
Likelihood training of Schrödinger bridge using Forward-backward SDEs theory

Benjamin Maurel
MVA - ENSAE
benjamin.maurel@ensae.fr

Jérémie Stym-Popper
MVA - ENSAE
jeremie.stym-popper@ensae.fr

Gabriel Watkinson
MVA - ENSAE
gabriel.watkinson@ensae.fr

Introduction

Scored-based Generative Models (SGMs) have gained significant attention in the field of deep image generation. Since their introduction by Song et al. (2020), SGMs have emerged as the state-of-the-art approach for image generation. Unlike likelihood-based models such as VAEs or normalizing flows, which learn the probability density function directly, or adversarial-based models like GANs. SGMs learn a mapping between a simple distribution, typically Gaussian, and complex, high-dimensional, and often intractable data distributions, such as images. These models offer several advantages over existing approaches, including high-quality sample generation comparable to GANs without the instability of adversarial training, flexible model architectures, exact log-likelihood computation, and the ability to solve inverse problems without re-training models. However, SGMs also suffer from pitfalls that limit their applicability, such as difficult training procedure, the impossibility to use a complex distribution as the prior, the fact that the drift can only be linear or degenerate, and slow training and sampling processes compared to other methods.

To address these limitations, recent works by De Bortoli et al. (2021); Wang et al. (2021); Vargas et al. (2021) have proposed a new class of generative models inspired by Optimal Transport (OT) theory, specifically using Schrödinger Bridges (SB) Schrödinger (1932). SBs, similar to SGMs, aim to learn a mapping between two distributions in a finite horizon. However, SBs offer greater flexibility by allowing the mapping between arbitrary distributions, making them more appealing. The relationship between SBs and SGMs was not clearly established, as SGMs rely on Stochastic Differential Equations (SDEs), whereas SBs are based on Partial Differential Equations (PDEs).

In this paper, Chen et al. (2023) bridges the gap between these seemingly distinct mathematical concepts by drawing inspiration from the field of Stochastic Optimal Control and utilizing Forward-Backward Stochastic Differential Equations (FBSDEs). The authors propose a method that enables the solution of PDEs with SDEs, offering computational efficiency and improved precision compared to solving PDEs directly. This approach introduces a novel training method for SBs, resembling the framework used in flow-based diffusion models. Additionally, the authors also use a more traditional Iterative Proportional Fitting approach (IPF; Kullback (1968); De Bortoli et al. (2021)).

In this report, we will first introduce the two classical methods, Scored-based Generative Models and Schrödinger Bridges, before exploring the Forward-Backward Stochastic Differential Equations method that establishes a connection between the two. Finally, we will discuss further extensions of this paper, including an application of the method with the Hamilton-Jacobi-Bellman PDE.

1 Some reminders about Score-based Generative Model

Given a dataset $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ of N samples in \mathbb{R}^d , the goal of generative models is to learn the underlying data distribution p_{data} , as to be able to generate new samples from this distribution. The go to approach for this kind of problem is to use maximum likelihood estimation (MLE) to learn the parameters of a parametric model p_θ that approximates the distribution. However, this relies on the assumption that p_θ be a normalized density function. To do so, we have to calculate the normalizing constant, which is intractable in high dimension. Therefore, to train such models using MLE, we either have to restrict the architectures, e.g. to invertible networks in flow models, or approximate the normalizing constant, using variational inference in VAEs for instance. This is not ideal, as it limits the flexibility of the model and the quality of the approximation.

Another solution is to use the score function $s_\theta(x) = \nabla_x \log p_\theta(x)$ instead of the density function p_θ . This makes the normalization constant disappear thanks to the gradient, allowing a wider range of architectures. Score matching methods can then be used to learn the score function s_θ , followed Langevin dynamics to sample from the learned score function, giving access to the wanted distribution. However, this naive approach doesn't work well in practice, especially in low density regions.

The solution is to add a diffusion process to the score matching method, which is the idea behind Score-based Generative Models (SGM). In SGMs, inputs are "diffused" toward a random noise. This diffusion process is characterized by a stochastic differential equation (SDE):

$$d\mathbf{X}_t = f(t, \mathbf{X}_t)dt + g(t)d\mathbf{W}_t, \quad \mathbf{X}_0 \sim p_{\text{data}} \quad (1)$$

where $f(\cdot, t) : \mathbb{R}^n \rightarrow \mathbb{R}^n$, $g(t) \in \mathbb{R}$ are the drift and diffusion coefficients, and $\mathbf{W}_t \in \mathbb{R}^n$ is a standard Wiener process. For a large time step, we have that $p_T \approx p_{\text{prior}}$ that closely resemble a Gaussian distribution, which is easy to sample from.

The associated backward SDE is:

$$d\mathbf{X}_t = [f(t, \mathbf{X}_t) - g(t)^2 \nabla_x \log p_t(\mathbf{X}_t)] dt + g(t)d\mathbf{W}_t, \quad \mathbf{X}_T \sim p_T \quad (2)$$

The score function appears naturally in the backward SDE, which is the reason why SGMs are called score-based. In the case where the drift is simple (e.g. linear or degenerate), it is possible to learn a score network $s(t, x; \theta)$ by regressing the outputs to the ground truth, i.e $\mathbb{E} \left[\lambda(t) \|s(t, x; \theta) - \nabla_x \log p_{t|x_0}\|^2 \right]$, where the expectation is taken over the forward SDE. $\lambda(t)$ is a weighting function that is crucial in practice but hard to choose well.

More recently, Song et al. (2021) have found a Evidence Lower bound (ELBO) that lower-bounds the log-likelihood of the SGM:

$$\log p_0^{\text{SGM}}(\mathbf{x}_0) \geq \mathbb{E} [\log p_T(\mathbf{X}_t)] - \int_0^T \mathbb{E} \left[\frac{1}{2} g^2 \|\mathbf{s}_t\|^2 - \nabla_x \cdot (g^2 \mathbf{s}_t - f) \right] dt \quad (3)$$

This can be obtained bounding the KL divergence by score-matching loss, where the weighting function $\lambda(t) = g(t)^2$. This lower bound is important since it will be one of the element that links together the SGM model and the Schrödinger-Bridge problem.

Once the model is trained, we can just substitute the score function by the learned one in the backward SDE (2) to sample from the data distribution.

2 Schrodinger Bridge

Unlike SGMs, Schrödinger Bridges (SB) are not generative models, but rather a way to interpolate between two distributions μ_0 and μ_1 . There were first introduced by Schrödinger (1932) and are related to the field of regularized optimal transport. The Schrödinger Bridge problem can be defined as follows:

$$\Pi^* = \arg \min \{ \text{KL}(\Pi | \mathbb{P}) \mid \Pi \in \mathcal{P}(\mathcal{C}), \Pi_0 = p_{\text{data}}, \Pi_T = p_{\text{prior}} \}$$

where Π^0 is the reference measure, which is set to the path measure of the forward diffusion process in (1).

The **Benamou-Brenier** reformulation of the entropy-regularized optimal transport problem gives the following partial differential equations (PDEs):

$$\begin{cases} \frac{\partial \Psi}{\partial t} = -\nabla_x \Psi^\top f - \frac{1}{2} \text{Tr}(g^2 \nabla_x^2 \Psi) \\ \frac{\partial \hat{\Psi}}{\partial t} = -\nabla_x \cdot (\hat{\Psi} f) + \frac{1}{2} \text{Tr}(g^2 \nabla_x^2 \hat{\Psi}) \end{cases} \quad \text{s.t.} \quad \begin{cases} \Psi(0, \cdot) \hat{\Psi}(0, \cdot) = p_{\text{data}} \\ \Psi(T, \cdot) \hat{\Psi}(T, \cdot) = p_{\text{prior}} \end{cases} \quad (4)$$

where Ψ and $\hat{\Psi}$ can also be seen as solutions to the Schrödinger system of equations (we recall that the trace of the Hessian is the Laplace operator). f and g are functions defined on the same space as above.

If Ψ and $\hat{\Psi}$ are solutions of these PDEs, then the solution of the SB problem is the path measure (i.e., that is in $\mathcal{P}(p_{\text{data}}, p_{\text{prior}})$) of the following SDEs:

$$\begin{aligned} d\mathbf{X}_t &= [f + g^2 \nabla_x \log \Psi(t, \mathbf{X}_t)] dt + g d\mathbf{W}_t, & \mathbf{X}_0 &\sim p_{\text{data}} \\ d\mathbf{X}_t &= [f - g^2 \nabla_x \log \hat{\Psi}(t, \mathbf{X}_t)] dt + g d\mathbf{W}_t, & \mathbf{X}_T &\sim p_{\text{prior}} \end{aligned} \quad (5)$$

This establishes a first link between PDEs founding the SB problem and SDEs characterizing the SGM model. Those SDEs are really similar to the ones of the SGM model, except for the terms $\nabla_x \log \Psi(t, \mathbf{X}_t)$ which adds non-linearity to the drift and $\nabla_x \log \hat{\Psi}(t, \mathbf{X}_t)$ which can be seen as a modified score function. In this situation, we have to learn both the score function and the drift function, while SGM only learns the score.

To create sample using this method, we have to solve the PDEs (4) to obtain $\hat{\Psi}$. However, those are hard to solve, the proposed method of FBSDEs (Forward-Backward SDEs) is shown to be a convenient way to link the optimality conditions of the SB problem to the ELBO maximization of the likelihood $\log_0^{\text{SGM}}(\mathbf{x}_0)$.

3 Proposed Method

There are two main methods to solve PDEs: linearization-based method and sampling-based method. The latter is a consequence of the Feynman-Kac lemma that provides a solution to parabolic PDEs with a conditional expectation (suitable for sampling), which establishes a link between this kind of PDE with stochastic processes. Here, the Nonlinear Feynman-Kac formula enables to link parabolic PDEs to forward-backward coupled SDEs as below. First let's introduce a more general framework, so that (1) and (2) fall inside:

$$\begin{cases} d\mathbf{X}_t = f(t, \mathbf{X}_t)dt + G(t, \mathbf{X}_t)d\mathbf{W}_t, & \mathbf{X}_0 = \mathbf{x}_0 \\ d\mathbf{Y}_t = -h(t, \mathbf{X}_t, \mathbf{Y}_t, \mathbf{Z}_t)dt + \mathbf{Z}(t, \mathbf{X}_t)^\top d\mathbf{W}_t, & \mathbf{Y}_T = \phi(\mathbf{X}_T). \end{cases} \quad (6)$$

Then, with the **Itô Formula** Exarchos and Theodorou (2018), it is shown that if $\mathbf{v} \equiv v(t, \mathbf{X}_t)$ and $G(t, \mathbf{X}_t)$ are solution of the following PDE:

$$\frac{\partial v}{\partial t} + \frac{1}{2} \text{Tr}(\nabla_{\mathbf{x}}^2 v G G^\top) + \nabla_{\mathbf{x}} v^\top f + h(t, \mathbf{x}, v, G^\top \nabla_{\mathbf{x}} v) = 0, \quad v(T, \mathbf{x}) = \phi(\mathbf{x}) \quad (7)$$

Then we get :

$$v(t, \mathbf{X}_t) = \mathbf{Y}_t \quad \text{and} \quad G(t, \mathbf{X}_t)^\top \nabla_{\mathbf{x}} v(t, \mathbf{X}_t) = \mathbf{Z}_t$$

Where \mathbf{Y}_t and \mathbf{Z}_t are variables from (6). Therefore, it is possible to solve one of the two problem 6 or 7 to solve the other. The equation (7) can be viewed as a second-order parabolic PDE, which can be solved using FBSDE (Pereira et al. 2020). Similarly, instead of solving the PDEs linked to the SB problem, it is possible to solve the underlying forward and backward SDEs. But some adaptations are necessary. Then, a new process is defined:

$$\begin{cases} d\mathbf{X}_t = (f + g\mathbf{Z}_t)dt + g d\mathbf{W}_t \\ d\mathbf{Y}_t = \frac{1}{2} \mathbf{Z}_t^\top \mathbf{Z}_t dt + \mathbf{Z}_t^\top d\mathbf{W}_t \\ d\hat{\mathbf{Y}}_t = \left(\frac{1}{2} \hat{\mathbf{Z}}_t^\top \hat{\mathbf{Z}}_t + \nabla_{\mathbf{x}} \cdot (g \hat{\mathbf{Z}}_t - f) + \hat{\mathbf{Z}}_t^\top \hat{\mathbf{Z}}_t \right) dt + \hat{\mathbf{Z}}_t^\top d\mathbf{W}_t \end{cases} \quad (8)$$

Where, under the same PDE as in (7), where \mathbf{v} is replaced with $\log \Psi$ and $\log \hat{\Psi}$. It is then possible to use two times the **Itô formula** with $d \log \Psi$ and $d \log \hat{\Psi}$. Therefore we have:

$$\begin{aligned}\mathbf{Y}_t &= \log \Psi(t, \mathbf{X}_t), & \mathbf{Z}_t &= g \nabla_{\mathbf{x}} \log \Psi(t, \mathbf{X}_t) \\ \hat{\mathbf{Y}}_t &= \log \hat{\Psi}(t, \mathbf{X}_t), & \hat{\mathbf{Z}}_t &= g \nabla_{\mathbf{x}} \log \hat{\Psi}(t, \mathbf{X}_t)\end{aligned}$$

Then, it is easy to check that $\mathbf{Y}_t + \hat{\mathbf{Y}}_t = \log(\Psi(t, \mathbf{X}_t)\hat{\Psi}(t, \mathbf{X}_t)) = \log p_t^{\text{SB}}(\mathbf{X}_t)$ according to SB-PDE constraints.

We can also notice that if we have an approximation of \mathbf{Z}_t and $\hat{\mathbf{Z}}_t$, g being known (by design), we also have an approximation of $\nabla_x \log \hat{\Psi}(t, \mathbf{X}_t)$ and $\nabla_x \log \Psi(t, \mathbf{X}_t)$. \mathbf{Z}_t and $\hat{\mathbf{Z}}_t$ can be understood as policy to learn in order that p_T reaches p_{prior} , which is imposed in the SB model but not in SGM (where the *prior* distribution is approximated). Therefore \mathbf{Z}_t force the forward process to reach the prior distribution while $\hat{\mathbf{Z}}_t$ makes adjustments accordingly in the backward process.

Now, using the fact that:

$$\mathbb{E} [\mathbf{Y}_0 + \hat{\mathbf{Y}}_0 | \mathbf{X}_0 = x] = \nabla_x \log \hat{\Psi}(t, \mathbf{x}) \quad (9)$$

$$\mathbb{E} [\mathbf{Y}_0 | \mathbf{X}_0 = x] = \mathbb{E} \left[\mathbf{Y}_T - \int_0^T d\mathbf{Y}_t \middle| \mathbf{X}_0 = x \right] \quad (10)$$

$$\mathbb{E} [\mathbf{Y}_0 | \mathbf{X}_0 = x] = \mathbb{E} [\mathbf{Y}_T | \mathbf{X}_0 = x] - \int_0^T \left(\mathbb{E} \left[\int_0^T \frac{1}{2} \|\mathbf{Z}_t\|^2 \middle| \mathbf{X}_0 = x \right] + \mathbb{E} [\mathbf{Z}_t d\mathbf{W}_t | \mathbf{X}_0 = x] \right) dt \quad (11)$$

$$\mathbb{E} [\hat{\mathbf{Y}}_0 | \mathbf{X}_0 = x] = \mathbb{E} [\hat{\mathbf{Y}}_T | \mathbf{X}_0 = x] - \int_0^T \mathbb{E} \left[\frac{1}{2} \|\hat{\mathbf{Z}}_t\|^2 + \nabla_x \cdot (g\hat{\mathbf{Z}}_t - f) + \hat{\mathbf{Z}}_t^\top \mathbf{Z}_t \right] dt \quad (12)$$

This allows them to derive the theorem that solves all our issues:

$$\log p_0^{\text{SB}}(\mathbf{x}_0) = \mathbb{E} [\log p_T(\mathbf{X}_T)] - \int_0^T \mathbb{E} \left[\frac{1}{2} \|\mathbf{Z}_t\|^2 + \frac{1}{2} \|\hat{\mathbf{Z}}_t\|^2 + \nabla_x \cdot (g\hat{\mathbf{Z}}_t - f) + \hat{\mathbf{Z}}_t^\top \mathbf{Z}_t \right] dt \quad (13)$$

We now have an ELBO to maximize with every term tractable. That allows us to use a trainable network to learn the optimal controls \mathbf{Z}_t and $\hat{\mathbf{Z}}_t$ only, that are the optimal forward and backward drifts for SB.

4 Going further : SB FB-SDE to solve PDEs

It is possible to train neural network in order to sample paths that solve partial differential equations. In this part, some applications will be presented using the FBSDEs method to solve complexe PDEs.

A famous example is the Hamilton-Jacobi-Bellman (HJB) PDE. It is often used in optimal control theory since its solution gives the sought value function for an optimal control problem, and it plays a crucial role in dynamic programming providing efficient methods solving variational problems. The HJB PDE can be written as follows:

$$\begin{aligned}\frac{\partial V(t, x)}{\partial t} + \min_u \left\{ \frac{\partial V(t, x)}{\partial x} \times F(x, u) + C(x, u) \right\} &= 0 \\ \text{s.t. } V(T, x) &= \phi(x(T))\end{aligned}$$

Where $\phi(\cdot)$ is the function that gives the terminal state cost, $C(\cdot)$ is a quadratic cost function and $F(x(t), u(t)) = \dot{x}(t)$, $\forall t \in [0, T]$. For instance, we can take the mathematical framework of Pereira et al. 2019 where the stochastic optimal control (SOC) problem is the following:

$$J(t, \mathbf{x}(t); \mathbf{u}(t)) = \mathbb{E}_{\mathbb{Q}} \left[\int_t^T \left(q(s, \mathbf{x}(s)) + \frac{1}{2} \mathbf{u}(s)^\top \mathbf{R} \mathbf{u}(s) \right) ds + \phi(\mathbf{x}(T)) \right] \quad (\text{SOC-HJB})$$

Where

$$\begin{cases} d\mathbf{x}(t) &= (\mathbf{f}(t, \mathbf{x}(t)) + \mathbf{G}(t, \mathbf{x}(t))\mathbf{u}(t))dt + \sigma\mathbf{G}(t, \mathbf{x}(t))\mathbf{u}(t)dv(t) + \Sigma(t, \mathbf{x}(t))d\mathbf{w}(t) \\ \mathbf{x}(0) &= \xi \end{cases}$$

and where \mathbf{w} is a vector of mutually independent one dimensional variables following a standard Brownian motion (hence the probability measure \mathbb{Q}). The aim is to find the function value of the SOC-HJB problem, meaning:

$$\begin{cases} V(t, \mathbf{x}(t)) &= \inf_{\mathbf{u}(t) \in \mathcal{U}([t, T])} J(t, \mathbf{x}(t); \mathbf{u}(t)) \\ V(T, \mathbf{x}(T)) &= \phi(\mathbf{x}(T)) \end{cases}$$

Finally, we get the following partial differential equation:

$$\begin{cases} \frac{\partial V(t, \mathbf{x})}{\partial t} + q + \nabla_{\mathbf{x}} V(t, \mathbf{x})^\top \mathbf{f} - \frac{1}{2} \nabla_{\mathbf{x}} V(t, \mathbf{x})^\top \mathbf{G} \hat{\mathbf{R}}^{-1} \mathbf{G}^\top \nabla_{\mathbf{x}} V(t, \mathbf{x}) + \frac{1}{2} \text{Tr}(\nabla_{\mathbf{x}}^2 V(t, \mathbf{x}) \Sigma \Sigma^\top) = 0 \\ V(T, \mathbf{x}(T)) = \phi(\mathbf{x}(T)) \end{cases} \quad (\text{PDE-HJB})$$

Where $\hat{\mathbf{R}} := \mathbf{R} + \sigma^2 \mathbf{G}^\top \nabla_{\mathbf{x}}^2 V(t, \mathbf{x})$ and the optimal control is $\mathbf{u}^*(t, \mathbf{x}) = -\hat{\mathbf{R}}^{-1} \mathbf{G}(t, \mathbf{x})^\top \nabla_{\mathbf{x}} V(t, \mathbf{x})$. Then, PDE-HJB is a parabolic PDE and can be solve using the FBSDE method as above. As well as before, the Itô formula gives a link between this PDE and forward-backward SDEs: TODO

5 From Schrodinger to SDE: j'écris juste là pour éviter de casser tes trucs, mais globalement on se retrouve à la fin pour fusionner

The Schrödinger Bridge problem, a fundamental topic in probability theory and stochastic analysis, involves finding a probability distribution that connects two given distributions at different time points. In this research project, we focus on solving this problem using the Forward-Backward Stochastic Differential Equations (FB-SDE) framework. Specifically, we compare two formulations: one based on diffusion processes and the other derived from the Benamou-Brenier formulation of the Schrödinger Bridge.

In this equation (named equation (1)):

$$\begin{aligned} d\mathbf{X}_t &= f(t, \mathbf{X}_t)dt + g(t)d\mathbf{W}_t, \quad \mathbf{X}_0 \sim p_{\text{data}} \\ (\text{time-reversal}) \quad d\mathbf{X}_t &= [f(t, \mathbf{X}_t) - g(t)^2 \nabla_x \log p_t(\mathbf{X}_t)] dt + g(t)d\mathbf{W}_t, \quad \mathbf{X}_T \sim p_T \end{aligned} \quad (14)$$

the SDE associated with the diffusion process is presented. It describes the evolution of \mathbf{X}_t from an initial distribution p_{data} to a final distribution p_T at time T . Notably, the time-reversal version of Equation (1) provides insights into the backward dynamics of the process.

On the other hand, this equation (named equation (2)):

$$\begin{aligned} d\mathbf{X}_t &= [f + g^2 \nabla_x \log \Psi(t, \mathbf{X}_t)] dt + g d\mathbf{W}_t, \quad \mathbf{X}_0 \sim p_{\text{data}} \\ d\mathbf{X}_t &= [f - g^2 \nabla_x \log \hat{\Psi}(t, \mathbf{X}_t)] dt + g d\mathbf{W}_t, \quad \mathbf{X}_T \sim p_{\text{prior}} \end{aligned}$$

corresponds to the SDE derived from the Benamou-Brenier formulation of the Schrödinger Bridge. It introduces two different potentials, $\Psi(t, \mathbf{X}_t)$ and $\hat{\Psi}(t, \mathbf{X}_t)$, which influence the evolution of \mathbf{X}_t . They follow the following PDEs:

$$\begin{cases} \frac{\partial \Psi}{\partial t} &= -\nabla_x \Psi^\top f - \frac{1}{2} \text{Tr}(g^2 \nabla_x^2 \Psi) \\ \frac{\partial \hat{\Psi}}{\partial t} &= -\nabla_x \cdot (\hat{\Psi} f) + \frac{1}{2} \text{Tr}(g^2 \nabla_x^2 \hat{\Psi}) \end{cases} \quad (\text{SB-PDE})$$

s.t. $\Psi(0, \cdot) \hat{\Psi}(0, \cdot) = p_{\text{data}}, \Psi(T, \cdot) \hat{\Psi}(T, \cdot) = p_{\text{prior}}$

The initial and final distributions, p_{data} and p_{prior} respectively, play a crucial role in this formulation.

Although these two formulations exhibit similarities, there exists a fundamental distinction. In certain cases, such as linear or degenerate situations, Equation (1) admits an analytical solution for the

score function. This analytical solution greatly simplifies the process of generating X_0 from X_T . However, Equation (2) requires solving a coupled system of PDEs to obtain an analytical solution.

In this project, our primary objective is to recover the score function, as it enables straightforward generation of X_0 from X_T . To achieve this, we focus on the formulation presented in Equation (4) :

$$\begin{cases} d\mathbf{X}_t &= (f + g\mathbf{Z}_t)dt + g d\mathbf{W}_t \\ d\mathbf{Y}_t &= \frac{1}{2}\mathbf{Z}_t^\top \mathbf{Z}_t dt + \mathbf{Z}_t^\top d\mathbf{W}_t \\ d\hat{\mathbf{Y}}_t &= \left(\frac{1}{2}\hat{\mathbf{Z}}_t^\top \hat{\mathbf{Z}}_t + \nabla_{\mathbf{x}} \cdot (g\hat{\mathbf{Z}}_t - f) + \hat{\mathbf{Z}}_t^\top \hat{\mathbf{Z}}_t \right) dt + \hat{\mathbf{Z}}_t^\top d\mathbf{W}_t \end{cases} \quad (15)$$

By knowing Z_t and \hat{Z}_t , we can determine the score function and effectively solve the Schrödinger Bridge problem.

To learn Z_t and \hat{Z}_t , we adopt a parametric approach, typically employing a neural network as the function approximator. Similar to the score function, we require a tractable and effective loss function. In this regard the following loss, proves advantageous as it introduces a novel loss function derived from the likelihood expression:

$$\begin{aligned} \log p_0^{\text{SB}}(x_0) &= E[\log p_T(X_T)] - \int_0^T E \left[\frac{1}{2}\|Z_t\|^2 + \frac{1}{2}\|\hat{Z}_t - g\nabla_x \log p_t^{\text{SB}} + Z_t\|^2 - \frac{1}{2}\|g\nabla_x \log p_t^{\text{SB}} - Z_t\|^2 - \nabla_x \cdot f \right] dt, \\ &= E[\log p_T(X_T)] - \int_0^T E \left[\frac{1}{2}\|Z_t\|^2 + \frac{1}{2}\|\hat{Z}_t\|^2 + \nabla_x \cdot (g\hat{Z}_t - f) + \hat{Z}_t^\top Z_t \right] dt \end{aligned} \quad (16)$$

The key advantage of this loss function is its exclusion of the score function, thereby simplifying the learning process to solely involve Z_t and Z_t^{hat} .

This function is derivated similarly to the following loss used in diffusion process but also using equation (4):

$$\begin{aligned} \log p_0^{\text{SGM}}(x_0) &\geq \mathcal{L}_{\text{SGM}}(x_0; \theta) = E[\log p_T(X_T)] - \int_0^T E \left[\frac{1}{2}g^2\|s_t\|^2 + \nabla_x \cdot (g^2 s_t - f) \right] dt, \\ &= E[\log p_T(X_T)] - \int_0^T E \left[\frac{1}{2}g^2\|s_t - \nabla_x \log p_{t|x_0}\|^2 - \frac{1}{2}\|g\nabla_x \log p_{t|x_0}\|^2 - \nabla_x \cdot f \right] dt, \end{aligned} \quad (17)$$

Bibliography

- Tianrong Chen, Guan-Horng Liu, and Evangelos A. Theodorou. 2023. Likelihood training of schrödinger bridge using forward-backward sdes theory.
- Yongxin Chen, Tryphon T. Georgiou, and Michele Pavon. 2020. Stochastic control liaisons: Richard sinkhorn meets gaspard monge on a schroedinger bridge.
- Valentin De Bortoli, James Thornton, Jeremy Heng, and Arnaud Doucet. 2021. Diffusion schrödinger bridge with applications to score-based generative modeling. In *Advances in Neural Information Processing Systems*, volume 34, pages 17695–17709. Curran Associates, Inc.
- Ioannis Exarchos and Evangelos A. Theodorou. 2018. Stochastic optimal control via forward and backward stochastic differential equations and importance sampling. *Automatica*, 87:159–165.
- Chin-Wei Huang, Jae Hyun Lim, and Aaron C Courville. 2021. A variational perspective on diffusion-based generative models and score matching. In *Advances in Neural Information Processing Systems*, volume 34, pages 22863–22876. Curran Associates, Inc.
- Solomon Kullback. 1968. Probability densities with given marginals. *The Annals of Mathematical Statistics*, 39:1236–1243.
- Guan-Horng Liu, Tianrong Chen, Oswin So, and Evangelos A. Theodorou. 2022. Deep generalized schrödinger bridge.
- Christian Léonard. 2013a. A survey of the schrödinger problem and some of its connections with optimal transport.
- Christian Léonard. 2013b. A survey of the schrödinger problem and some of its connections with optimal transport.
- Marcus Pereira, Ziyi Wang, Tianrong Chen, Emily Reed, and Evangelos Theodorou. 2020. Feynman-kac neural network architectures for stochastic control using second-order fbsde theory. In *Proceedings of the 2nd Conference on Learning for Dynamics and Control*, volume 120 of *Proceedings of Machine Learning Research*, pages 728–738. PMLR.
- Marcus A Pereira, Ziyi Wang, Tianrong Chen, Emily Reed, and Evangelos A Theodorou. 2019. Deep 2fbsdes for systems with control multiplicative noise.
- Erwin Schrödinger. 1932. Sur la théorie relativiste de l’électron et l’interprétation de la mécanique quantique. In *Annales de l’institut Henri Poincaré*, volume 2, pages 269–310.
- Yang Song, Conor Durkan, Iain Murray, and Stefano Ermon. 2021. Maximum likelihood training of score-based diffusion models.
- Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. 2020. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*.
- Francisco Vargas, Pierre Thodoroff, Austen Lamacraft, and Neil Lawrence. 2021. Solving schrödinger bridges via maximum likelihood. *Entropy*, 23(9):1134.
- Gefei Wang, Yuling Jiao, Qian Xu, Yang Wang, and Can Yang. 2021. Deep generative learning via schrödinger bridge. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 10794–10804. PMLR.