be a set of \(A\)-bridges indexed by a variable \(z\), then \(Q^A := \int Q^{z:A} \mu(dz)\) is an \(x\)-bridge for any distribution \(\mu\) on \(z\).

A special case is to take the mixture of the conditional bridges in (5) starting from different deterministic initialization, which shows that we can obtain a valid \(x\)-bridge by equipping the same drift in (5) with essentially any initialization. Hence, the choices of the drift force and initialization in \(Q^{x}\) can be completely decoupled.

Proposition A.6. Let \(\hat{Q}\) is a path measure and \(\Omega_x\) is the set of \(z\) for which \(Q^{z_0,x}(\cdot) := \hat{Q}(\cdot|Z_T = x, Z_0 = z_0)\) exists. Then \(Q^x := \int Q^{z_0,x} \mu(dz_0 | x)\) is an \(x\)-bridge, for any distribution \(\mu\) on \(\Omega \times \Omega\).

[Proof of Proposition A.6] This is an obvious result. We have \(Q^{z_0,x}(Z_T = x) = 1\) by the definition of conditioned processes. Hence \(Q^x(Z_T = x) = \int Q^{z_0,x}(Z_T = x)\mu(dz_0 | x) = \int \mu(dz_0 | x) = 1.\)

Markov and Reciprocal Properties of \(Q^{\Pi^*}\) If \(Q^x\) is constructed as \(Q^x = Q(\cdot | Z_T = x)\), it is easy to see that \(Q^{\Pi^*} := \int Q^x(\cdot)\Pi^*(dx)\) is Markov iff \(Q\) is Markov.

Proposition A.7. Assume \(Q^x = Q(\cdot | Z_T = x)\) and \(\pi^*(z) := \frac{d\Pi^*}{dQ^x}(z)\) exists and is positive everywhere. Then \(Q^{\Pi^*}\) is Markov, iff \(Q\) is Markov.

[Proof of Proposition A.7] If \(Q^x = Q(\cdot | Z_T = x)\), we have from the definition of \(Q^{\Pi^*}\):

\[
Q^{\Pi^*}(Z) = Q(Z|Z_T)\Pi^*(Z_T) = Q(Z)\pi^*(Z_T),
\]
where \( \pi^*(Z_T) = \frac{d\Pi^*}{d\Pi}(Z_T) \). Therefore, \( \Pi^{\Pi^*} \) is obtained by multiplying a positive factor \( \pi^*(Z_T) \) on the terminal state \( Z_T \) of \( \Pi \). Hence \( \Pi^{\Pi^*} \) has the same Markov structure as that of \( \Pi \).

If \( \Pi^x \) is constructed from mixtures of bridges as above, the resulting \( \Pi^{\Pi^*} \) is more complex. In fact, simply varying the initialization \( \mu \) in Proposition (A.6) can change the Markov structure of \( \Pi^{\Pi^*} \).

**Proposition A.8.** Take \( \Pi^x \) to be the dynamics in (7) initialized from \( Z_0 \sim \mathcal{N}(0, \nu_0) \). Assume \( \sigma_i > 0 \), \( \forall i \in [0, T] \). Then \( \Pi^{\Pi^*} \) is Markov only when \( v_0 = 0 \) or \( v_0 = +\infty \).

[Proof of Proposition A.8] When taking \( \Pi^x \) to be the dynamics (7) initialized from \( Z_0 \sim \mu_0 = \mathcal{N}(0, \nu_0) \), we have \( \Pi^x = \int \mu_0(dz_0)\Pi^{z_0,x} \), where \( \Pi^{z_0,x} = \tilde{\Pi}(\cdot | Z_0 = z_0, Z_T = x) \) with \( \tilde{\Pi} \) following Brownian motion \( dZ_t = dW_t \). Hence, we can write \( \Pi^{\Pi^*} = \Pi(dZ_T) = \Pi(dZ_T)r(Z_0, Z_T) \), where \( r(z_0, z_T) = \frac{d\Pi^{\Pi^*}}{d\Pi^{z_0,x}}(z_0, z_T) \). From Léonard et al. (2014), \( \Pi^{\Pi^*} \) is Markov iff \( r(x, z_0) = f(x)g(z_0) \) for some \( f \) and \( g \), which is not the case except the degenerated case \( (v_0 = 0 \text{ and } v_0 = +\infty) \) because \( \tilde{\Omega}_{0,1} \) is not factorized.

On the other hand, when \( v_0 = 0 \), we have that \( \Pi^x = \Pi(\cdot | Z_T = x) \) is the standard Brownian bridge and hence \( \Pi^{\Pi^*} \) is Markov following Proposition A.7. When \( v_0 = +\infty \), as the case of SMLD, \( \Pi^{\Pi^*} \) is the law of \( Z_t = Z_{T-t} \) with \( dZ_t = dW_t \) and \( \tilde{\Omega}_0 \sim \Pi^*, \) which is also Markov.

The right characterization of \( \Pi^{\Pi^*} \) from Proposition (A.6) involves reciprocal processes (Léonard et al., 2014).

**Definition A.9.** A process \( Z \) with law \( \Pi_\cdot \) on \( [0, T] \) is said to be reciprocal if it can be written into \( \Pi = \int \tilde{\Pi}^{z_0,x}(\cdot | d\pi^{x}) + \Pi_\cdot \Pi_\cdot \) \( \tilde{\Pi} \) is a Markov process and \( \Pi^{z_0,x} = \Pi(\cdot | Z_0 = z_0, Z_T = x) \), and \( \mu \) is a probability measure on \( \Omega \times \Omega \).

**Proposition A.10.** \( \Pi^{\Pi^*} \) is reciprocal iff \( \Pi^x = \int \tilde{\Pi}^{z_0,x}(d\pi^{x}) \) for a Markov \( \tilde{\Pi} \) and distribution \( \pi^{x} \).

[Proof of Proposition A.10] Note that \( \Pi^{\Pi^*}(\cdot) = \int \pi(dx)\Pi^x(\cdot) = \int \pi(dx)\mu(dz_0 | x)\tilde{\Pi}^{z_0,x}(\cdot) \).

Hence if \( \tilde{\Pi} \) is Markov, \( \Pi^{\Pi^*} \) is reciprocal by Definition A.9.

On the other hand, if \( \Pi^{\Pi^*} \) is reciprocal, we have \( \Pi^{\Pi^*}(\cdot) = \int \tilde{\Pi}^{z_0,x}(\cdot)\mu(dz_0, dx) \) for some Markov process \( \tilde{\Pi} \) and probability measure \( \mu \) on \( \Omega \times \Omega \). In this case, we have \( \Pi^x(\cdot) = \Pi^{\Pi^*}(\cdot | Z_T = x) = \int \tilde{\Pi}^{z_0,x}(\cdot)\mu(dz_0 | x) \), assuming it exits.

Intuitively, a reciprocal process can be viewed as connecting the head and tail of a Markov chain, yielding a single loop structure. A characteristic property is \( \Pi(X_{[s,t]} \in A | Z_{[0,s]}, Z_{[t,T]} = \Pi(X_{[s,t]} \in A | Z_s, Z_t) \), where \( A \) is any event that occur between time \( s \) and \( t \). Solutions of the Schrodinger bridge problems are reciprocal processes (Léonard et al., 2014).

### A.5 Practical Algorithm and Error Analysis

In practice, we need to introduce empirical and numerical approximations in both training and inference phases. Denote by \( \tau = \{\tau_i\}_{i=1}^{K+1} \) a grid of time points with \( 0 = \tau_1 < \tau_2 \ldots < \tau_{K+1} = T \). During training, we minimize an empirical and time-discretized surrogate of \( \mathcal{L}(\theta) \) as follows

\[
\hat{\mathcal{L}}(\theta) = \frac{1}{n} \sum_{i=1}^{n} \ell(\theta; Z^{(i)}, \tau^{(i)}) + \ell(\theta; Z, \tau) := -\log p^0_\theta(Z_0) + \frac{1}{2K} \sum_{k=1}^{K} \Delta(\theta; Z, \tau_k),
\]

where \( \Delta(\theta; Z, \tau) := \|\sigma^{-1}(Z_t, t)(s^\theta(Z_t, t) - \eta^\theta(Z_{[0,t]}, t))\|^2 \), and \( \{Z^{(i)}\} \) is drawn from \( \Pi^{\Pi^*} \), and \( \tau^{(i)} \) can be either a deterministic uniform grid of \([0, T]\), i.e., \( \tau^{(i)} = \{i/K\}_{i=0}^{K} \), or drawn i.i.d. uniformly on \([0, T]\) (see e.g., Song et al. (2020b); Ho et al. (2020)). A subtle problem here is that the
variance of $\Delta(\theta; Z, t)$ grows to infinite as $t \uparrow T$. Hence, we should not include $\Delta(\theta; Z, T)$ at the end point $\tau^{K+1} = T$ into the sum in the loss $\ell(\theta, Z, \tau)$ to avoid variance exploding.

In the sampling phase, the continuous-time model $P^\theta$ should be approximated numerically. A standard approach is the Euler-Maruyama method, which simulates the trajectory on a time grid $\tau$ by
\[
\hat{Z}_{\tau_k+1} = \hat{Z}_{\tau_k} + \epsilon_k s^\theta_\tau(\hat{Z}_{\tau_k}, \tau_k) + \sqrt{\epsilon_k} \sigma(\hat{Z}_{\tau_k}, \tau_k) \xi_k, \quad \epsilon_k = \tau_{k+1} - \tau_k, \quad \xi_k \sim \mathcal{N}(0, I_d),
\]
where $\epsilon_k$ is equivalent to the Euler method above, and
\[
\xi_k \sim \mathcal{N}(0, I_d), \quad k = 1, 2, \ldots, K.
\]
The final output is $\hat{Z}_T$. The following result shows the KL divergence between $P^\theta$ and the distribution of $\hat{Z}_T$ can be bounded by the sum of the step size and the expected optimality gap $\mathbb{E}[L(\theta) - L(\theta^*)]$ of the time-discretized loss in (16).

A.5.1 Time-Discretization Error Analysis (Proposition A.11)

**Proposition A.11.** Assume $\Omega = \mathbb{R}^d$ and $\sigma(z, t) = \sigma(t)$ is state-independent. Take the uniform time grid $\tau_{\text{unif}} \equiv \{t_k\}_{k=0}^K$ with step size $\epsilon = T/K$ in the sampling step (17). Assume $\sigma(t) > c > 0$, $\forall t \in [0, T]$, and $\sigma(t)$ is piecewise constant w.r.t. time grid $\tau_{\text{unif}}$. Let $L_c(\theta) = \mathbb{E}_{Z \sim Q^\theta}[\ell(\theta; Z, \tau_{\text{unif}})]$. Let $P^\theta_T$ be the distribution of the resulting sample $\hat{Z}_T$. Let $\theta^*$ be an optimal parameter satisfying (14). Assume $C_0 = \sup_{z,t} \left( \|s^\theta(z, t)\|^2 / (1 + \|z\|^2), \|\sigma^2(z, t), \mathbb{E}_{s \sim P}(\|Z_0\|^2) \right) < +\infty$, and
\[
\|s^\theta(z, t) - s^\theta(z', t')\|^2 \leq L \left( \|z - z'\|^2 + |t - t'| \right) \text{ for } \forall z, z' \in \mathbb{R}^d \text{ and } t, t' \in [0, T].
\]

\[
\sqrt{K} L(\Pi^* || P^\theta_T) \leq \sqrt{L_c(\theta) - L_c(\theta^*)} + O(\sqrt{\epsilon}).
\]

We provide the analysis and proof for Proposition A.11 in the following text.

**Proposition A.12.** Assume $\Omega = \mathbb{R}^d$ and $\sigma(z, t) = \sigma(t)$ is state-independent and $\sigma(t) > c > 0$, $\forall t \in [0, T]$. Take the uniform time grid $\tau_{\text{unif}} \equiv \{t_k\}_{k=0}^K$ with step size $\epsilon = T/K$ in the sampling step (17). Let $L_c(\theta) = \mathbb{E}_{Z \sim Q^\theta}[\ell_c(\theta; Z)]$ with
\[
\ell_c(\theta, Z_t) = -\log p^\theta_0(Z_0) + \frac{1}{2 \kappa} \sum_{k=1}^K \|s^{-1}(\sigma(Z_{t_k}, t_k)) - \eta Z^c_{\tau_k}(Z_{[0,t_k]}, t_k)\|^2,
\]
where $\epsilon > 0$ is a step size with $T = K \epsilon$ and $t_k = (k-1) \epsilon$, and $\sigma^2 := (t_{k+1} - t_k)^{-1} \int_{t_k}^{t_{k+1}} \sigma(t)^2 dt$. Let $P^\theta_T$ be the distribution of the sample $\hat{Z}_T$ resulting from the following Euler method:
\[
Z_{t_{k+1}} = Z_{t_k} + \epsilon s^\theta(Z_{t_k}, t_k) + \sqrt{\epsilon} \sigma \xi_k,
\]
where $\xi_k \sim \mathcal{N}(0, I_d)$ is the standard Gaussian noise in $\mathbb{R}^d$. Let $\theta^*$ be an optimal parameter satisfying (14). Assume $C_0 = \sup_{z,t} \left( \|s^\theta(z, t)\|^2 / (1 + \|z\|^2), \|\sigma^2(z, t), \mathbb{E}_{s \sim P}(\|Z_0\|^2) \right) < +\infty$, and $s^\theta$ satisfies $\|s^\theta(z, t) - s^\theta(z', t')\|^2 \leq L \left( \|z - z'\|^2 + |t - t'| \right)$ for $\forall z, z' \in \mathbb{R}^d$ and $t, t' \in [0, T]$. Then we have
\[
\sqrt{K} L(\Pi^* || P^\theta_T) \leq \sqrt{L_c(\theta) - L_c(\theta^*)} + O(\sqrt{\epsilon}).
\]

[Proof of Proposition A.11] This is the result of Lemma A.13 below by noting that the $\hat{P}^\theta$ there is equivalent to the Euler method above, and $L_c(\theta) - L_c(\theta^*) \leq \hat{L}_c(\theta) - \hat{L}_c(\theta^*)$ (because $\sigma^2 = \int_{t_k}^{t_{k+1}} \sigma(t)^2 dt^{-1} \leq (t_{k+1} - t_k)^{-1} \int_{t_k}^{t_{k+1}} \sigma(t)^2 dt$).

**Lemma A.13.** Let $h$ be a step size and $\epsilon = T/K$ for a positive integer $K$. For each $t \in [0, \infty)$, denote by $[t]_\epsilon = \max\{k : k \in \mathbb{N} \cap [0, \epsilon]\}$. Assume
\[
Q^H_T : \quad \frac{dZ_t}{dt} = \eta Z_{[0,t], t} dt + \sigma(Z_t, t) dW_t, \quad Z_0 \sim Q_0
\]
\[
P^\theta : \quad \frac{dZ_t}{dt} = s^\theta(Z_t, t) dt + \sigma(Z_t, t) dW_t, \quad Z_0 \sim P_0
\]
\[
\hat{P}^\theta : \quad \frac{dZ_t}{dt} = s^\theta(Z_{[t], t}) dt + \sigma(Z_t, t) dW_t, \quad Z_0 \sim \hat{P}_0.
\]
where $\mathbb{P}^0$ is the Markovianization of $Q^{11}$, and $\tilde{\mathbb{P}}^0$ is a discretized version of $\mathbb{P}^0$. Define

$$\tilde{\mathcal{L}}_\epsilon(\theta) = \mathbb{E}_{Q^{11}} \left[ -\log p^0_0(Z_0) + \frac{1}{2} \int_0^T \| \sigma^{-1}(Z_t, t)(s^0(Z_{[t]}, [t], \epsilon)) - \eta^Z_{\tilde{\mathcal{E}}_r}(Z_{[0,t]}, [t], \epsilon) \|_2^2 \, dt \right].$$

Assume the conditions of Lemma A.17 holds for $\mathbb{P}^0$, and $\sigma(z,t) \geq c > 0$ for all $z, t$, and $s^0$ satisfies $\| s^0(z, t) - s^0(z', t') \|^2_2 \leq L \left( \| z - z' \|^2 + |t - t'| \right)$ for $z, z' \in \mathbb{R}^d$ and $t, t' \in [0,T]$. Then

$$\sqrt{K\mathcal{L}(\mathbb{P}^{\theta*} \| \tilde{\mathbb{P}}^0)} \leq \sqrt{\tilde{\mathcal{L}}_\epsilon(\theta) - \tilde{\mathcal{L}}_\epsilon(\theta^*)} + C\sqrt{\epsilon},$$

where $C$ is a constant that is independent of $\epsilon$.

[Proof] Define $\| f \|^2_{Q,\sigma} = \mathbb{E}_{Q \sim Q} \left[ \int_0^T \| \sigma(Z_t) f(Z_t) \|^2 \right]$ for convenient notation. Let $s^0_t(Z, t) = s^0(Z_{[t]}, [t], \epsilon)$, and $\eta_\epsilon = \eta^Z_{\tilde{\mathcal{E}}_r}(Z_{[0,t]}, [t], \epsilon)$.

$$K\mathcal{L}(\mathbb{P}^{\theta*} \| \tilde{\mathbb{P}}^0) = \mathbb{E}_{Q \sim Q} \left[ \int_0^T \| \sigma(Z_t) f(Z_t) \|^2 \right] = \mathbb{E}_{Q \sim Q} \left[ \int_0^T \| \sigma(Z_t) f(Z_t) \|^2 \right] = \mathbb{E}_{Q \sim Q} \left[ \int_0^T \| \sigma(Z_t) f(Z_t) \|^2 \right].$$

where $\omega > 0$ is any positive number and

$$I_1 := \frac{1}{1 + \omega} K\mathcal{L}(\mathbb{P}^{\theta*} \| \tilde{\mathbb{P}}^0) + \frac{1}{2} \| s^0 - s^0_{\epsilon} \|^2_{p_{\theta^*},\sigma}$$

$$= \frac{1}{1 + \omega} K\mathcal{L}(\mathbb{P}^{\theta*} \| \tilde{\mathbb{P}}^0) + \frac{1}{2} \| s^0 - s^0_{\epsilon} \|^2_{p_{\theta^*},\sigma}$$

$$\leq \frac{1}{1 + \omega} K\mathcal{L}(\mathbb{P}^{\theta*} \| \tilde{\mathbb{P}}^0) + \frac{1}{2} \| s^0 - s^0_{\epsilon} \|^2_{Q^{11},\sigma},$$

$$\leq \frac{1}{1 + \omega} K\mathcal{L}(\mathbb{P}^{\theta*} \| \tilde{\mathbb{P}}^0) + \frac{1}{2} \| s^0 - s^0_{\epsilon} \|^2_{Q^{11},\sigma}$$

$$\leq \frac{1}{1 + \omega} K\mathcal{L}(\mathbb{P}^{\theta*} \| \tilde{\mathbb{P}}^0) + \frac{1}{2} \left( \| s^0 - s^0_{\epsilon} \|^2_{Q^{11},\sigma} - \| s^0_{\epsilon} - \eta^Z_{\tilde{\mathcal{E}}_r} \|^2_{Q^{11},\sigma} \right) \quad \text{//Lemma A.5}$$

$$= \tilde{\mathcal{L}}_\epsilon(\theta) - \tilde{\mathcal{L}}_\epsilon(\theta^*),$$

and

$$I_2 := \frac{1}{2} \| s^0 - s^0_{\epsilon} \|^2_{p_{\theta^*},\sigma}$$

$$\leq \frac{L}{2c^2} E_{\theta^*} \left[ \int_0^T \sigma^{-2}(Z_t, t) \left( \| Z_t - Z_{[t],\epsilon} \|^2 + (t - [t],\epsilon) \right) dt \right]$$

$$\leq \frac{L}{2c^2} E_{\theta^*} \left[ \int_0^T \left( \| Z_t - Z_{[t],\epsilon} \|^2 + (t - [t],\epsilon) \right) dt \right]$$

$$\leq \frac{L}{2c^2} \left( C_{\theta^*} + 1 \right) \int_0^T (t - [t],\epsilon) dt \quad \text{//Lemma A.17}$$

$$= \frac{L}{2c^2} \left( C_{\theta^*} + 1 \right) \frac{T\epsilon}{2}. \quad \text{//Lemma A.15},$$

where $C_{\theta^*}$ is a constant depending on $\mathbb{P}^{\theta^*}$ that comes from Lemma A.17. Hence

$$K\mathcal{L}(\mathbb{P}^{\theta*} \| \tilde{\mathbb{P}}^0) \leq \inf_{\omega \geq 0} \left( (1 + \omega) I_1 + (1 + 1/\omega) I_2 \right)$$

$$= \left( \sqrt{I_1} + \sqrt{I_2} \right)^2$$

$$\leq \left( \sqrt{\tilde{\mathcal{L}}_\epsilon(\theta) - \tilde{\mathcal{L}}_\epsilon(\theta^*)} + \frac{1}{2} \sqrt{\frac{L}{c^2} \left( C_{\theta^*} + 1 \right) \frac{T\epsilon}{2}} \right)^2.$$
Lemma A.14. For any $a, b \in \mathbb{R}^d$, and $\omega \geq 0$,
$$\|a + b\|_2^2 \leq (1 + \omega) \|a\|_2^2 + (1 + 1/\omega) \|b\|_2^2.$$  

[Proof]
$$(1 + \omega) \|a\|_2^2 + (1 + 1/\omega) \|b\|_2^2 \geq \|a\|_2^2 + \|b\|_2^2 + 2a^\top b = \|a + b\|_2^2.$$  

Lemma A.15. Assume $T \geq 0$, $\epsilon \geq 0$ and $T/\epsilon \in \mathbb{N}$. We have
$$\int_0^T (t - \lfloor t \rfloor \epsilon) \, dt = \frac{T \epsilon}{2}.$$  

[Proof]
$$\int_0^T (t - \lfloor t \rfloor \epsilon) \, dt = \sum_{k=0}^{K-1} \int_0^{\epsilon} (hk + x - hk) \, dx$$
$$= \sum_{k=0}^{K-1} \int_0^{\epsilon} x \, dx$$
$$= K \epsilon^2 / 2$$
$$= T \epsilon / 2.$$  

Lemma A.16 (Grönwall’s inequality). Let $I$ denote an interval of the real line of the form $[a, \infty)$ or $[a, b]$ or $[a, b)$ with $a < b$. Let $\alpha, \beta$ and $u$ be real-valued functions defined on $I$. Assume that $\beta$ and $u$ are continuous and that the negative part of $\alpha$ is integrable on every closed and bounded subinterval of $I$.

(a) If $\beta$ is non-negative and if $u$ satisfies the integral inequality
$$u(t) \leq \alpha(t) + \int_a^t \beta(s) u(s) \, ds, \quad \forall t \in I,$$
which is true if
$$u'(t) \leq \alpha'(t) + \beta(t) u(t),$$
then
$$u(t) \leq \alpha(t) + \int_a^t \alpha(s) \beta(s) \exp \left( \int_s^t \beta(r) \, dr \right) ds, \quad t \in I.$$  

(b) If, in addition, the function $\alpha$ is non-decreasing, then
$$u(t) \leq \alpha(t) \exp \left( \int_a^t \beta(s) \, ds \right), \quad t \in I.$$  

Lemma A.17. Consider
$$dX_t = b(Z_t, t) dt + \sigma(Z_t, t) dW_t, \quad X_0 = 0, t \in [0, T].$$
Assume there exists a finite constant $C_0$, such that
$$\|b(x, t)\|_2^2 \leq C_0(1 + \|x\|_2^2), \quad \forall x \in \mathbb{R}^d, \ t \in [0, T],$$
$$\text{tr}(\sigma^\top(x, t)) \leq C_0, \quad \forall x \in \mathbb{R}^d, t \in [0, T].$$

and $\mathbb{E}[\|Z_0\|_2^2] \leq C_0$.

Then for any $0 \leq s \leq t \leq T$, we have
$$\mathbb{E}[\|Z_t - Z_s\|_2^2] \leq K_{C_0,T}(t - s),$$
where $K_{C_0,T}$ is a finite constant that depends on $C_0$ and $T$.  

20
[Proof] Let $\eta = \sup_{x,t} tr(\sigma \sigma^\top(x,t))$. We have by Ito Lemma,
\[
\frac{d}{dt} E \left[\|Z_t - Z_s\|^2_2\right] = E \left[2(Z_t - Z_s)^\top(b(Z_t, t) + dW_t) + \eta\right]
= E \left[2(Z_t - Z_s)^\top b(Z_t, t) + \eta\right]
= E \left[\|Z_t - Z_s\|^2_2 + \|b(Z_t, t)\|^2 + d\right]
\leq E \left[\|Z_t - Z_s\|^2_2 + C_0(1 + \|Z_t\|^2_2) + \eta\right]
\leq (1 + 2C_0) E \left[\|Z_t - Z_s\|^2_2 + \eta + C_0(1 + 2E[\|Z_s\|^2_2])\right].
\]

Using Gronwall’s inequality,
\[
E \left[\|Z_t - Z_s\|^2_2\right] \leq (t - s)(\eta + C_0(1 + 2E[\|Z_0\|^2_2])) \exp((t - s)(1 + 2C_0)).
\]

Taking $s = 0$ yields that
\[
E \left[\|Z_t - Z_0\|^2_2\right] \leq t(\eta + C_0(1 + 2E[\|Z_0\|^2_2])) \exp(t(1 + 2C_0)).
\]

Hence
\[
E[\|Z_t\|^2_2] \leq 2E \left[\|Z_t - Z_0\|^2_2\right] + 2E \left[\|X_0\|^2_2\right]
\leq 2(t(\eta + C_0(1 + 2E[\|Z_0\|^2_2])) \exp(t(1 + 2C_0)) + 2E[\|Z_0\|^2_2]
\leq 4T(C_0 + C_0^2) \exp(T(1 + 2C_0)) + 2C_0.
\]

Therefore,
\[
E \left[\|Z_t - Z_s\|^2_2\right]
\leq (t - s)(\eta + C_0(1 + 2E[\|Z_s\|^2_2])) \exp((t - s)(1 + 2C_0))
\leq C(t - s),
\]

where
\[
C = (2C_0 + 4C_0^2 + 8C_0T(C_0 + C_0^2) \exp(T(1 + 2C_0))) \exp(T(1 + 2C_0)).
\]

**A.5.2 Statistical Error Analysis (Proposition A.18)**

To provide a simple analysis of the statistical error, we assume that $\hat{\theta}_n = \arg\min_{\theta} \hat{L}(\theta)$ is an asymptotically normal M-estimator of $\theta^*$ following classical asymptotic statistics (Van der Vaart, 2000), with which we can estimate the rate of the excess risk $\mathcal{L}_e(\hat{\theta}_n) - \mathcal{L}_e(\theta^*)$ and hence the KL divergence.

**Proposition A.18.** Assume the conditions in Proposition (A.11). Assume $\hat{\theta}_n = \arg\min_{\theta} \hat{L}_e(\theta)$ with $\hat{L}_e(\theta) = \sum_{i=1}^n \ell(\theta; Z(i), \pi^{unif})/n, Z(i) \sim \mathcal{Q}^{\Pi^*}$. Take $\mathcal{Q}^\theta$ to be the standard Brownian bridge $dZ_t^\theta = \frac{z - z_t^\theta}{\sqrt{t}} dt + dW_t$ with $Z_0 \sim \mathcal{N}(0, v_0)$ and $v_0 > 0$. Assume $\sqrt{n}(\hat{\theta}_n - \theta^*) \xrightarrow{d} \mathcal{N}(0, \Sigma_n)$ as $n \to +\infty$, where $\Sigma_n$ is the asymptotic covariance matrix of the $M$ estimator $\hat{\theta}_n$. Assume $\mathcal{L}_e(\theta)$ is second order continuously differentiable and strongly convex at $\theta^*$. Assume $\Pi^*$ has a finite covariance and admits a density function $\pi$ that satisfies $\sup_{t \in [0, T]} E_{\mathcal{Q}^{\Pi^*}} \left[\|\nabla \pi^\theta(Z_t, t)\|^2(1 + \|\nabla \log \pi(Z_T)\|^2 + \text{tr}(\nabla^2 \log \pi(Z_T)))\right] < +\infty$. We have
\[
E \left[\sqrt{\mathcal{K}\mathcal{L}(\Pi^*) \left\|\mathbb{P}^{\hat{\theta}_n, \tau}_T\right\|}\right] = O\left(\sqrt{\frac{\log(1/\epsilon)}{n} + \sqrt{\epsilon}}\right). \tag{18}
\]
The expectation in Eq. (18) is w.r.t. the randomness of $\hat{\theta}_n$. The log$(1/\epsilon)$ factor shows up as the sum of a harmonic series as the variance of $\Delta(\theta; Z, t)$ grows with $O(1/(T - t))$ when $t \uparrow T$. Taking $\epsilon = 1/n$ yields $K\mathcal{L}(\Pi^* \mid \mathbb{P}^*_T) = O(\log n/n)$. If we want to achieve $K\mathcal{L}(\Pi^* \mid \mathbb{P}^*_T) = O(\eta)$, it is sufficient to take $K = T/\epsilon = O(1/\eta)$ steps and $n = O(\log(1/\eta)/\eta)$ data points.

[Proof of Proposition A.18] Let

$$\theta^* = \arg\min_{\theta} \mathcal{L}_e(\theta) := E_{Z \sim \mathbb{Q}^n} [\ell(\theta; Z)], \quad \hat{\theta}_n = \arg\min_{\theta} \hat{\mathcal{L}}_e(\theta) := \frac{1}{n} \sum_{i=1}^n \ell(\theta; Z^{(i)})$$

where $\{Z^{(i)}\}_{i=1}^n$ is drawn i.i.d. from $\mathbb{Q}^\Pi^*$. We assume that $\hat{\theta}_n$ is an asymptotically normal M-estimator, in which case we have

$$\sqrt{n}(\theta_n - \theta^*) \xrightarrow{d} \mathcal{N}(0, \Sigma_*)$$

where

$$\Sigma_* = H_*^{-1} V_* H_*^{-1}, \quad H_* = E_{Z \sim \mathbb{Q}^n} \left[\nabla^2_{\theta\theta} \ell(\theta^*; Z)\right], \quad V_* = E[\nabla_\theta \ell(\theta^*; Z) \nabla_\theta \ell(\theta^*; Z)^\top]$$

and

$$nE[(\mathcal{L}(\hat{\theta}_n) - \mathcal{L}(\theta^*)) \approx \left[\frac{1}{2} \sqrt{n}(\theta_* - \hat{\theta}_n)^\top H_* \sqrt{n}(\theta_* - \hat{\theta}_n)\right] \approx \frac{1}{2} \text{tr}(H_*^{-1} V_*)$$

where $f \asymp g$ denotes that $f - g = o(1)$. We now need to bound $\text{tr}(H_*^{-1} V_*)$. Combining the results in Lemma A.19 and Lemma A.23, we have when $t_k = (k - 1)\epsilon$ and $T = K\epsilon$,

$$\text{tr}(H_*^{-1} V_*) = O\left(1 + \frac{1}{K} \sum_{k=1}^K \frac{1}{T - t_k}\right) = O(1 + \log(1/\epsilon)).$$

Hence,

$$E[\sqrt{K\mathcal{L}(\Pi^* \mid \mathbb{P}^*_T)}] = O\left(E[\sqrt{\mathcal{L}(\hat{\theta}_n) - \mathcal{L}(\theta^*))} + \sqrt{\epsilon}\right)$$

$$= O\left(\sqrt{E[\mathcal{L}(\hat{\theta}_n) - \mathcal{L}(\theta^*))] + \sqrt{\epsilon}\right)$$

$$= O\left(\sqrt{\log(1/\epsilon + 1)/n} + \sqrt{\epsilon}\right).$$

Lemma A.19. Assume the conditions in Proposition A.18. Define

$$I_0 = E_{Z \sim \mathbb{Q}^n} \left[\|\nabla \log p_{0^*}(Z_0)\|^2\right], \quad I_k = E_{Z \sim \mathbb{Q}^n} \left[\|\nabla_\theta s^{\theta^*}(Z_{t_k}, t_k)\|^2 \text{tr} (\text{cov}(\eta^{2T}(Z_{[0,t_k]}, t_k) | Z_{t_k})) \right],$$

for $\forall k = 1, \ldots, K$. Then

$$\text{tr}(H_*^{-1} V_*)^{1/2} \leq \frac{1}{\lambda_{\text{min}}(H_*)^{1/2}} \left(\frac{I_0^{1/2}}{2} + \left(\frac{1}{K} \sum_{k=1}^K I_k\right)^{1/2}\right).$$

[Proof] From Lemma A.21, $\text{tr}(H_*^{-1} V_*) \leq (\lambda_{\text{min}}(H_*))^{-1} \text{tr}(V_*)$. Hence we just need to bound $\text{tr}(V_*)$. 

22
\[ \text{tr}(V_s)^{1/2} = E_{z \sim Q(z)} \left[ \| \nabla \theta \ell(\theta^*, Z) \|^2 \right]^{1/2} \]

\[ \leq E_{z \sim Q(z)} \left[ \| \nabla \theta \ell(\theta^*, Z) \|^2 \right]^{1/2} + \frac{1}{K} \sum_{k=1}^{K} E_{z \sim Q(z)} \left[ \| \nabla \theta s(\theta^*, Z_{t_k}, t_k) (s(\theta^*, Z_{t_k}, t_k) - \eta^T (Z_{[0,t_k]}, t_k)) \|^2 \right]^{1/2} \]

\[ \leq E_{z \sim Q(z)} \left[ \| \nabla \theta \ell(\theta^*, Z) \|^2 \right]^{1/2} + \frac{1}{K} \sum_{k=1}^{K} E_{z \sim Q(z)} \left[ \| \nabla \theta s(\theta^*, Z_{t_k}, t_k) \|^2 \| (s(\theta^*, Z_{t_k}, t_k) - \eta^T (Z_{[0,t_k]}, t_k)) \|^2 \right]^{1/2} \]

\[ = E_{z \sim Q(z)} \left[ \| \nabla \theta \ell(\theta^*, Z) \|^2 \right]^{1/2} + \frac{1}{K} \sum_{k=1}^{K} E_{z \sim Q(z)} \left[ \| \nabla \theta s(\theta^*, Z_{t_k}, t_k) \|^2 \right]^{1/2} \]

\[ = I_0^{1/2} + \frac{1}{K} \sum_{k=1}^{K} I_k^{1/2} \]

\[ \leq I_0^{1/2} + \sqrt{\frac{1}{K} \sum_{k=1}^{K} I_k} \]

**Lemma A.20.** Assume the results in Lemma A.19 and Lemma A.23 hold. Assume

\[ \max_{k \in 1, \ldots, K} E_{z \sim Q(z)} \left[ \left( \left\| \nabla \theta s(\theta^*, Z_{t_k}, t_k) \right\|^2 + \| \nabla \log \pi^*(Z_T) \|^2_{2} + \text{tr}(\nabla^2 \log \pi^*(Z_T)) \right) \right] < +\infty, \]

Then for \( k = 1, \ldots, K \), we have \( I_k = O \left( \frac{1}{T-t_k} + 1 \right) \).

[Proof] It is a direct application of (20).

**Lemma A.21.** Let \( A \) and \( B \) be two \( d \times d \) positive semi-definite matrices. Then \( \text{tr}(AB) \leq \lambda_{\text{max}}(A) \text{tr}(B) \).

[Proof] Write \( A = \sum_{i=1}^{d} \lambda_i u_i u_i^T \) where \( \lambda_i \) and \( u_i \) is the \( i \)-th eigenvalue and eigenvectors of \( A \), respectively. Then

\[ \text{tr}(AB) = \text{tr} \left( \sum_{i=1}^{d} \lambda_i u_i u_i^T B u_i \right) \leq \lambda_{\text{max}}(A) \text{tr} \left( \sum_{i=1}^{d} u_i^T B u_i \right) = \lambda_{\text{max}}(A) \text{tr}(B). \]

**Controlling the Conditional Variance of the Regression Problem**  Assume \( Q^\theta \) is the standard Brownian bridge:

\[ Q^\theta: \quad dZ_t^\theta = \frac{x - Z_t^\theta}{T-t} \, dt + dW_t, \quad Z_0 \sim \mathcal{N}(0, v_0). \]  

(19)

In this case, the (ideal) loss function is

\[ \mathcal{L}(\theta) = -E_X \mathbb{E}_{\theta^*, \theta \sim Q^\theta} \left[ \log p^\theta_0(Z_0) + \frac{1}{2} \int_0^T \| s^\theta(Z_t, t) - Y_t \|^2 \, dt \right], \quad \text{where} \ Y_t = \frac{X - Z_t}{1-t}. \]

The second part of the loss is a least square regression for predicting \( Y_t = \eta^X(Z_t, t) \) with \( s^\theta(Z_t, t) \). The conditioned variance \( \text{cov}(Y_t | Z_t) \) is an important factor that influences the error of the regression problem. We now show that \( \text{tr}(\text{cov}(Y_t | Z_t)) = O(1/T-t) \) which means that it explodes to infinity when \( t \uparrow T \).

First, note that \( \text{tr}(\text{cov}(Y_t | Z_t)) = \frac{1}{T-t} \text{tr}(\text{cov}(X | Z_t)) \). Using the estimate in Lemma A.23, we have

\[ \text{tr}(\text{cov}(Y_t | Z_t)) = O \left( \frac{1}{T-t} + E \left[ \| \nabla_s \log \pi^*(X) \|^2_{2} + \text{tr} \left( \nabla^2 \log \pi^*(X) \right) \bigg| Z_t \right] \right). \]  

(20)
Lemma A.22. For the standard Brownian bridge in (19), we have
\[ Z_t^x \sim N\left( \frac{t}{T} x, \frac{(T-t)^2}{T^2} \right). \]

[Proof] Let \( Z_t^{z_0,x} \) be the same process that is initialized from \( Z_0^{z_0,x} = z_0 \). We have from the textbook result regarding Brownian bridge that we can write \( Z_t^{z_0,x} = \frac{t x + (T-t) z_0}{T} + T^{-1/2} \xi_t \) where \( \xi_t \) is some standard Gaussian random variable. The result follows directly as \( Z_t^x = Z_t^{z_0,x} \) with \( Z_0^x \sim N(0, v_0) \).

Lemma A.23. Let \( \pi^* \) be the density function \( \Pi^* \) on \( \mathbb{R}^d \) whose covariance matrix exists. When \( X \sim \Pi^* \) and \( Z \sim Q_X \) from (19) with \( v_0 > 0 \). Then the density function \( \rho_t(x|z) \) of \( X|Z_t = z_t \) satisfies
\[ \rho_t(x|z) \propto \pi^*(x) \exp \left( -\frac{\| \frac{T}{T} z - x \|^2}{2\left( \frac{T}{T-t} + v_0 \frac{(T-t)^2}{t} \right)} \right). \]

[Proof] We know that \( X \sim \Pi^* \) and \( Z_t^x|X \sim \mathcal{N}(t/T X, w_t) \). Hence, (21) is a direct result of Bayes rule. Then Lemma A.25 gives
\[ \text{tr}(\text{cov}_{\rho_t}(x|z)) = w_t d + w_t^2 \mathbb{E}_{\rho_t} \left[ \| \nabla_x \log \pi^*(x) \|^2 + \text{tr}(\nabla^2 \log \pi^*(x)) | z \right]. \]

On the other hand,
\[ \rho_t(x|z) \propto \pi^*(x) \exp \left( -\frac{1}{2w_t} \| x \|^2 + \frac{T}{t w_t} z^\top x \right), \]

When \( t \to 0 \), we have \( 1/w_t \to 0 \) and \( T/(t w_t) \to 0 \). Hence, \( \rho_t(x|z) \) converges to \( \pi^*(x) \) as \( t \to 0 \), as a result, \( \text{tr}(\text{cov}_{\rho_t}(x|z)) \to \text{tr}(\text{cov}_{\pi^*}(x)) < +\infty \). Therefore, for any \( c > 0 \), there exists \( t_0 > 0 \), such that \( \text{tr}(\text{cov}_{\rho_t}(x|z)) \leq \text{tr}(\text{cov}_{\pi^*}(x)) + c \) when \( 0 < t \leq t_0 \).

Remark A.24. We need to have \( v_0 > 0 \) to ensure that \( T/(t w_t) \to 0 \) in the proof of Lemma A.23. This is purely a technical reason, for yielding a finite bound of the conditioned variance when \( t \) is close to 0. We can establish the same result when \( v_0 = 0 \) by adding the assumption that \( \max_{k \in 1, \ldots, K} \mathbb{E}_{\Pi^*} \left[ \| \nabla_{\theta} \phi^*(Z_{t_k}, t_k) \|^2 \right] < +\infty \), where \( \Pi^*_x \) is the distribution with density \( \pi^*_x(x) \propto \pi^*(x) \exp(z^\top x/T) \).

Lemma A.25. Let \( \rho(x) \propto \pi(x) \exp \left( -\alpha \frac{\| x - b \|^2}{2} \right) \) be a positive probability density function on \( \mathbb{R}^d \), where \( \alpha > 0, b \in \mathbb{R} \) and \( \log \pi \) is continuously second order differentiable. Then
\[ \text{tr}(\text{cov}_p(x)) \leq \mathbb{E}_p[\| x \|^2] = \alpha^{-1} d + \alpha^{-2} \left( \mathbb{E}_p[\| \nabla_x \log \pi(x) \|^2 + \text{tr}(\nabla^2 \log \pi(x))] \right). \]

[Proof] Let us focus on the case when \( b = 0 \) first. Stein’s identity says that
\[ \mathbb{E}_p \left[ (\nabla_x \log \pi(x) - \alpha x)^\top \phi(x) + \nabla_x^\top \phi(x) \right] = 0, \]

for a general continuously differentiable function \( \phi \) when the integrals above are finite.
Taking $\phi = x$ yields that
\[
E_p \left[ (\nabla_x \log \pi(x) - \alpha x)^\top x + d \right] = 0,
\]
which gives
\[
E_p[\|x\|^2] = \alpha^{-1}(E_p[\nabla_x \log \pi(x)^\top x] + d).
\]
On the other hand, taking $\phi(x) = \nabla_x \log \pi(x)$ yields
\[
E_p \left[ (\nabla_x \log \pi(x) - \alpha x)^\top \nabla_x \log \pi(x) + \text{tr}(\nabla^2 \log \pi(x)) \right] = 0,
\]
which gives
\[
E_p[\nabla_x \log \pi(x)^\top x] = \alpha^{-1} \left( E_p[\|\nabla_x \log \pi(x)\|^2 + \text{tr}(\nabla^2 \log \pi(x))] \right).
\]
This gives
\[
E_p[\|x\|^2] = \alpha^{-1} + \alpha^{-2} \left( E_p[\|\nabla_x \log \pi(x)\|^2 + \text{tr}(\nabla^2 \log \pi(x))] \right).
\]
For $b \neq 0$, define $\tilde{p}(x) \propto \pi(x + b) \exp \left( -\frac{d}{2} \|x\|^2 \right)$, which is the distribution of $\tilde{x} = x - b$ when $x \sim p$. Then applying the result above to $\tilde{p}$ yields
\[
\text{tr}(\text{cov}_p(x)) = \text{tr}(\text{cov}_{\tilde{p}}(x))
\leq \alpha^{-1} d + \alpha^{-2} E_{x \sim \tilde{p}} \left[ \|\nabla_x \log \pi(x + b)\|^2 + \text{tr} \left( \nabla^2 \log \pi(x + b) \right) \right]
= \alpha^{-1} d + \alpha^{-2} E_{x \sim p} \left[ \|\nabla_x \log \pi(x)\|^2 + \text{tr} \left( \nabla^2 \log \pi(x) \right) \right].
\]

A.6 Condition for $\Omega$-Bridges

We provide the proof for Proposition 2.3.

Proof of Proposition 2.3: By the formula of KL divergence between two diffusion processes, we have
\[
\mathcal{KL}(Q^\Omega \| P^\theta) = \mathcal{KL}(Q_0^\Omega \| P_0^\theta) + \frac{1}{2} E_{Z \sim Q^\Omega} \left[ \int_0^T \| f^\theta(Z_t, t) \|^2_2 dt \right] < +\infty.
\]
This means that $Q^\Omega$ and $P^\theta$ are absolutely continuous to each other, and hence have the same support. Therefore, $Q^\Omega(Z_T \in \Omega) = 1$ implies that $P^\theta(Z_T \in \Omega) = 1$.

A.7 Examples of $\Omega$-Bridges

If $\Omega$ is a product space, the integration can be factorized into one-dimensional integrals. Specifically, assume $\Omega = I_1 \times \cdots \times I_d$, then
\[
\eta^\Omega(z, t) = [\eta^i(z_i, t)]_{i=1}^d,
\]
where $\eta^i$ is the drift fore of the $I_i$-bridge, and $z_i$ is the $i$-th element of $z = [z_i]$. Therefore, it is sufficient to focus on 1D case below.

Consider the bridge process constructed from the Brownian motion in (9). If $\Omega$ is a discrete set, say $\Omega = \{e_1, \ldots, e_K\}$, we have
\[
\eta^\Omega(z, t) = \sigma_I^2 \sum_{k=1}^K \omega(e_k, z, t) \sum_{k=1}^K \omega(e_k, z, t) \frac{e_k - z}{\beta_I - \beta_t}
= \sigma_I^2 \nabla_z \log \sum_{k=1}^K \omega(e_k, z, t),
\]
Published as a conference paper at ICLR 2023

Figure 5: Generated tabular data from Mixed-Bridge.

where

\[ \omega(e_k, z, t) = \exp \left( -\frac{\|z - e_k\|^2}{2(\beta_T - \beta_t)} \right). \]

If \( \Omega = [a, b] \), we have

\[ \eta^{\Omega}(z, t) = \sigma^2 \frac{1}{\int_a^b \omega(e, z, t) de} \int_a^b \omega(e, z, t) \frac{e - z}{\beta_T - \beta_t} de \]
\[ = \sigma^2 \nabla_z \log \int_a^b \omega(e, z, t) de \]
\[ = \sigma^2 \nabla_z \log \left( F\left( \frac{z - a}{\sqrt{\beta_T - \beta_t}} \right) - F\left( \frac{z - b}{\sqrt{\beta_T - \beta_t}} \right) \right), \]

where \( F \) is the standard Gaussian CDF.

B Additional Materials of the Experiments

In our experiments, \( T = 1 \) and \( \epsilon = T/K = 1/K \). Moreover, we take the time grid by randomly sampling from \( \{i/K\}_{i=0}^{K-1} \) for the training objective Eq. (13). For evaluation, we calculate the standard evidence lower bound (ELBO) by viewing the resulting time-discretized model as a latent variable model:

\[ \mathbb{E}_{X \sim \Pi^*} [-\log \hat{p}_T^\theta(X)] \leq \mathbb{E}_{Z \sim \mathcal{Q}^*} \left[ -\log \frac{p_0^\theta(Z_0)}{q_0(Z_0)} - \sum_{k=1}^{K} \log \frac{q_{t_k+1|t_k}}{q_{t_k+1|t_k}(Z_{t_k+1} | Z_{t_k})} \right], \]

where \( t_k = (k-1)\epsilon \), and \( \hat{p}_T^\theta \) is the density function of the time-discretized version of \( \mathbb{P}^\theta \), and \( q \) is the density function of \( \mathcal{Q} \). We adopt Monte-Carlo sampling to estimate the log-likelihood. As in (Song et al., 2020b), we repeat 5 times in the test set for the estimation. For categorical/integer/grid generation, the likelihood of the last step should take the rounding into account: in practice, we have

\[ \hat{Z}_T = \text{rounding}(\hat{Z}_{t_K} + \epsilon^s(\hat{Z}_{t_K}, t_K) + \sqrt{\epsilon}c(Z_{t_k}, t_k)\xi_K, \Omega), \]

where \( \text{rounding}(x, \Omega) \) denotes finding the nearest element of \( x \) on \( \Omega \), and hence the likelihood \( \hat{p}_T^\theta_{t_K} \) of the last step should incorporate the rounding operator as a part of the model.

B.1 Generating Mixed-Type Tabular Data

In this experiment, the metrics are measured by the implementation from Synthetic Data Vault (SDV) (Patki et al., 2016). For baseline methods, we adopt their open-sourced official implementation \(^1\). For the machine learning models adopted for evaluation, logistic regression, AdaBoost and MLP, we directly use their default configuration in SDV. For the results in Table 1, we repeat the experiments with 5 different random seeds and report their standard deviation. We provide additional generated samples from Mixed-Bridge in Figure 5.

\(^1\)https://github.com/sdv-dev/CTGAN
B.2 Generating Integer-valued Point Clouds

In this experiment, we need to process point cloud data on integer grid. To prepare the data, we firstly sample 2048 points from the ground truth mesh. Then, we normalize all the point clouds to a unit bounding box. After this, we simply project the points onto grid point by rounding the coordinate to integer. The metrics in the main text, MMD, COV and 1-NNA are computed with respect to the post-processed integer-valued training point clouds. For the results in Table 2, we repeat the experiments for 3 times and report the mean of the experiments.

B.3 Generating Semantic Segmentation Maps on CityScapes

In this experiment, we set \( (\text{Noise Decay A}) \) \( \sigma^2_t = 3 \exp(-3t) \); \( (\text{Noise Decay B}) \) \( \sigma^2_t = 3(1 - t) \); \( (\text{Noise Decay C}) \) \( \sigma^2_t = 3 - 3 \exp(-3(1 - t)) \). We visualize the noise schedule in Figure 6. Note that, except for Constant Noise, all the other three processes gradually decrease the magnitude of the noise as \( t \to 1 \). For fair comparison, we use the same neural network as in Hoogeboom et al. (2021). The network is optimized with Adam optimizer with a learning rate of 0.0002. The model is trained for 500 epochs. The CityScapes dataset (Cordts et al., 2016) contains photos captured by the cameras on the driving cars. A pixel-wise semantic segmentation map is labeled for each photo. As in (Hoogeboom et al., 2021), we rescale the segmentation maps from cityscapes to \( 32 \times 64 \) images using nearest neighbour interpolation. Our training set and test set is exactly the same as that of (Hoogeboom et al., 2021) for fair comparison. For the results in Table 3, we repeat the experiments for 3 times and report the mean of the experiments. We provide more samples in Figure 9.

B.4 Discrete CIFAR10 Generation

The model is trained using the same training strategy as DDPM (Ho et al., 2020) with the code base provided in Song et al. (2020b). Specifically, the neural network is the same U-Net structure as the implementation in Song et al. (2020b). The optimizer is Adam with a learning rate of 0.0002. According to common practice (Song & Ermon, 2020; Song et al., 2020b), the training is smoothed by exponential moving average (EMA) with a factor of 0.999. We use \( K = 1000 \) and \( dt = 0.001 \) for discretizing the SDE. To account for the discretization error, after the final step, we apply rounding to the generated images to get real integer-valued images. We compare the value distribution of the generated images in Figure 8.
Figure 7: Diffusion process of one pixel (a 8-dimensional vector) in CityScapes. As $t \rightarrow 1$, 7 of the dimensions reach 0, while 1 of the dimensions reaches 1, turning the vector into a one-hot vector.

Figure 8: Final value distribution of the generated images with Bridge-Continuous and Bridge-Integer (before rounding) on CIFAR10. We only show the values in $[125.5, 130.5]$ for visual clarity. Integer-Bridge generates discrete values.
Figure 9: Additional samples from real data, Constant Noise, Noise decay A, Noise decay B and Noise decay C.