DeepMind

From Denoising Diffusion Models to Dynamic Transport Models

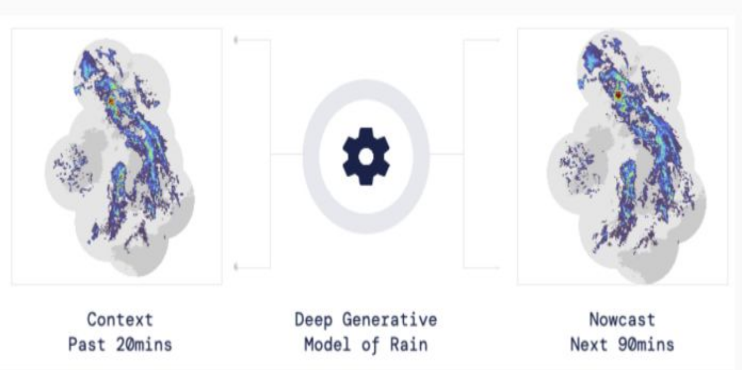
Arnaud Doucet
with Yuyang Shi, Valentin De Bortoli & Andrew Campbell
https://arxiv.org/abs/2303.16852

Fondements Mathématiques de l'IA
2 Octobre 2023

Generative Modeling

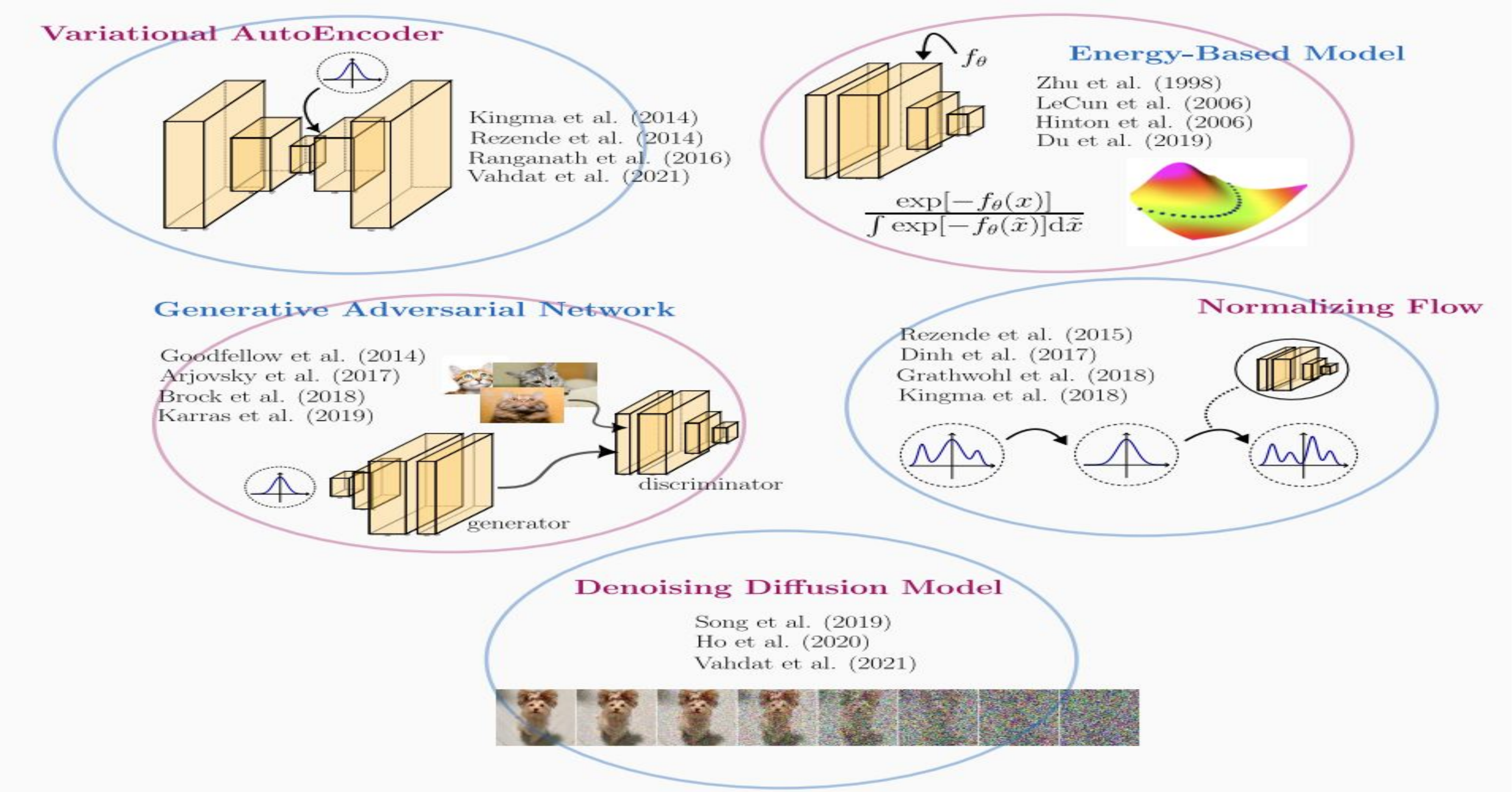
- **Formulation:** Given samples $(x_i)_{i=1}^N$ from distribution π, generate new samples distributed approximately from π.
- **Numerous applications**, e.g. data augmentation for downstream tasks (e.g. videos for self-driving cars), high-resolution nowcasting, data-driven priors for inverse problems/Bayesian inference.

From Ho et al., NeurIPS 2020



from Ravuri et al. (2021)

Generative Models

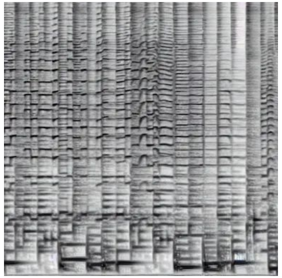


Denoising Diffusion Models

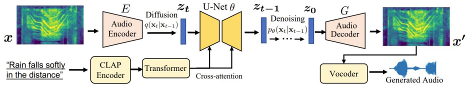
DALLE-3
OpenAI

MidJourney

Imagen
Google

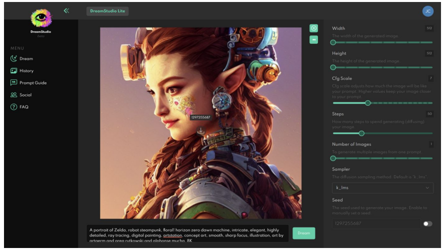


Riffusion



Make-an-audio

Runway



Stable Diffusion

Denoising Diffusions aka Score-Based Generative Models

Public

- Having a hard time keeping up with the literature?
  - List of references: https://scorebasedgenerativemodeling.github.io/
- Advantages of the method:
  - **State-of-the-art** results Dhariwal and Nichol (2021); Karras et al. (2022).
  - **High flexibility** Poole et al. (2022); Rombach et al. (2022); Balaji et al. (2022); Saharia et al. (2022).
  - **Theoretical analysis** De Bortoli et al. (2021b); Chen et al. (2022); Pidstrigach (2022); Lee et al. (2022).
- Some drawbacks:
  - **Statistical understanding** is still limited.

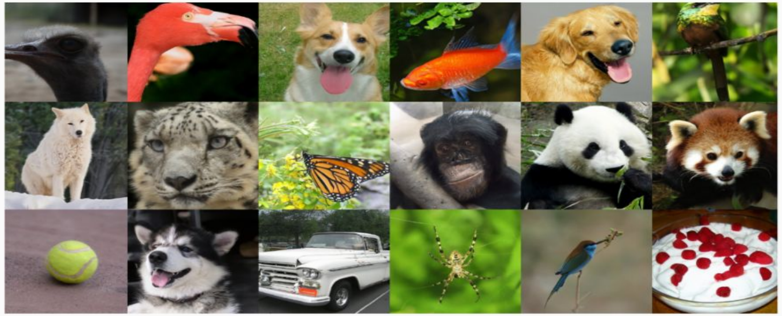
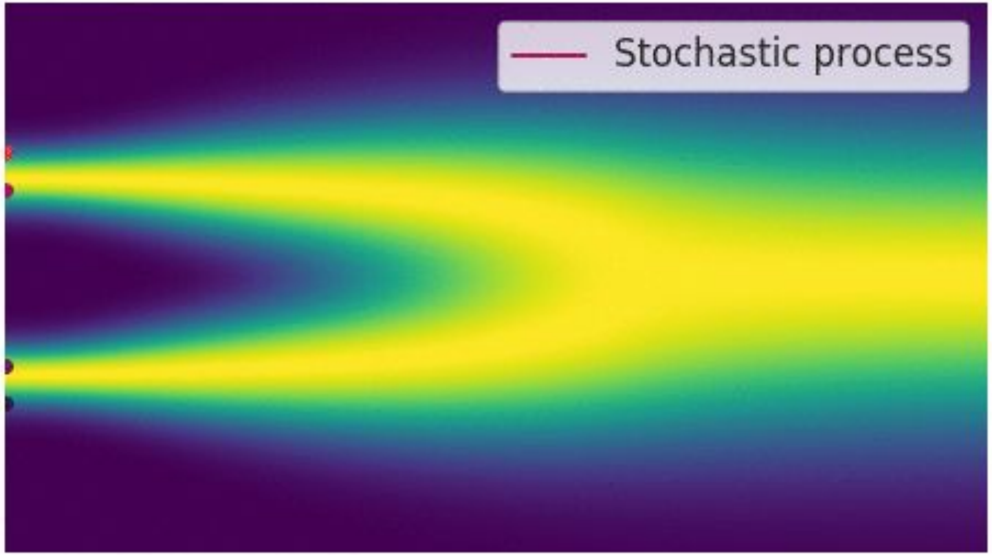


Figure 1: DDM results. Image extracted from Dhariwal and Nichol (2021).

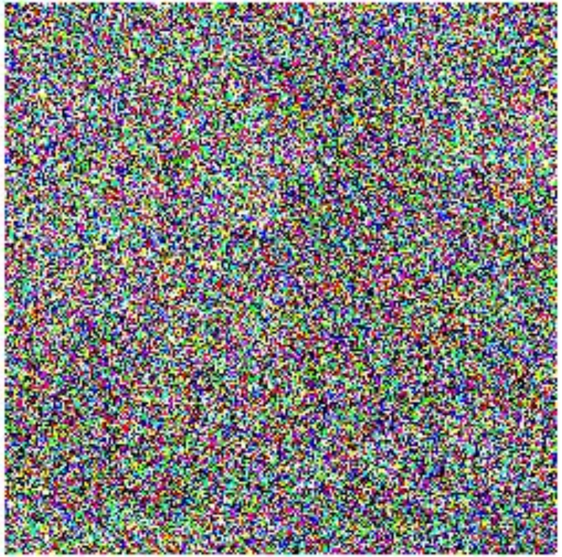
Introduction: Noising Mechanism

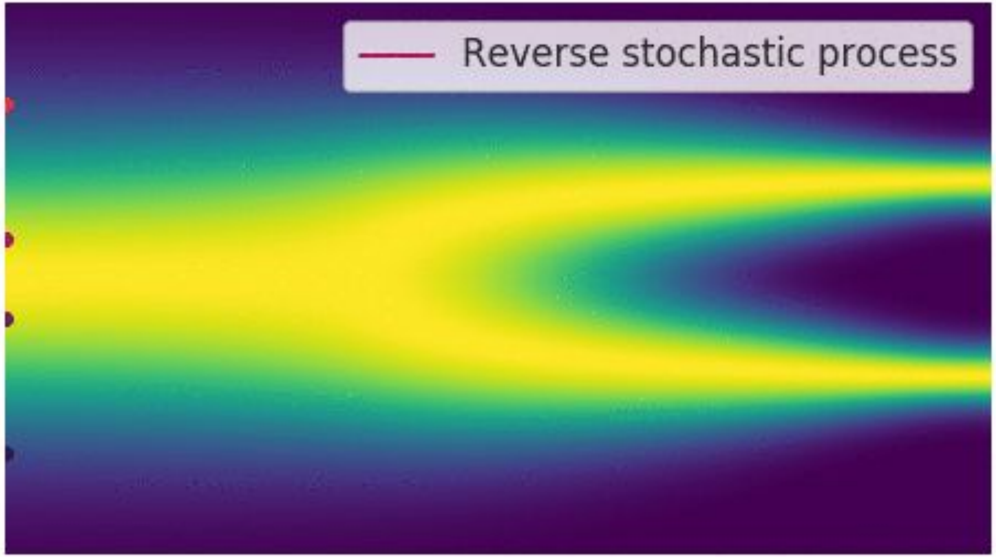


Bimodal distribution progressively diffused to Gaussian distribution

Song et al. 'Score-Based Generative Modeling through Stochastic Differential Equations' (2021)

Introduction: Denoising Mechanism





Song et al. 'Score-Based Generative Modeling through Stochastic Differential Equations' (2021)

Discrete-Time: Noising Process

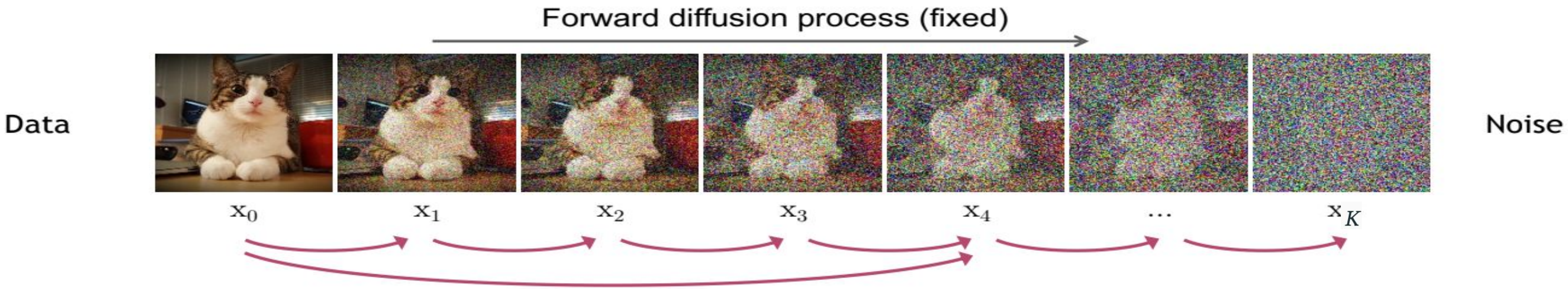


Fig. from Kreis, Gao and Vahdat, tutorial CVPR 2022.

- **At sample level**: Given data sample $\mathbf{X}_0$, set for $k = 0, \ldots, K-1$

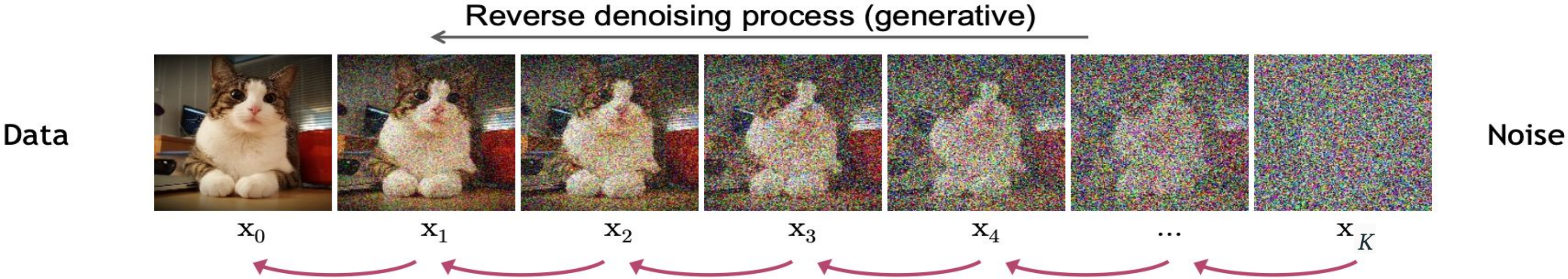
$$\mathbf{X}_{k+1} = \sqrt{1-\beta}\mathbf{X}_k + \sqrt{\beta}\epsilon_{k+1}, \quad \epsilon_{k+1} \sim \mathcal{N}(0, I)$$

- **At distribution level:**

$$q(x_{0:K}) = \underbrace{q_0(x_0)}_{\text{data dist}} \prod_{k=0}^{K-1} \underbrace{q(x_{k+1}|x_k)}_{\mathcal{N}(x_{k+1};\sqrt{1-\beta}x_k,\beta I)}$$

so $q(x_k|x_0) = \mathcal{N}(x_k; \sqrt{\alpha_k}x_0, (1-\alpha_k)I)$ with $\alpha_k = (1-\beta)^k$ (Ho et al., 2020)

Discrete-Time: Denoising Process



- **At distribution level**: **Ancestral Sampling**

$$q(x_{0:K}) = q_K(x_K) \prod_{k=0}^{K-1} q(x_k|x_{k+1})$$

where Bayes' rule yields

$$q(x_k|x_{k+1}) = q(x_{k+1}|x_k) \overbrace{\frac{q_k(x_k)}{q_{k+1}(x_{k+1})}}^{\textbf{intractable}}$$

- **At sample level**: Sample $\mathbf{X}_K \sim q_K$ then $\mathbf{X}_k \sim q(\cdot|\mathbf{X}_{k+1})$ for $k = K-1, \ldots, 0$.

Discrete-Time: Denoising Process

Public

- **First step**: Use the fact that

$$q(x_k|x_{k+1}) = \frac{q(x_{k+1}|x_k)\,q_k(x_k)}{q_{k+1}(x_{k+1})} \propto q(x_{k+1}|x_k)\exp(\log q_k(x_k))$$

where $q_k \approx q_{k+1}$ and a Taylor expansion yields

$$\log q_k(x_k) \approx \log q_{k+1}(x_{k+1}) + \nabla \log q_{k+1}(x_{k+1})^{\mathrm{T}}(x_k - x_{k+1}) \quad (*)$$

- **Second step:** Combining $q(x_{k+1}|x_k)$ and $(*)$

$$\begin{aligned} q(x_k|x_{k+1}) &\approx q(x_{k+1}|x_k)\exp[\nabla \log q_{k+1}(x_{k+1})^{\mathrm{T}}(x_k - x_{k+1})] \\ &\approx \mathcal{N}\left(x_k; \frac{1}{\sqrt{1-\beta}}\left(x_{k+1} + \beta\,\underbrace{\nabla \log q_{k+1}(x_{k+1})}_{\textbf{intractable}}\right), \beta I\right) \end{aligned}$$

From Discrete-Time to Continuous-Time

Public

- **Noising diffusion**: consider the diffusion $(\mathbf{X}_t)_{t\in[0,T]}$

$$\mathrm{d}\mathbf{X}_t = -\gamma\mathbf{X}_t\mathrm{d}t + \sqrt{2\gamma}\mathrm{d}\mathbf{B}_t, \quad \mathbf{X}_0 \sim q_0.$$

where $(\mathbf{B}_t)_{t\in[0,T]}$ is a Brownian motion and let $q_t = \mathrm{Law}(\mathbf{X}_t)$.
$\rightsquigarrow$ transports **data to noise**, i.e. q_0 to $q_T \approx \mathcal{N}(0, I)$.

- **Denoising diffusion** (Anderson, 1982) : the time-reversal $(\mathbf{Y}_t)_{t\in[0,T]}$ with $\mathbf{Y}_t = \mathbf{X}_{T-t}$ verifies

$$\mathrm{d}\mathbf{Y}_t = \gamma(\mathbf{Y}_t + 2\nabla\log q_{T-t}(\mathbf{Y}_t))\mathrm{d}t + \sqrt{2\gamma}\mathrm{d}\mathbf{W}_t, \quad \mathbf{Y}_0 \sim q_T.$$

where $(\mathbf{W}_t)_{t\in[0,T]}$ is a Brownian motion.
$\rightsquigarrow$ transports **noise to data**, i.e. $q_T \approx \mathcal{N}(0, I)$ to q_0.

- **Problem**: Score $\nabla\log q_t$ is **intractable**.

Score Approximation

Public

- **"Ideal" denoising diffusion**:

$$d\mathbf{Y}_t = \gamma(\mathbf{Y}_t + 2\nabla \log q_{T-t}(\mathbf{Y}_t))dt + \sqrt{2\gamma}d\mathbf{W}_t, \quad \mathbf{Y}_0 \sim q_T,$$

  induces path measure $\mathbb{P}$.

- **Approximate denoising diffusion**:

$$d\mathbf{Y}_t = \gamma(\mathbf{Y}_t + 2s_\theta(T-t, \mathbf{Y}_t))dt + \sqrt{2\gamma}d\mathbf{W}_t, \quad \mathbf{Y}_0 \sim \mathcal{N}(0, I),$$

  induces path measure $\mathbb{Q}_\theta$.

- **Learning the score**: Minimizing $\mathrm{KL}(\mathbb{P}||\mathbb{Q}_\theta)$ w.r.t. θ is equivalent to minimizing

$$\begin{aligned}\mathcal{L}(\theta) &= \int_0^T \mathbb{E}[||s_\theta(t, \mathbf{X}_t) - \nabla \log q_t(\mathbf{X}_t)||^2]dt \\ &= \int_0^T \mathbb{E}[||s_\theta(t, \mathbf{X}_t) - \nabla \log q_{t|0}(\mathbf{X}_t|\mathbf{X}_0)||^2]dt + C.\end{aligned}$$

What's the score?

Public

- **Tweedie's formula**

$$\nabla \log q_t(\mathbf{X}_t) = \mathbb{E}[\nabla \log q_{t|0}(\mathbf{X}_t|\mathbf{X}_0)].$$

- **Explicit expression**: Ornstein–Uhlenbeck transition

$$q_{t|0}(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\alpha_t}\mathbf{x}_0, (1-\alpha_t)I)$$

so

$$\nabla \log q_t(\mathbf{X}_t) = \frac{\sqrt{\alpha_t}\ \overbrace{\mathbb{E}[\mathbf{X}_0|\mathbf{X}_t]}^{\textbf{denoiser}} - \mathbf{X}_t}{1-\alpha_t}.$$

Learning the score $\iff$ Learning a denoiser

Benefits of Continuous-Time

Public

- **Sophisticated numerical integrators**, predictor-corrector schemes.

- **Probability flow ODE** (Song et al., 2021):

$$\mathrm{d}\mathbf{X}_t = -\gamma(\mathbf{X}_t + \nabla \log q_t(\mathbf{X}_t))\mathrm{d}t, \quad \mathbf{X}_0 \sim q_0$$

admits same marginals as noising diffusion; i.e. $\mathrm{Law}(\mathbf{X}_t) = q_t$
$\rightsquigarrow$ transports **data to noise**, i.e. q_0 to $q_T \approx \mathcal{N}(0, I)$.

- **Deterministic generation**: Let $\mathbf{Y}_t = \mathbf{X}_{T-t}$ then

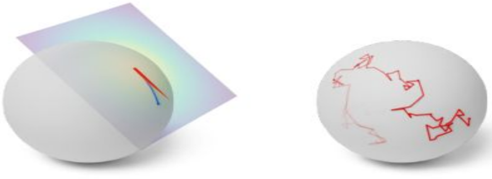
$$\mathrm{d}\mathbf{Y}_t = \gamma(\mathbf{Y}_t + \nabla \log q_{T-t}(\mathbf{Y}_t))\mathrm{d}t, \quad \mathbf{Y}_0 \sim q_T$$

$\rightsquigarrow$ transports **noise to data**, i.e. q_0 to $q_T \approx \mathcal{N}(0, I)$.

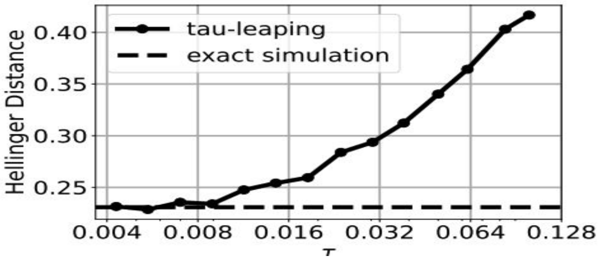
Extensions Beyond $\mathbb{R}^d$

Public

- **Riemmanian manifolds** (De Bortoli et al., 2022; Huang et al., 2022): applications to protein backbone generation (Watson et al., 2022) and joint grasp and motion optimization (Urain et al., 2022).
- **Discrete state-space** (Campbell et al., 2022): allows to exploit fast samplers from chemical physics.
- **Simplex** (Benton et al., 2022; Richemond et al., 2022): application to compositional data.
- **Function spaces** (Dutordoir et al., 2022; Kerrigan et al., 2022)



(a) A single step of a Geodesic Random Walk. (b) Many steps yield an approximate trajectory.



(a) GP Regression (b) Attentive Latent NP (c) Neural Diffusion Process (ours)

Dutortoir et al., "Neural Diffusion Processes"

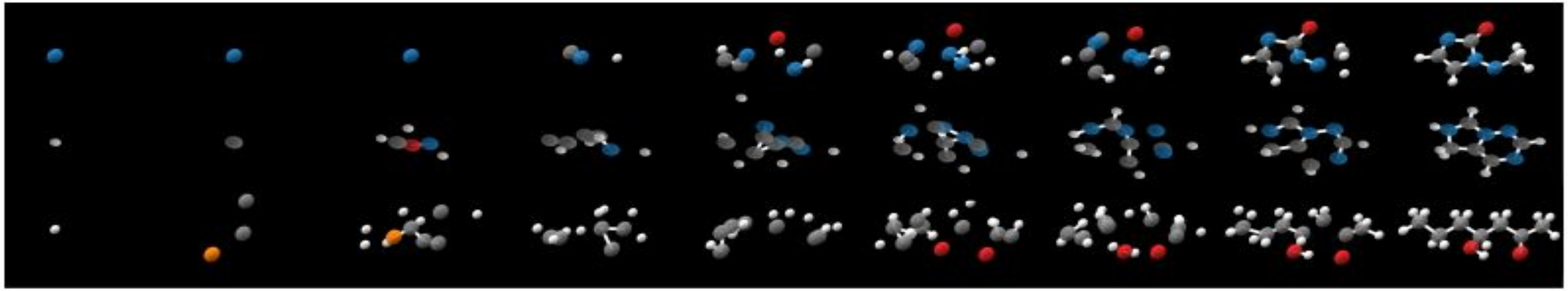
Trans-dimensional generative modeling via jump diffusions

Public

- **Data of varying dimension**: molecules with varying number of atoms, videos with varying number of frames; i.e. π lives on $\cup_{n=1}^{n_{\max}} \mathbb{R}^n$.
- **Difficulties**: Training $n_{\max}$ models expensive and conditional generation would require predicting n given observations.
- **Jump Diffusion**: We diffuse and kill components until one remains and is Gaussian, time-reversal adds components.

Algorithm 1: Sampling the Generative Process

$t \leftarrow T$
$\mathbf{X} \sim p_{\text{ref}}(\mathbf{X}) = \mathbb{I}\{n=1\}\mathcal{N}(\mathbf{x}; 0, I_d)$
while $t > 0$ **do**
  if $u < \overleftarrow{\lambda}_t^\theta(\mathbf{X})\delta t$ with $u \sim \mathcal{U}(0,1)$ **then**
    Sample $\mathbf{x}^{\text{add}}, i \sim A_t^\theta(\mathbf{x}^{\text{add}}, i|\mathbf{X})$
    $\mathbf{X} \leftarrow \text{ins}(\mathbf{X}, \mathbf{x}^{\text{add}}, i)$
  end
  $\mathbf{x} \leftarrow \mathbf{x} - \overleftarrow{\mathbf{b}}_t^\theta(\mathbf{X})\delta t + g_t\sqrt{\delta t}\epsilon$ with $\epsilon \sim \mathcal{N}(0, I_{nd})$
  $\mathbf{X} \leftarrow (n, \mathbf{x}), t \leftarrow t - \delta t$
end



Visualization of the jump-diffusion backward generative process on molecules.

Campbell et al., arXiv:2305.16261, NeurIPS 2023

Beyond Diffusion Models: Transport using ODEs

Public

- **Diffusion models**: noising "transports" π_0 to $\pi_T \approx \mathcal{N}(0, I)$, denoising does the reverse.

- **Limitations**: What if T is not large enough? What if you want π_T be non-normal? Do we need diffusions at all?

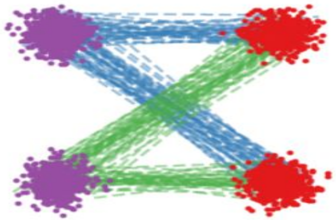
- **ODE Transport Models**: stochastic interpolants (Albergo and Vanden-Eijnden, 2023), flow matching (Lipman et al., 2023), rectified flow (Liu et al., 2023): Let $\mathbf{X}_0 \sim \pi_0$ then $\mathbf{X}_1 = \Phi(\mathbf{X}_0) \sim \pi_1$ where Φ is built using an ODE whose drift is learned using samples of π_0, π_1.

- Connections to **Optimal Transport** unclear.

Transport using ODEs: Flow Matching

- **Path Measure**: Let $\mathbf{X}_0 \sim \pi_0$, $\mathbf{X}_1 \sim \pi_1$ and

$$\mathbf{X}_t = (1-t)\mathbf{X}_0 + t\mathbf{X}_1 \sim \pi_t$$



(a) Linear interpolation
$X_t = tX_1 + (1-t)X_0$

Liu et al., "Flow Straight and Fast", ICLR 2023

- **Useless Transport ODE**: As $\mathbf{X}_t = \mathbf{X}_0 + t(\mathbf{X}_1 - \mathbf{X}_0)$ then the ODE

$$\frac{\mathrm{d}\mathbf{X}_t}{\mathrm{d}t} = \mathbf{X}_1 - \mathbf{X}_0 = \frac{\mathbf{X}_1 - \mathbf{X}_t}{1-t}, \qquad \mathbf{X}_0 \sim \pi_0,\ \mathbf{X}_1 \sim \pi_1,$$

is such that $\mathbf{X}_t \sim \pi_t$.

Transport using ODEs: Flow Matching

Public

- **Useful Transport ODE**: The ODE with drift

$$v(t, \mathbf{x}) = \mathbb{E}[\mathbf{X}_1 - \mathbf{X}_0 | \mathbf{X}_t = \mathbf{x}] = \frac{\mathbb{E}[\mathbf{X}_1 | \mathbf{X}_t = \mathbf{x}] - \mathbf{x}}{1 - t}$$

is such that $\mathbf{X}_t \sim \pi_t$!

- **Learning the Drift**: $v_{\theta^\star}(t, \mathbf{x}) = (\mathbf{x}^*_\theta(t, x) - \mathbf{x})/(1 - t) \approx v(t, \mathbf{x})$ by minimizing

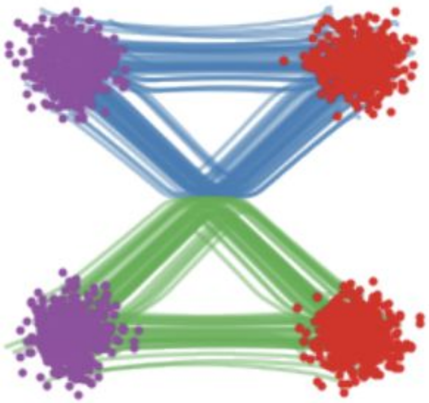
$$\mathcal{L}(\theta) = \mathbb{E}[||\mathbf{x}_\theta(t, \mathbf{X}_t) - \mathbf{X}_1||^2]$$

Liu et al.., "Flow Straight and Fast", ICLR 2023

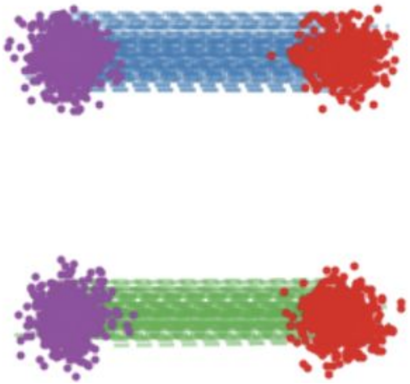
(a) Linear interpolation $X_t = tX_1 + (1 - t)X_0$ (b) Rectified flow Z_t induced by (X_0, X_1)

Transport using ODEs: Flow Matching

Public



Non-optimal transport



Optimal transport

Liu et al.., "Flow Straight and Fast", ICLR 2023

Optimal Transport

Public

- The **unpaired problem**:
  - Learn a **coupling** Π between π_0 and π_1.
  - Learn to **sample** from this coupling Π.

Optimal transport (Monge-Kantorovich formulation)

Find a **coupling** $\Pi^\star$ such that

$$\Pi^\star = \operatorname{argmin}\{\int_{\mathbb{R}^d \times \mathbb{R}^d} ||x-y||^2 \mathrm{d}\Pi(x,y)\ ,\ \Pi_0 = \pi_0,\ \Pi_1 = \pi_1\}.$$

- Sample from π_0 and then $\Pi^\star_{1|0}$ to get a sample from π_1.

- **Limitations** of the formulation:
  - Can be **unstable**.
  - Even for discrete data, **very costly**.

- Solution? **Entropic regularization**!
  - H is the **entropy**.
  - $\sigma^2 > 0$ a **regularization** parameter.

Schrödinger Bridge: From Static to Dynamic

Static formulation

$$\Pi^\star = \operatorname{argmin}\{\int_{\mathbb{R}^d \times \mathbb{R}^d} \tfrac{1}{2}\|x - y\|^2 \mathrm{d}\Pi(x, y) - \sigma^2 \mathrm{H}(\Pi)\ , \Pi_0 = \pi_0,\ \Pi_1 = \pi_1\}.$$

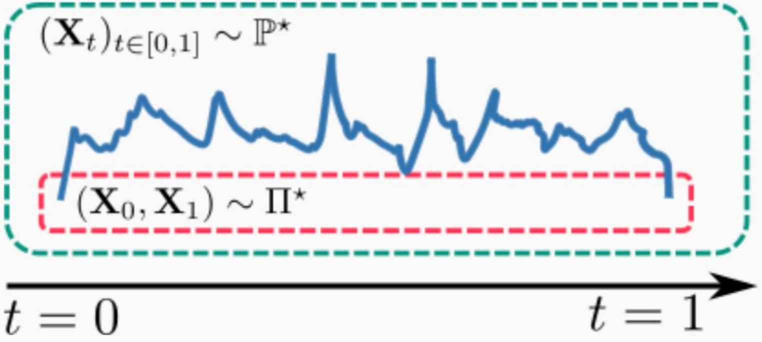
Dynamic formulation

Find $\mathbb{P}^\star$ a **path measure** such that

$$\mathbb{P}^\star = \operatorname{argmin}\{\mathrm{KL}(\mathbb{P}|\mathbb{Q})\ , \mathbb{P}_0 = \pi_0,\ \mathbb{P}_1 = \pi_1\}.$$

- $\mathbb{Q}$ is associated with $(\sigma \mathbf{B}_t)_{t\in[0,1]}$, with $(\mathbf{B}_t)_{t\in[0,1]}$ a **Brownian motion**.

- From dynamic to static: $\mathbb{P}^\star_{0,1} = \Pi^\star$.



Schrödinger Bridge Problem

Public

Dynamic formulation (Schrödinger Bridge)

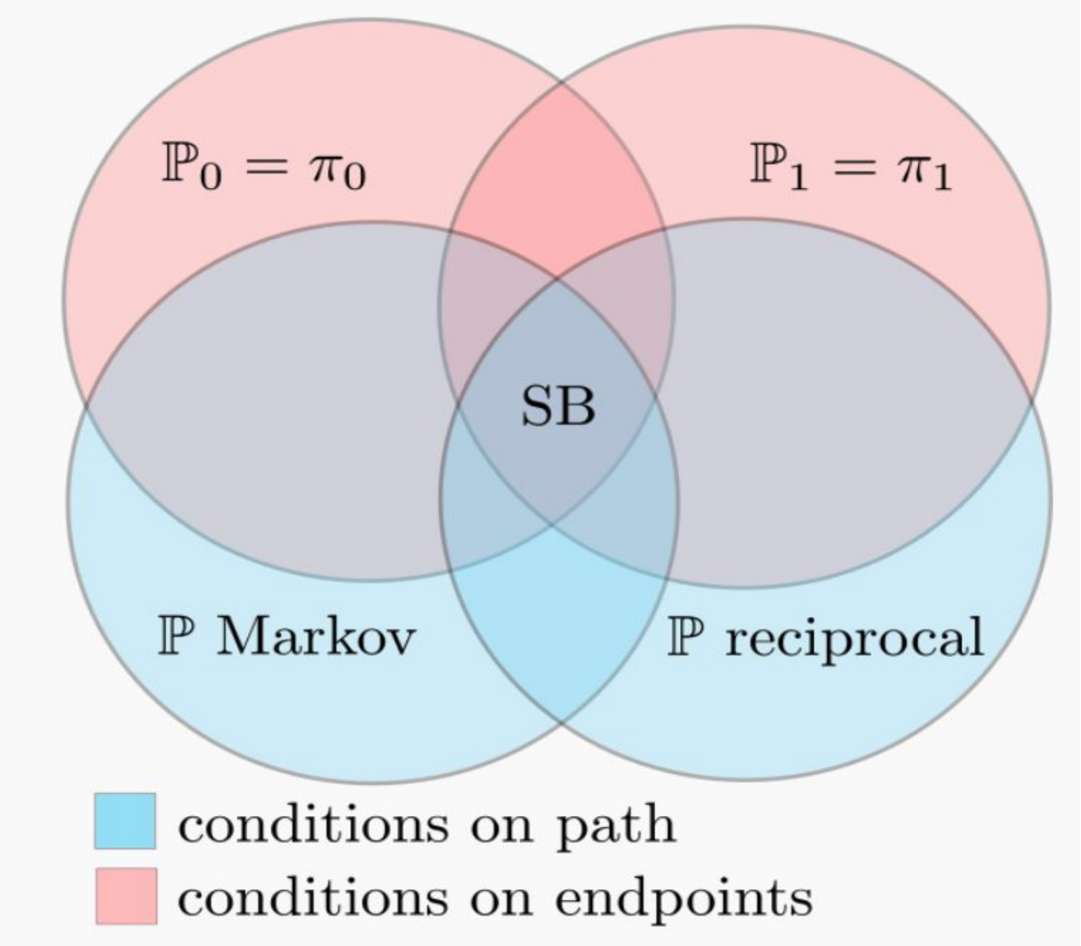
Find $\mathbb{P}^\star$ a **path measure** such that

$$\mathbb{P}^\star = \operatorname{argmin}\{\mathrm{KL}(\mathbb{P}|\mathbb{Q}), \mathbb{P}_0 = \pi_0, \ \mathbb{P}_1 = \pi_1\}.$$

- $\mathbb{Q}$ is associated with $(\sigma \mathbf{B}_t)_{t\in[0,1]}$, with $(\mathbf{B}_t)_{t\in[0,1]}$ a **Brownian motion**.

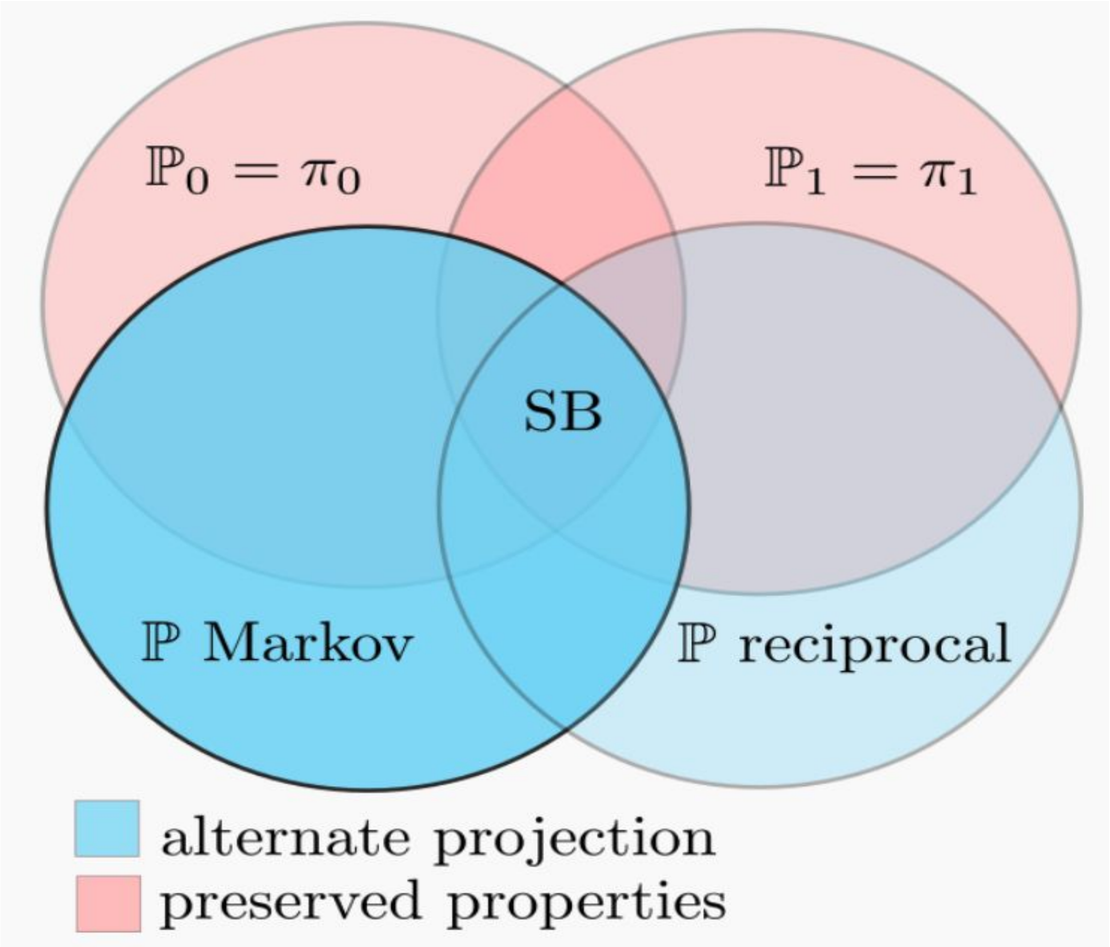
- $\mathbb{P}^\star$ is called the **Schrödinger Bridge**.
- Can be thought of:
  - **Closest path measure** to $\mathbb{Q}$ such that,
  - the **marginal constraints** are respected.
- Given $\Pi^\star$, how to sample from $\mathbb{P}^\star$?
  - Sample $(\mathbf{X}_0, \mathbf{X}_1) \sim \Pi^\star$.
  - Sample $\mathbf{X}_t = (1-t)\mathbf{X}_0 + t\mathbf{X}_1 + \sigma\sqrt{t(1-t)}\mathbf{Z}$, $\mathbf{Z} \sim \mathrm{N}(0, \mathrm{Id})$.
  - "$\mathbb{P}^\star$ is in the **reciprocal class** of $\mathbb{Q}$" (see Thieullen (1993)).

Schrödinger Bridge



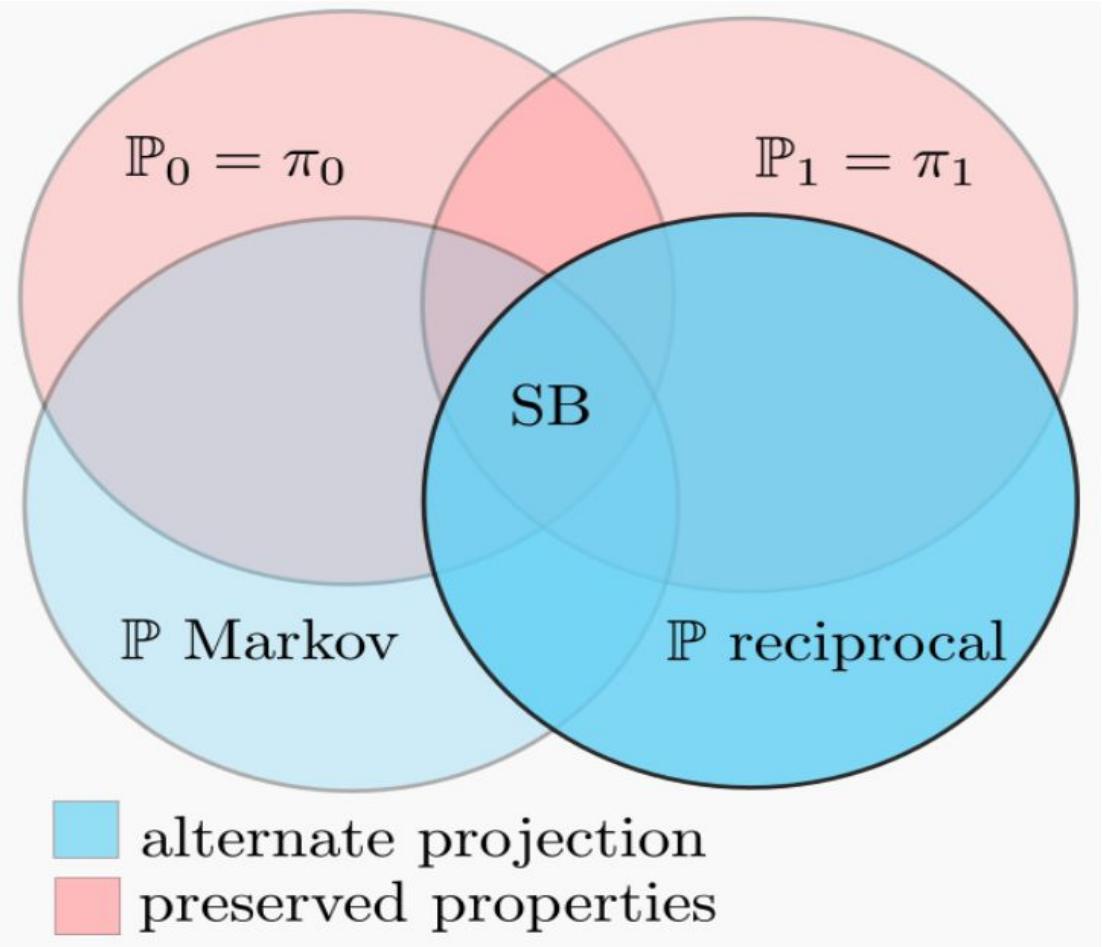
Iterative Markovian Fitting

Public



Iterative Markovian Fitting

Public



Markovian Projection

Public

- Start with $\mathbb{P}$ **non-Markov**, $(\mathbf{X}_t)_{t\in[0,1]} \sim \mathbb{P}$,

$$\mathrm{d}\mathbf{X}_t = b(t, \mathbf{X}_t, \mathbf{X}_1)\mathrm{d}t + \sigma \mathrm{d}\mathbf{B}_t$$

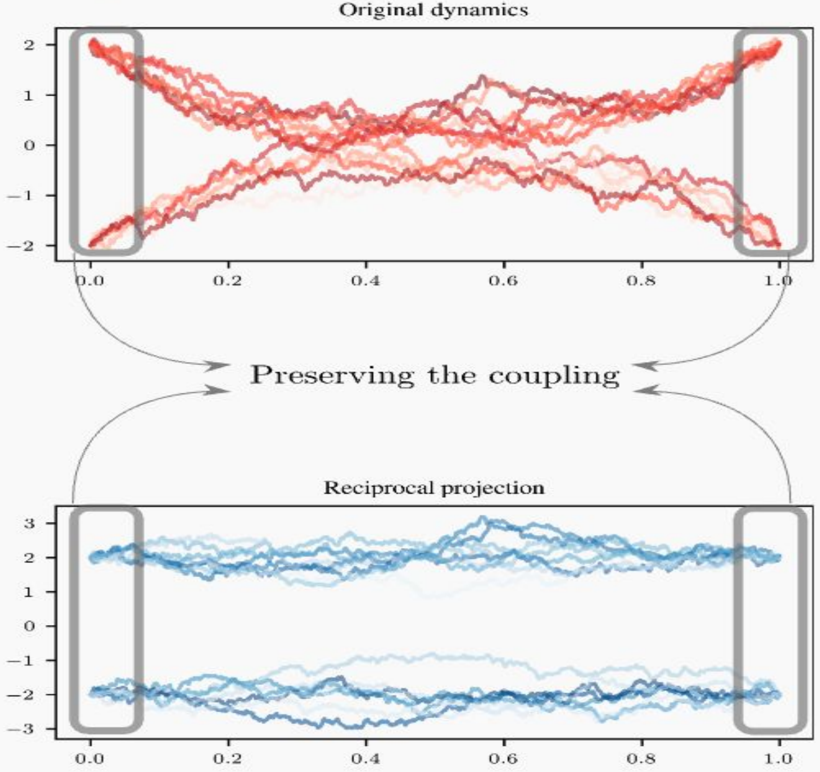
future dependency

with $b(t, \mathbf{X}_t, \mathbf{X}_1)$ linear in $\mathbf{X}_1$.

- **Markovian projection** = remove the dependency on the future.
  - Preserve **marginals** for free!
- In **practice**:
  - Loss function $||x_\theta(t, \mathbf{X}_t) - \mathbf{X}_1||^2$.
  - *At equilibrium:* $x_\theta(t, x_t) = \mathbb{E}[\mathbf{X}_1|\mathbf{X}_t]$.
  - $\mathrm{d}\mathbf{X}_t = b(t, \mathbf{X}_t, x_\theta(t, \mathbf{X}_t))\mathrm{d}t + \sigma\mathrm{d}\mathbf{B}_t$.
- The special case of **flow matching** (Lipman et al., 2022):
  - $(\mathbf{X}_0, \mathbf{X}_1) \sim \pi_0 \otimes \pi_1$, $\mathbf{X}_t = t\mathbf{X}_1 + (1-t)\mathbf{X}_0$.
  - $\mathrm{d}\mathbf{X}_t = (\mathbf{X}_1 - \mathbf{X}_t)/(1-t)\mathrm{d}t$.

Reciprocal Projection

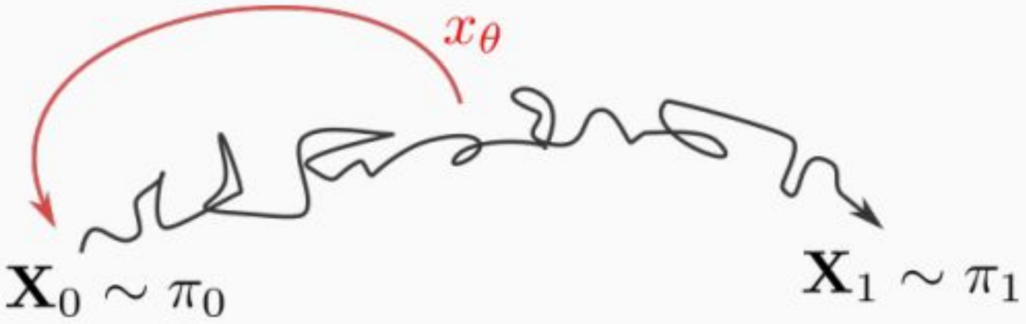
- $\mathbb{P}$ in the **reciprocal class** if for $(\mathbf{X}_t)_{t\in[0,1]} \sim \mathbb{P}$:
  - $\mathbf{X}_t = (1-t)\mathbf{X}_0 + t\mathbf{X}_1 + \sigma\sqrt{t(1-t)}\mathbf{Z}$, $\mathbf{Z} \sim \mathrm{N}(0, \mathrm{Id})$.
  - "Same bridge as the **Brownian bridge**".
- Projection on the reciprocal class (no neural network involved).



Diffusion Schrödinger Bridge Matching

Public

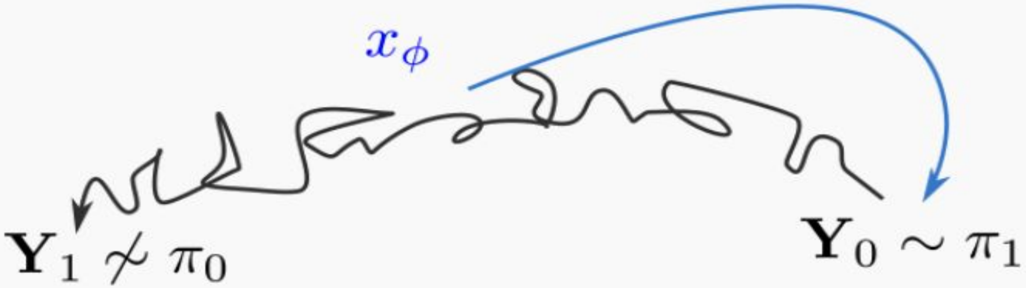
- **Diffusion Schrödinger Bridge matching**:
  - **Alternating projections** on **Markov measures** and **reciprocal class**.
  - two networks: x_θ (backward), x_ϕ (forward).
- DSBM iteration 1: **training of the backward**
  - Sample from $(\mathbf{X}_0, \mathbf{X}_1) \sim \pi_0 \otimes \pi_1$.
  - Sample $\mathbf{X}_t = (1-t)\mathbf{X}_0 + t\mathbf{X}_1 + \sigma\sqrt{t(1-t)}\mathbf{Z}$, $\mathbf{Z} \sim \mathrm{N}(0, \mathrm{Id})$.
  - Loss $\|x_\theta(1-t, \mathbf{X}_t) - \mathbf{X}_0\|^2$.



Diffusion Schrödinger Bridge Matching

- **Diffusion Schrödinger Bridge matching**:
  - **Alternating projections** on **Markov measures** and **reciprocal class**.
  - two networks: x_θ (backward), x_ϕ (forward).

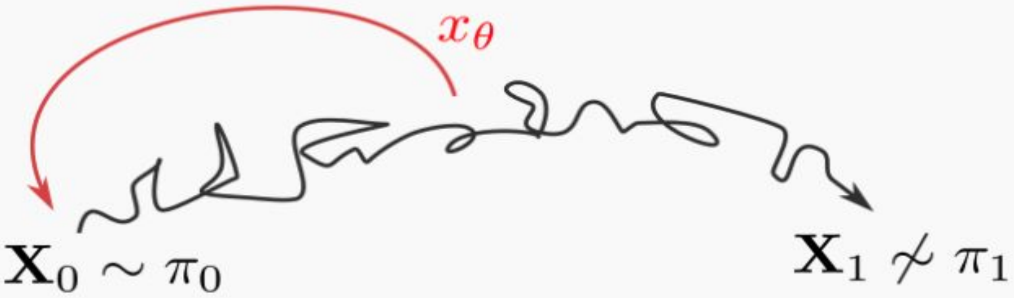
- DSBM iteration 2: **training of the forward**
  - Sample from $\mathbf{Y}_0 \sim \pi_1$, $\mathrm{d}\mathbf{Y}_t = (x_\theta(t, \mathbf{Y}_t) - \mathbf{Y}_t)/(1-t)\mathrm{d}t + \sigma \mathrm{d}\mathbf{B}_t$.
  - Keep $(\mathbf{Y}_0, \mathbf{Y}_1)$.
  - Sample $\mathbf{Y}_t = (1-t)\mathbf{Y}_0 + t\mathbf{Y}_1 + \sigma\sqrt{t(1-t)}\mathbf{Z}$, $\mathbf{Z} \sim \mathrm{N}(0, \mathrm{Id})$.
  - Loss $\|x_\phi(1-t, \mathbf{Y}_t) - \mathbf{Y}_0\|^2$.



Diffusion Schrödinger Bridge Matching

Public

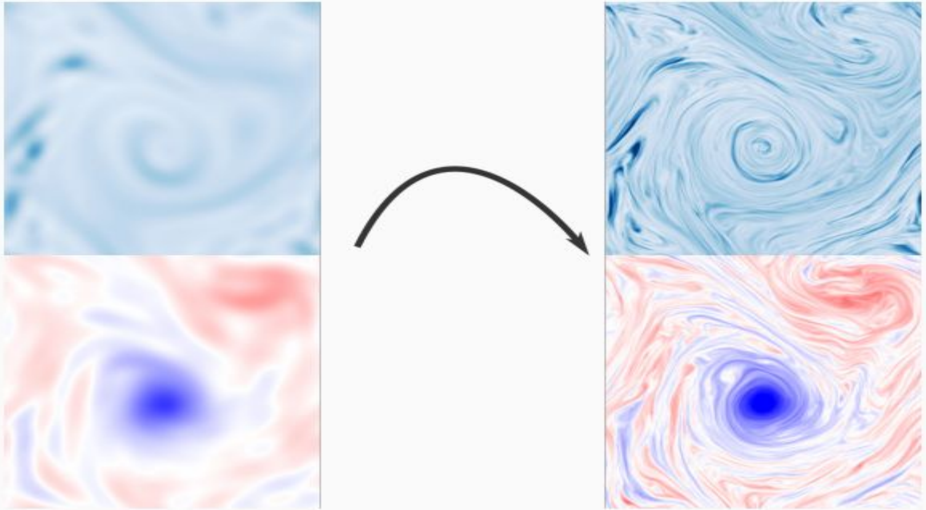
- **Diffusion Schrödinger Bridge matching**:
  - **Alternating projections** on **Markov measures** and **reciprocal class**.
  - two networks: x_θ (backward), x_ϕ (forward).
- DSBM iteration 3: **training of the backward**
  - Sample from $\mathbf{X}_0 \sim \pi_0$, $\mathrm{d}\mathbf{X}_t = (x_\phi(t, \mathbf{X}_t) - \mathbf{X}_t)/(1-t)\mathrm{d}t + \sigma \mathrm{d}\mathbf{B}_t$.
  - Keep $(\mathbf{X}_0, \mathbf{X}_1)$.
  - Sample $\mathbf{X}_t = (1-t)\mathbf{X}_0 + t\mathbf{X}_1 + \sigma\sqrt{t(1-t)}\mathbf{Z}$, $\mathbf{Z} \sim \mathrm{N}(0, \mathrm{Id})$.
  - Loss $\|x_\theta(1-t, \mathbf{X}_t) - \mathbf{X}_0\|^2$.



Climate Science Experiment

Public

- Dataset Bischoff and Deck (2023):
  - **Supersaturation** and **vorticity** field.
  - Low resolution ($64 \times 64 \times 2$) to high resolution ($512 \times 512 \times 2$).
- **Goal**: superresolution (downscaling)



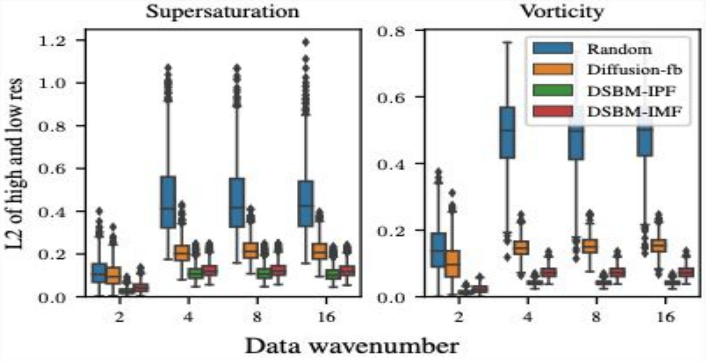
- **Unpaired problem**

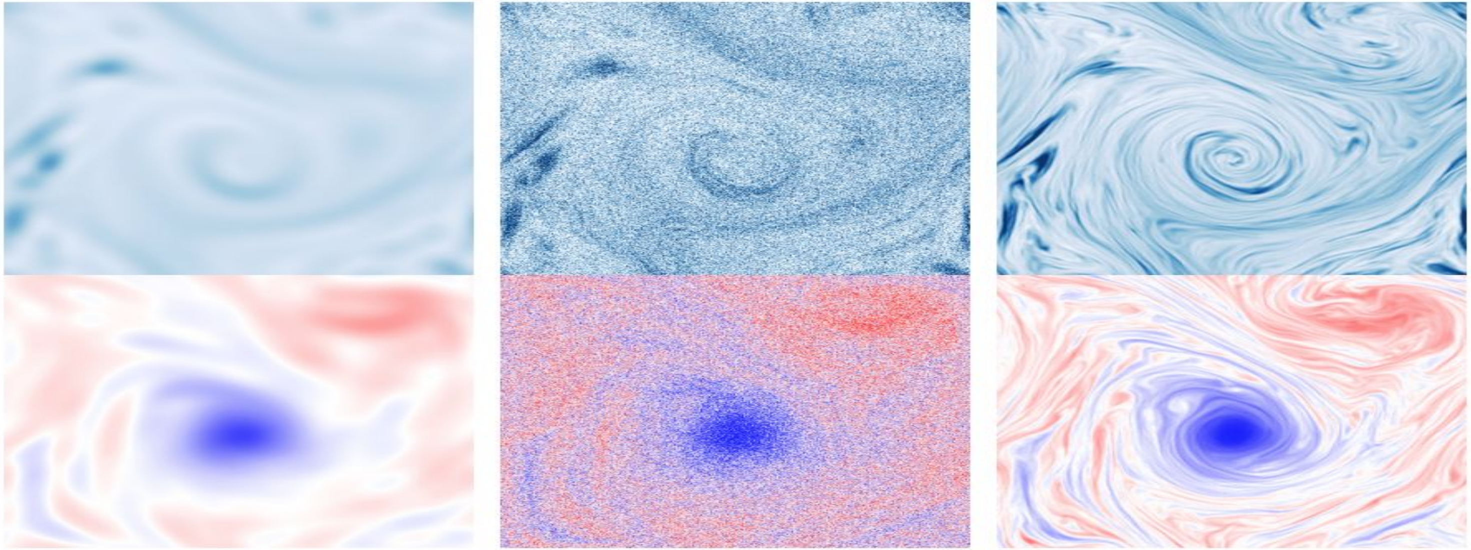
  - for downscaling tasks, paired datasets of high and low resolution climate *simulations* do not truly exist, due to deterministic chaos and the feedback of small scale motion to large scales;

Climate Science Experiment

Public

- Same setting as Bischoff and Deck (2023).
- **Super resolution** task.
- Quality measure (frequency histogram).
- Similarity measure (ℓ_2 with upscaling).





Discussion

- **Denoising Diffusion Models** provide state-of-the-art performance in numerous domains: image, audio, proteins etc.
- **Dynamic transport** alternatives are now also available.
- A lot of **open problems** at the interface of control, generative modeling, transport and sampling.