Adversarial Schrödinger Bridge Matching

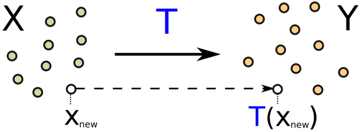
Nikita Gushchin[1,*] **Daniil Selikhanovych**[1,*] **Sergei Kholkin**[1,*]
Evgeny Burnaev[1,2] **Alexander Korotin**[1,2]

[1]Skolkovo Institute of Science and Technology
[2]Artificial Intelligence Research Institute
[*] *Nikita Gushchin, Daniil Selikhanovych and Sergei Kholkin contributed equally*

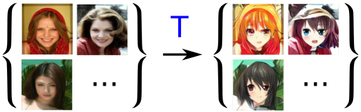
2024

Unpaired Domain Translation: The Problem which Motivated the Study[1]

The task: learn (from samples) a *generalizable translation* map between the two given data domains.



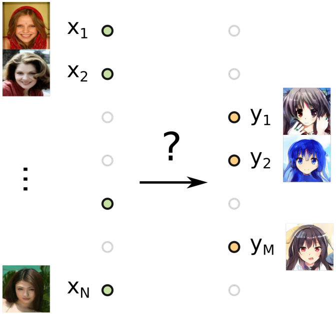
Example: Style Translation



Unsupervised setup

Only *unpaired* train samples are given:

$\{x_1, \ldots, x_N\}, \{y_1, \ldots, y_M\}$.



[1] Jun-Yan Zhu et al. (2017). "Unpaired image-to-image translation using cycle-consistent adversarial networks". In: *Proceedings of the IEEE international conference on computer vision*, pp. 2223–2232.

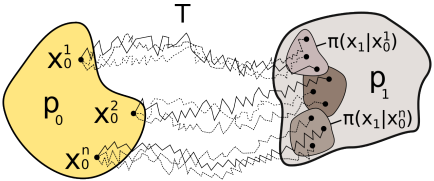
Schrödinger Bridge problem[2]

The Schrödinger Bridge problem

For two continuous distributions p_0 and p_1 on $\mathbb{R}^D$, the Schrödinger Bridge problem is:

$$\inf_{T \in \mathcal{F}(p_0, p_1)} \mathrm{KL}(T \| W^{\epsilon}).$$

Here $\mathcal{F}(p_0, p_1)$ are stochastic processes with marginals p_0, p_1 at $t = 0$ and $t = 1$.



Here W^{ϵ} is a Wiener process with the variance ϵ, i.e., it is a stochastic process with the stochastic differential equation (SDE): $dX_t = \sqrt{\epsilon} dW_t$.

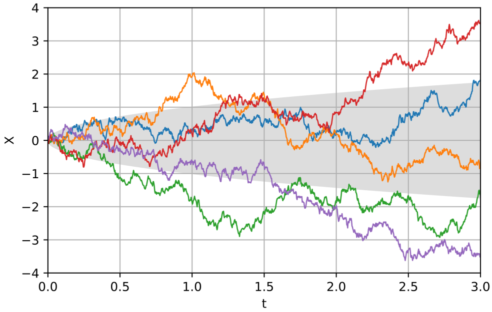


Figure 1: Wiener process with $\epsilon = 1$.

[2] Erwin Schrödinger (1931). *Über die umkehrung der naturgesetze*. Verlag der Akademie der Wissenschaften in Kommission bei Walter De Gruyter u ....

Reciprocal[3] and Markovian processes

Let $\mathcal{F}$ denote the set of all stochastic processes in $\mathbb{R}^D$ for $t \in [0,1]$ with continuous trajectories $\{x_t\}_{t\in[0,1]}$. We also denote Brownian Bridge $W^{\epsilon}_{|x_0,x_1}$ as the W^{ϵ} conditioned on x_0, x_1 at $t = 0, 1$.

Reciprocal processes. Let $\mathcal{R} \subset \mathcal{F}$ denote the subset of **reciprocal** processes, i.e., those processes can be represented as mixtures of Brownian bridges:

$$T \in \mathcal{R} \qquad \Leftrightarrow \qquad \exists \pi = \pi^T \in \mathcal{P}(\mathbb{R}^D \times \mathbb{R}^D) \text{ s.t. } T = T_\pi \stackrel{\text{def}}{=} \int W^{\epsilon}_{|x_0,x_1} \pi^T(x_0,x_1) dx_0 dx_1.$$

Markov Processes. Let $\mathcal{M} \subset \mathcal{F}$ denote the subset of **Markovian** processes, i.e.,

$$T \in \mathcal{M} \qquad \Leftrightarrow \qquad \forall N > 1,\ 0 \leq t_1 < \cdots < t_N \leq 1 :\ p^T(x_{t_N} | x_{t_{N-1}} \ldots, x_1) = p^T(x_{t_N} | x_{t_{N-1}}).$$

Schrödinger Bridge T^* is the only process starting at p_0 and ending at p_1 that is **both Markovian and reciprocal**.

[3] Yuyang Shi et al. (2023). "Diffusion Schrödinger Bridge Matching". In: *Thirty-seventh Conference on Neural Information Processing Systems*.

Bridge matching

Reciprocal projection

- Defined for any process $T \in \mathcal{F}$:

$$\text{proj}_{\mathcal{R}}(T) \stackrel{\text{def}}{=} \text{argmin}_{R \in \mathcal{R}} \text{KL}(T \| R)$$

- Yields a mixture of Brownian bridges:

$$\int W^{\epsilon}_{|x_0,x_1} \pi^T(x_0, x_1) dx_0 dx_1$$

Markovian projection

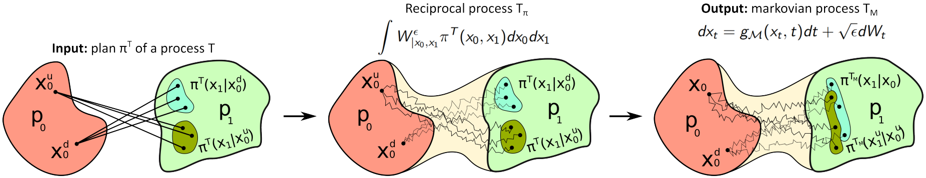
- Defined for a *reciprocal* process $T_\pi \in \mathcal{R}$:

$$\text{proj}_{\mathcal{M}}(T_\pi) \stackrel{\text{def}}{=} \text{argmin}_{M \in \mathcal{M}} \text{KL}(T_\pi \| M)$$

- Yields a **diffusion** with the SDE

$$dx_t = g_{\mathcal{M}}(x_t, t)dt + \sqrt{\epsilon} dW_t, \qquad x_0 \sim p_0.$$

Bridge matching = combination of Reciprocal and Markovian Projections

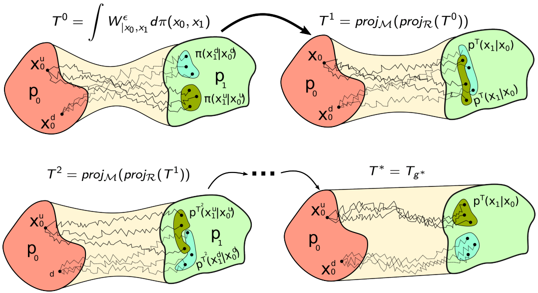


Iterative Markovian Fitting[4] (Iterative Diffusion Bridge Matching)

Alternating Markovian and Reciprocal projections is called the **Iterative Markovian Fitting** (IMF) procedure. Starting from a reciprocal process $T_0 = \int W^{\epsilon}_{|x_0,x_1} d\pi(x_0, x_1)$ induced by some initial plan $\pi(x_0, x_1)$, one performs iterative updates

$$T^{2n+1} = \text{proj}_{\mathcal{M}}(T^{2n}),\ T^{2n+2} = \text{proj}_{\mathcal{R}}(T^{2n+1})$$

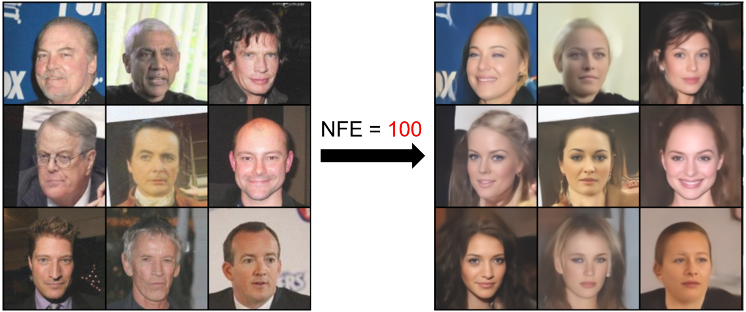
$\{T^n\}_{n=1}^{\infty}$ converges to the SB T^*: $\lim_{n\to+\infty} \text{KL}(T^n \| T^*) = 0$.



[4] Yuyang Shi et al. (2023). "Diffusion Schrödinger Bridge Matching". In: *Thirty-seventh Conference on Neural Information Processing Systems*.

Issues with IMF

Learning **continuous-time** SDEs in IMF is non-trivial and, unfortunately, leads to **long inference** due to the necessity to use many steps of numerical solvers. In the DSBM method[5] the number of sampling steps is 100, which is a lot.



[5]Yuyang Shi et al. (2023). "Diffusion Schrödinger Bridge Matching". In: *Thirty-seventh Conference on Neural Information Processing Systems*.

Our contributions

This paper addresses the above-mentioned limitation of the existing IMF framework by introducing a novel approach to learn the Schrödinger Bridge:

1. **Theory I.** We introduce a Discrete Iterative Markovian Fitting **(D-IMF)** procedure, which innovatively applies **discrete** Markovian projection to solve the SB problem without relying on SDE.
2. **Theory II.** We derive closed-form update formulas for the D-IMF procedure when dealing with high-dimensional Gaussian distributions.
3. **Practice.** For general data distributions available by samples, we propose an algorithm **(ASBM)** to implement the discrete Markovian projection and our D-IMF procedure in practice. Our algorithm is based on adversarial learning and DDGAN. Our learned SB model uses just 4 evaluation steps for inference instead of hundreds of the basic IMF.

Discrete Markovian and reciprocal stochastic processes

We define the **discrete reciprocal processes** using the finite-time projection of $W^{\epsilon}_{|x_0,x_1}$:

$$p^{W^\epsilon}(x_{t_1},\dots,x_{t_N}|x_0,x_1) = \prod_{n=1}^{N} p^{W^\epsilon}(x_{t_n}|x_{t_{n-1}},x_1),$$

$$p^{W^\epsilon}(x_{t_n}|x_{t_{n-1}},x_1) = \mathcal{N}\left(x_{t_n}|x_{t_{n-1}} + \frac{t_n - t_{n-1}}{1-t_{n-1}}(x_1 - x_{t_{n-1}}), \epsilon\frac{(t_n-t_{n-1})(1-t_n)}{1-t_{n-1}}\right).$$

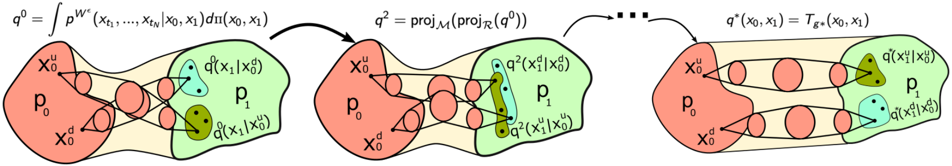
We introduce the **reciprocal projection** $\text{proj}_{\mathcal{R}}(q)$ as a process with the joint distribution:

$$\big[\text{proj}_{\mathcal{R}}(q)\big](x_0,x_{t_1},\dots,x_{t_N},x_1) = p^{W^\epsilon}(x_{t_1},\dots,x_{t_N}|x_0,x_1)q(x_0,x_1).$$

The **discrete Markovian projection** of q is a process $\text{proj}_{\mathcal{M}}(q)$ with the joint distribution:

$$\big[\text{proj}_{\mathcal{M}}(q)\big](x_0,x_{t_1},\dots,x_{t_N},x_1) = q(x_0)\prod_{n=1}^{N+1} q(x_{t_n}|x_{t_{n-1}}).$$

D-IMF procedure starts from any discrete Brownian mixture and constructs the following sequence of discrete stochastic processes: $q^{2l+1} = \text{proj}_{\mathcal{M}}(q^{2l}), \quad q^{2l+2} = \text{proj}_{\mathcal{R}}(q^{2l+1})$.



Main theorems

Theorem (Discrete Markovian and reciprocal process is the solution of static SB)

Consider any discrete process q, which is simultaneously reciprocal and Markovian, and has marginals $p_0(x_0)$ *and* $p_1(x_1)$*:*

$$q(x_0, x_{t_1}, \dots, x_{t_N}, x_1) = p^{W^\epsilon}(x_{t_1}, \dots, x_{t_N} | x_0, x_1) q(x_0, x_1) = q(x_0) \prod_{n=1}^{N+1} q(x_{t_n} | x_{t_{n-1}}).$$

Then $q(x_0, x_{t_1}, \dots, x_{t_N}, x_1) = p^{T^*}(x_0, x_{t_1}, \dots, x_{t_N}, x_1)$*, i.e., it is the finite-dimensional projection of the SB to the considered times.*

Theorem (D-IMF procedure converges to the the Schrödinger Bridge)

Under mild assumptions, the sequence q^l *constructed by our D-IMF procedure converges in KL to* p^{T^*}*. Namely, we have*

$$\lim_{l \to \infty} KL\left(q^l \| p^{T^*}\right) = 0, \qquad \text{and} \qquad \lim_{l \to \infty} KL\left(q^l(x_0, x_1) \| p^{T^*}(x_0, x_1)\right) = 0.$$

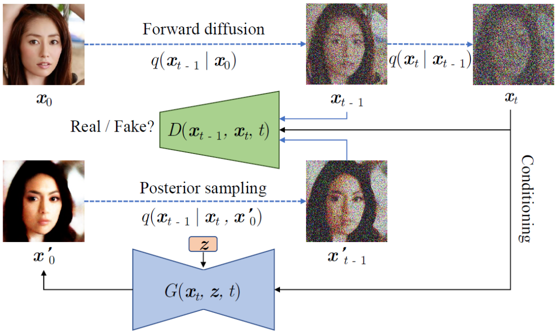
Practical implementation of D-IMF

To implement D-IMF in practice we need:

1. **Implementation of the discrete reciprocal projection.** To sample from reciprocal projection
$$\big[\text{proj}_{\mathcal{R}}(q)\big](x_0, x_{t_1}, \dots, x_{t_N}, x_1) = p^{W^\epsilon}(x_{t_1}, \dots, x_{t_N} | x_0, x_1) q(x_0, x_1)$$
it is enough to sample first a pair $(x_0, x_1) \sim q(x_0, x_1)$ and then sample intermediate points $x_{t_1}, \dots, x_{t_N}$ from the Brownian Bridge $p^{W^\epsilon}(x_{t_1}, \dots, x_{t_N} | x_0, x_1)$.
2. **Implementation of the discrete Markovian projection via DD-GAN.** To find the Markovian projection of a reciprocal process
$$\big[\text{proj}_{\mathcal{M}}(q)\big](x_0, x_{t_1}, ..., x_{t_N}, x_1) = q(x_0) \prod_{n=1}^{N+1} q(x_{t_n} | x_{t_{n-1}}),$$
one just needs to estimate the transition probabilities $\{q(x_{t_n} | x_{t_{n-1}})\}_{n=1}^{N+1}$ and use the starting marginal $q(x_0) = p_0(x_0)$. Similarly to DDGAN, we parametrize all these distributions as $\{q_\theta(x_{t_n} | x_{t_{n-1}})\}_{n=1}^{N+1}$ via a time-conditioned generator $G_\theta(x_{t_{n-1}}, z, t_{n-1})$. For a given $x_{t_{n-1}}$ sample $x_{t_n} \sim q_\theta(x_{t_n} | x_{t_{n-1}})$ is obtained by first sampling x_1 from the G_θ and then using sampling from the Brownian Bridge $p^{W^\epsilon}(x_{t_n} | x_{t_{n-1}}, x_1)$.

Denoising Diffusion GANs[6]

We use D_{adv} as a non-saturating GAN loss. To optimize this loss, an additional conditional discriminator $D(x_{t_{n-1}}, x_{t_n}, t_{n-1})$ is needed. In the DDGAN the distribution $q(x_{\text{in}}|x_0, x_1)$ is used from DDPM and **it is the main difference between our discrete Markovian projection and DDGAN**.



We minimize over θ the following loss:

$$\sum_{n=1}^{N+1} \mathbb{E}_{q(x_{t_{n-1}})} D_{\text{adv}}\left(q(x_{t_n}|x_{t_{n-1}})||q_\theta(x_{t_n}|x_{t_{n-1}})\right).$$

[6]Zhisheng Xiao, Karsten Kreis, and Arash Vahdat (2022). "Tackling the Generative Learning Trilemma with Denoising Diffusion GANs". In: *International Conference on Learning Representations*.

Evaluation

To test our approach on real data, we consider the unpaired image-to-image translation setup of learning *male* $\to$ *female* faces of Celeba dataset:

- **Train-test split**. We use 10% of *male* and *female* images as the test set for evaluation.
- **Hyperparameters**. We train our ASBM based on the D-IMF procedure with $\epsilon = 1$ and $\epsilon = 10$. Following the best practices of DD-GAN, we use $N = 3$, intermediate times $t_1 = \frac{1}{4}, t_2 = \frac{2}{4}, t_3 = \frac{3}{4}$ and $K = 5$ outer iterations of D-IMF.
- **Evaluation protocol**. We provide qualitative results and the FID metric on the test set.
- **Comparison**. We focus our comparison on the DSBM algorithm[7] since it is closely related to our method. We train DSBM following the authors and use NFE $= 100$. As well as for ASBM, we use 5 outer iterations of IMF for continuous processes.
- We use 42M and 38M parameters of neural networks for ASBM and DSBM respectively.

[7] Yuyang Shi et al. (2023). "Diffusion Schrödinger Bridge Matching". In: *Thirty-seventh Conference on Neural Information Processing Systems*.

Results on Celeba-128, *male* $\to$ *female*

(a) $x \sim p_0$ **(b)** ASBM (**ours**), $\epsilon = 1$ (lower diversity) **FID** $= 16.08$, **NFE** $= 4$. **(c)** DSBM, $\epsilon = 1$ (lower diversity) FID $= 37.8$, NFE $= 100$.

Our algorithm is scalable and provides better results while using only 4 evaluation steps.

Results on Celeba-128, *male* $\to$ *female*

(a) $x \sim p_0$

(b) ASBM (**ours**), $\epsilon = 10$ (higher diversity)
FID $= 17.44$, **NFE** $= 4$.

(c) DSBM, $\epsilon = 10$ (higher diversity)
FID $= 89.19$, NFE $= 100$.

DSBM experiences a notable increase in FID with $\epsilon = 10$. We conjecture that this is due to the FID unstability w.r.t. slightly noisy images from integration of noisy trajectories.

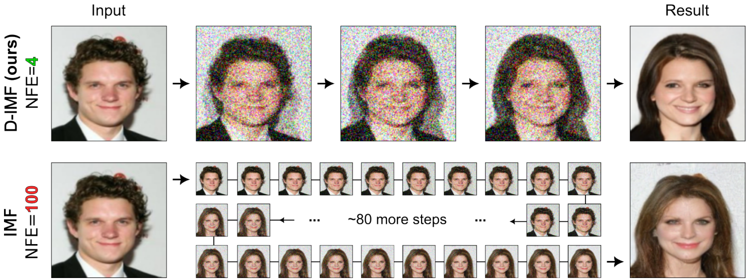
Results on Celeba-128, *female* $\to$ *male*

(a) $x \sim p_0$ **(b)** ASBM (**ours**), $\epsilon = 1$, (lower diversity) **FID** $= 16.87$, **NFE** $= 4$. **(c)** DSBM, $\epsilon = 1$, (lower diversity) FID $= 24.06$, NFE $= 100$.

Similar to DSBM, our algorithm trains both forward and backward models. The backward model also achieves good results.

Adversarial Schrödinger Bridge Matching (ASBM)

A novel Discrete-time IMF procedure in which learning of stochastic processes is replaced by learning just a few transition probabilities in discrete time.



GitHub

https://github.com/Daniil-Selikhanovych/ASBM