SDEs and Schrödinger Bridges

Cambridge MLG Reading Group

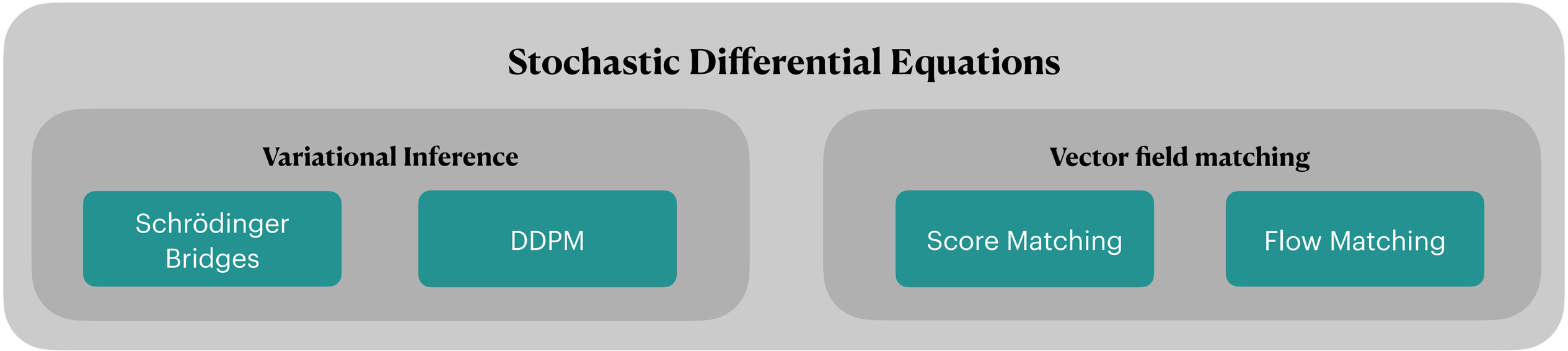
Stratis Markou and Shreyas Padhy

21 June 2023

Computational and Biological Learning
DEPARTMENT OF ENGINEERING

UNIVERSITY OF CAMBRIDGE

The Landscape



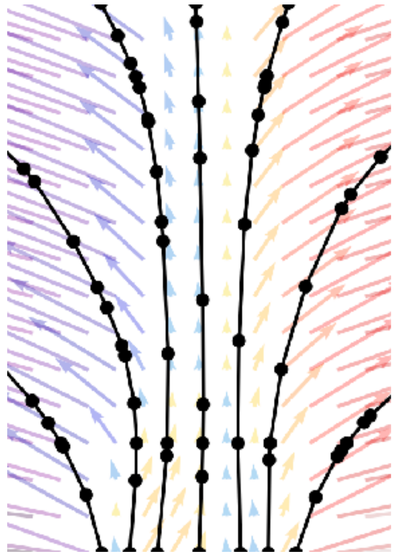
Introduction to SDEs

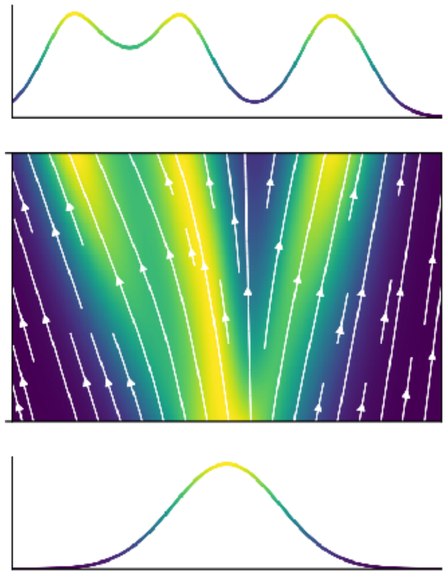
Ordinary Differential Equations

$$x_0 = x \qquad dx_t = f(x_t, t)\ dt$$

$$x_0 \sim p(x_0)$$

Continuous Normalising Flows & Neural ODEs
(Chen et al. 2018)





- CNFs: Scalability issues — $\mathcal{O}(D^2)$ complexity *can improve this to $\mathcal{O}(D)$ with approximations (FFJORD; Grathwohl et al., 2018)
- NODEs: Represent all randomness of trajectory x_t within $x_0 \sim p(x_0)$

Introduction to SDEs

Stochastic Differential Equations

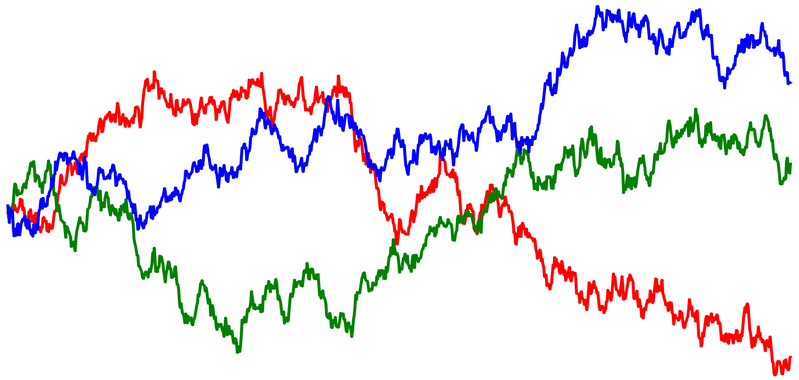
$$x_0 \sim p(x_0)$$

$$dx_t = \underbrace{f(x_t, t)}_{\text{drift}}\, dt + \underbrace{g(x_t, t)}_{\text{diffusion}}\, \underbrace{dw_t}_{\text{Brownian motion}}$$

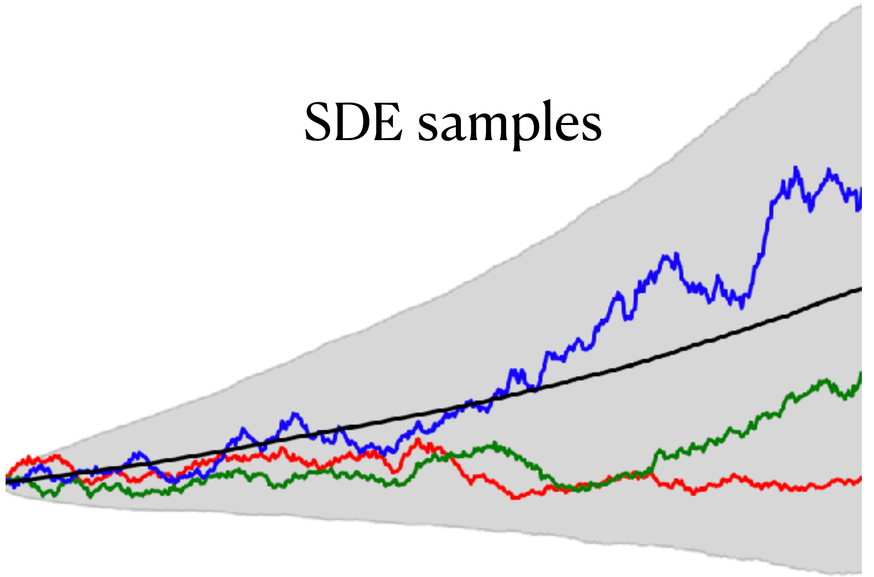
1. $w_0 = 0$
2. $w_{t_2} - w_{t_1} \sim \mathcal{N}(0,\ t_2 - t_1)$
3. $w_{t_1} \perp w_{t_3} \mid w_{t_2}$ whenever $t_1 \le t_2 \le t_3$

$$x_{t+\Delta t} = f(x_t, t)\ \Delta t + g(x_t, t)\ \Delta w_t$$

Brownian motion samples



SDE samples



- Large-scale generative models using SDEs (Song et al. 2020; Ho et al. 2020; Chen et al. 2023).
- Same x_0, different x_t : no need to encode all randomenss in x_0 (Li et al., 2020).

Stochastic Integration

Integration with SDEs is much more involved than with ODEs. An example:

$$x_0 = 0 \qquad\qquad dx_t = w_t \, dw_t$$

Let's discretise $[0, T]$ into N intervals, with $0 = t_0 < t_1 < \ldots < t_N = T$, and compute x_t using limits:

$$\begin{aligned} x_t &= \lim_{N\to\infty} \sum_{n=0}^{N-1} w_{t_n} (w_{t_{n+1}} - w_{t_n}) \\ &= \lim_{N\to\infty} \frac{1}{2} \sum_{n=0}^{N-1} w_{t_{n+1}}^2 - w_{t_n}^2 - (w_{t_{n+1}} - w_{t_n})^2 \\ &= \frac{1}{2}(w_T^2 - w_0^2) - \lim_{N\to\infty} \frac{1}{2} \sum_{n=1}^{N} (w_{t_{n+1}} - w_{t_n})^2 \\ &= \frac{1}{2} w_T^2 - \frac{1}{2} t^2 \end{aligned}$$

Itô integral

$$x_t = \sum_{n=1}^{N} w_{\frac{t_n + t_{n-1}}{2}} (w_{t_n} - w_{t_{n-1}})$$

$$= \frac{1}{2} w_T^2$$ 😬

Stratonovich integral

Stochastic Differentiation

Unsurprisingly, differentiation gives different results too. Define $y_t = h(x_t, t)$ where:

$$dx_t = f(x_t, t)\ dt + g(x_t, t)\ dw_t$$

$$\frac{dy_t}{dt} = \frac{\partial y_t}{\partial t}\ dt + \frac{\partial y_t}{\partial x_t}\ dx_t + \frac{1}{2}\frac{\partial^2 y_t}{\partial x_t^2}\ g(x_t, t)^2\ dt$$

Itô formulation

$$dx_t = f(x_t, t)\ dt + g(x_t, t) \circ dw_t$$

$$\frac{dy_t}{dt} = \frac{\partial y_t}{\partial t}\ dt + \frac{\partial y_t}{\partial x_t}\ dx_t$$

Stratonovich formulation

Fokker-Planck-Kolmogorov (FPK) equation

Given the SDE

$$dx = f(x_t, t)\ dt + g(x_t, t)\ dw_t$$

what are its marginals $p_t(x)$? No closed-form, but $p_t(x)$ follows the ODE (Särkkä and Solin; 2019):

$$\frac{dp(x_t)}{dt} = -\sum_{i=1}^{D} \frac{\partial}{\partial x_i}[p(x_t) f(x_t, t)] + \frac{1}{2}\sum_{i=1}^{D}\sum_{j=1}^{D} \frac{\partial^2}{\partial x_i \partial x_j}[p_t g_i(x_t, t) g_j(x_t, t)]$$

FPK equation

Further, if we sample $x_0 \sim p(x_0)$, and simulate the following ODE (Maoutsa et al. 2020; Song et al. 2020):

$$\frac{dx_t}{dt} = f(x_t, t) - \frac{1}{2}\left[\ \nabla g^2(x_t, t) + g^2(x_t, t)\, \nabla \log p(x_t)\ \right]$$

Probability flow ODE

this yields the same marginals as simulating the FPK.

Score-based generative models (Song et al. 2020)

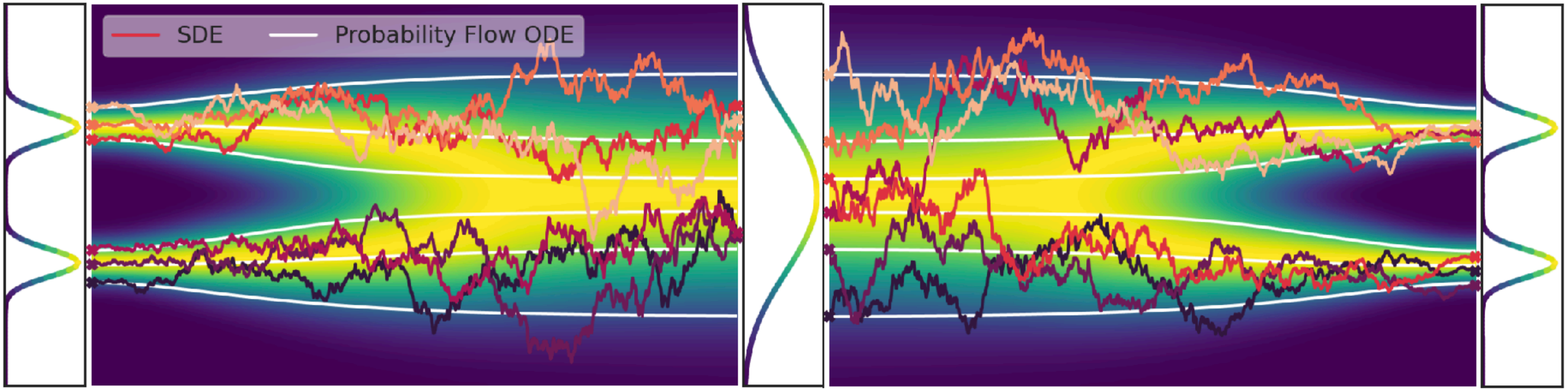
Idea: Given a *forward* SDE that mixes the data distribution $p_0(x)$ into a simple distribution $p_T(x)$ (e.g. Gaussian)

1. Sample $x_T \sim p_T(x)$,
2. Run its *reverse* SDE to obtain $x_0 \sim p_0(x)$.

$$dx = f(x_t, t)\ dt + g(t)\ dw_t$$

What is the form of the reverse SDE? (Andersson 1979)

$$dx = [f(x_t, t) - g(t)^2 \nabla \log p_t(x)]\ dt + g(t)\ d\bar{w}_t$$



Problem: The $\nabla \log p_t(x)$ term (score) does not have a nice tractable form (intuition: it depends on $p_0(x)$).

Denoising score matching (Hyvärinen, 2005; Vincent, 2010)

Reverse SDE requires access to $\nabla \log p_t(x)$. To get around this, this we will learn this from the data.

Score matching (Hyvärinen, 2005): Given a distribution $p(x_t)$ and a model distribution $p_\theta(x_t)$ with parameters θ, define the score-matching loss

$$L_{SM}(\theta) = \frac{1}{2}\mathbb{E}_{x_t \sim p(x_t)}\left[\|\nabla \log \tilde{p}_\theta(x_t) - \nabla \log p(x_t)\|_2^2\right]$$

Then, under some technical conditions

$$L_{SM}(\theta) = \mathbb{E}_{x_t \sim p(x_t)}\left[\frac{1}{2}\|\nabla \log \tilde{p}_\theta(x_t)\|_2^2 + \mathrm{Tr}\left[\nabla \nabla \log \tilde{p}_\theta(x_t)\right]\right] + \text{const.}$$

and the minimiser θ^* of $L_{SM}(\theta)$ satisfies $\tilde{p}_{\theta^*}(x_t) = p(x_t)$.

This is still no good because the trace of the Hessian (second term) is computationally costly.

Denoising score matching (Vincent, 2010): It holds that

$$L_{DSM}(\theta) := \frac{1}{2}\mathbb{E}_{x_0 \sim p(x_0)\ x_t \sim p(x_t|x_0)}\left[\|\nabla \log \tilde{p}_\theta(x_t) - \nabla \log p(x_t \,|\, x_0)\|_2^2\right] = L_{SM}(\theta) + \text{const.}$$

Training score-based diffusion models

Parameterise $\nabla \log \tilde{p}_{\theta,t}(x)$ by a neural network $s_\theta(x, t)$, and minimise the loss

$$L_{DSM}(\theta) = \frac{1}{2}\mathbb{E}_{x_0 \sim p(x_0)\ t \sim U[0,T]\ x_t \sim p(x_t|x_0)} \left[\lambda(t)\ \|s_\theta(x_t, t) - \nabla \log p(x_t \,|\, x_0)\|_2^2 \right].$$

by evaluating an unbiased Monte Carlo estimate of L_{DSM}

1. Sample the data distribution $x_0 \sim p(x_0)$,
2. Draw corrupted version of the data $x_t \sim p(x_t \,|\, x_0)$ using the forward SDE,
3. Evaluate empirical estimate of L_{DSM} and take gradients.

So far we have not discussed how to pick the forward SDE. It's essential that:

- The forward SDE mixes to our simple distribution $p(x_T \,|\, x_0) \approx p(x_T)$.
- Sampling $x_t \sim p(x_t \,|\, x_0)$ is tractable and inexpensive.

We can satisfy the above by using a corruption SDE with Gaussian marginals:

$$dx_t = f(t)\ x\ dt + g(t)\ dw_t$$

Choosing the forward (corruption) SDE

Two popular choices for the forward corruption process:

- The Variance Preserving SDE (VPSDE):

$$dx_t = -\frac{1}{2}\beta(t)\ x\ dt + \sqrt{\beta(t)}\ dw_t$$

with $\beta(t) \geq 0$, has the conditional distribution

$$p(x_t \mid x_0) = \mathcal{N}\left(x_t; \sqrt{1-\alpha(t)}\ x_0, \alpha(t)I\right), \text{ where } \alpha(t) = 1 - e^{-\int_0^t \beta(t')dt'}$$

- The Variance Exploding SDE (VESDE):

$$dx_t = \sqrt{\frac{d\sigma^2(t)}{dt}}\ dw_t$$

with $\beta(t) \geq 0$, has the conditional distribution

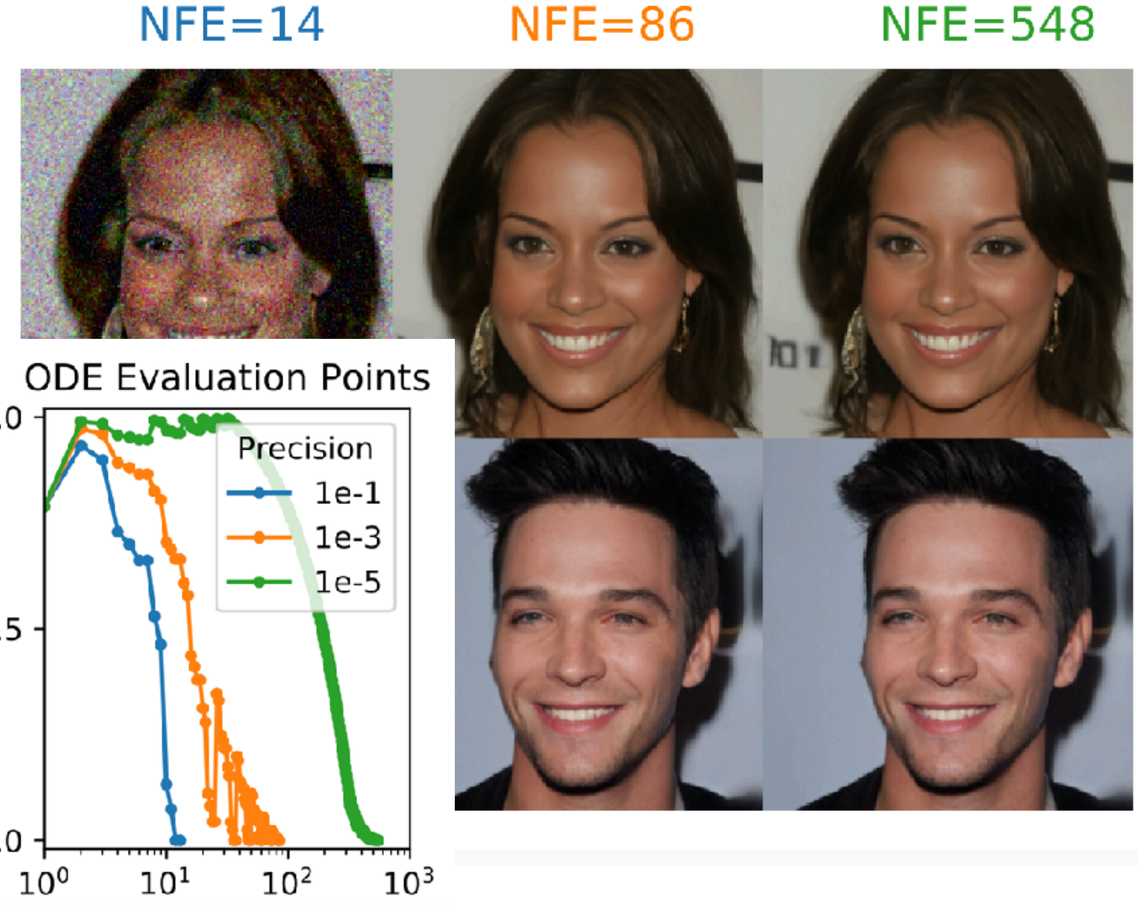
$$p(x_t \mid x_0) = \mathcal{N}\left(x_t; x_0, \sigma^2(t)I\right).$$

Sampling with score based diffusion models

Having learnt $s_\theta(x_t, t)$, we can use it to draw samples:

$$dx_t = [f(t) - g(t)^2 s_\theta(t)]\ dt + g(t)\ d\bar{w}_t$$

$$dx_t = \left[f(t) - \frac{1}{2} g(t)^2 s_\theta(x_t, t) \right]\ dt$$



Conditional generation: classifier guidance

In conditional generation, we are interested in sampling from $p_0(x \mid y)$.

Classifier guidance:

- Train a classifier $p(y \mid x_t)$, mapping noisy data x_t to distributions over labels y.
- Use classifier to *guide* an unconditional score model:

$$
\begin{aligned}
dx_t &= \left[f(t) - g^2(t) \nabla \log p(x_t \mid y) \right] dt + g(t)\, d\bar{w}_t \\
&= \left[f(t) - g^2(t) [\nabla \log p(x_t) + \nabla \log p(y \mid x_t)] \right] dt + g(t)\, d\bar{w}_t \\
&\approx \left[f(t) - g^2(t) [s_\theta(x_t, t) + \nabla \log p(y \mid x_t)] \right] dt + g(t)\, d\bar{w}_t
\end{aligned}
$$

Can also use tempering, i.e. use $\nabla \log p(x_t) + \gamma \nabla \log p(y \mid x_t)$ instead, to enhance the effect of the classifier:

from Dhariwal and Nichol (2021)

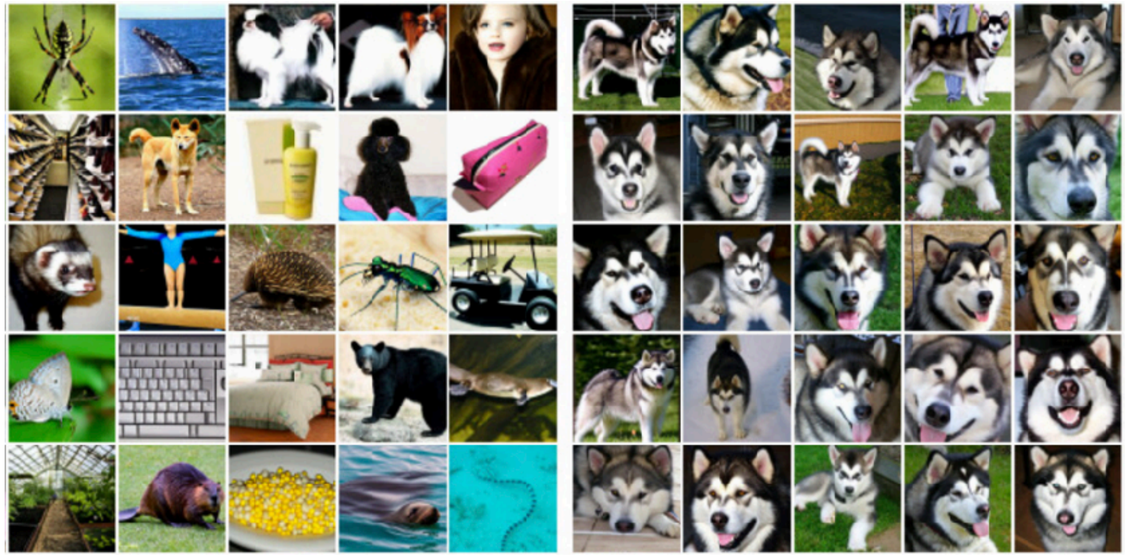
Conditional generation: classifier-free guidance

Can avoid classifiers (Ho and Salimans 2021) by providing conditioning y (some of the time) at training-time:

$$L_{CF-DSM}(\theta) = \frac{1}{2}\mathbb{E}_{x_0 \sim p(x_0)\ t \sim U[0,T]\ x_t \sim p(x_t|x_0)} \left[\lambda(t) \ \| s_\theta(x_t, y, t) - \nabla \log p(x_t \,|\, x_0) \|_2^2 \right] .$$

The minimiser of L_{CF-DSM} yields correct samples from the conditional distribution (Batzolis et al. 2021).

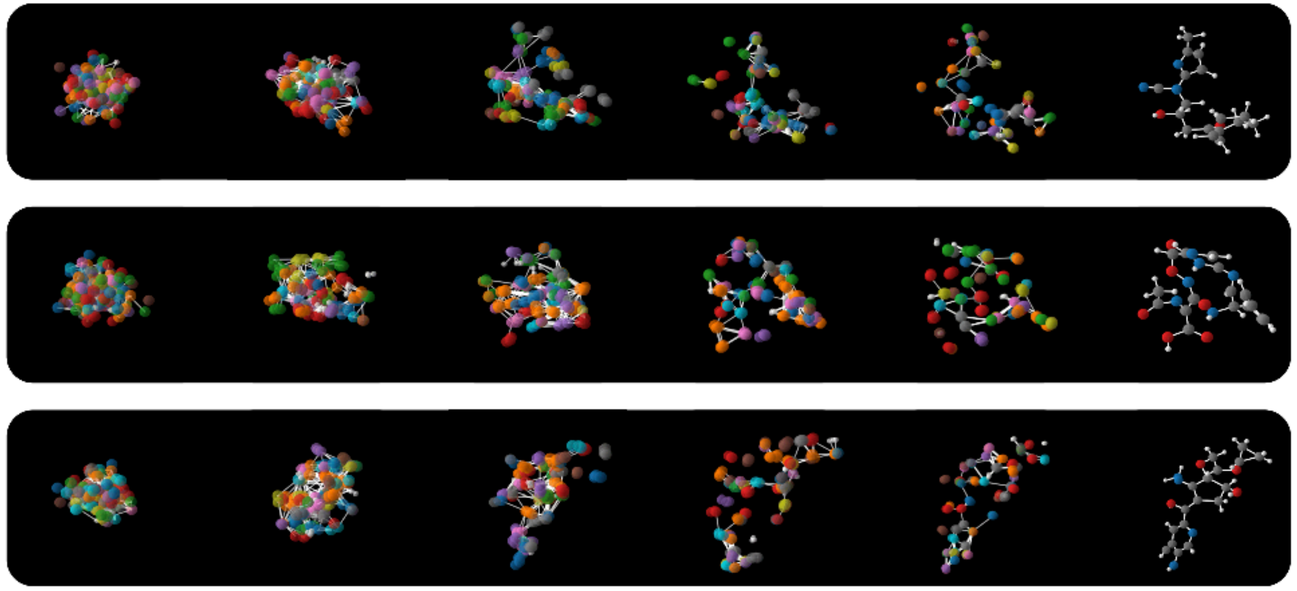
Classifier-free guidance is also amenable to tempering:



from Ho and Salimans (2021)

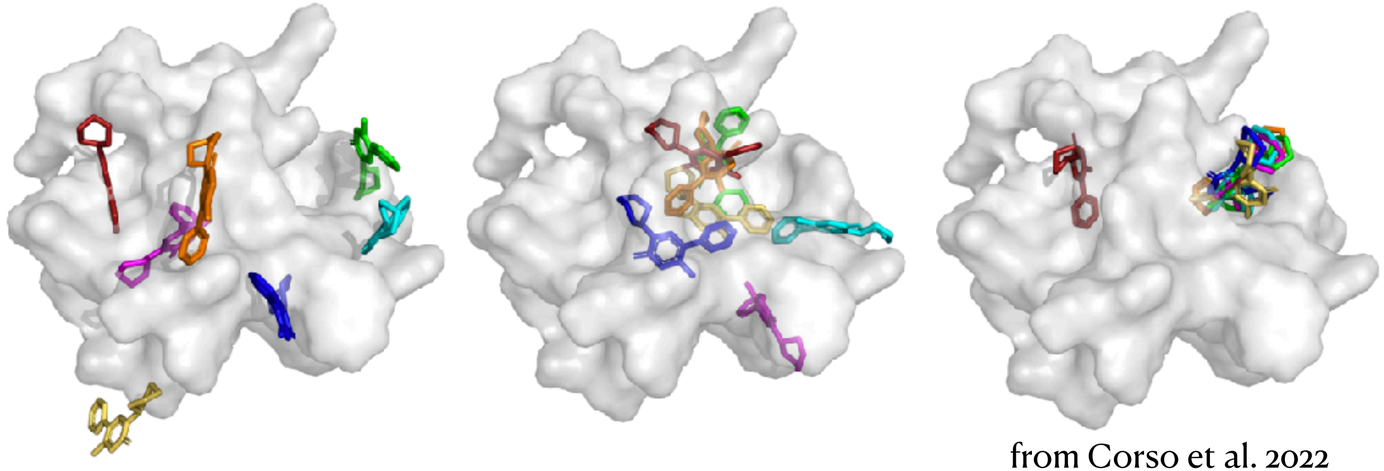
Applications — beyond images

Generating new molecules



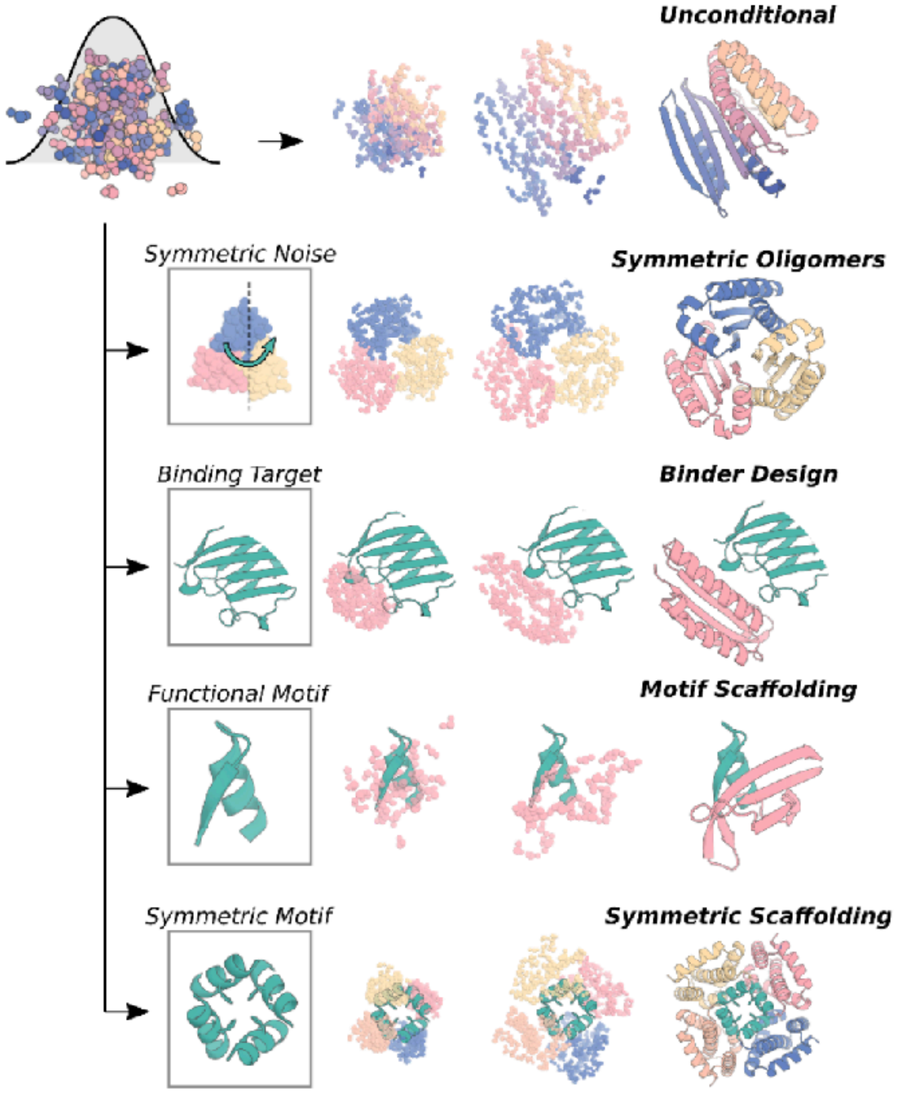
from Hoogenboom et al. 2022

Predicting how molecules will bind to proteins



from Corso et al. 2022

Generating protein conformations



from Watson et al. 2022

Flow matching (Lipman et al. 2022)

Idea: use ideas from score matching to scale continuous normalising flows.

Given a vector field $u_t(x)$. We can define the *flow* $\phi_t(x)$ of $u_t(x)$ as the solution to the ODE

$$\frac{d}{dt}\phi_t(x) = u_t(\phi_t(x))$$

$$\phi_T(x) = x$$

Now suppose that $u_t(x)$ is such that

$$x_T \sim p_T(x), \quad \frac{d}{dt}\phi_t(x_T) = u_t(\phi_t(x_T)) \implies x_0 \sim p_0(x),$$

then we could draw samples by simulating the ODE. We don't have $u_t(x)$ so we will learn it:

$$\mathscr{L}_{FM}(\theta) = \frac{1}{2}\mathbb{E}_{t\sim U[0,T],\ x\sim p_t(x)}\left[\|v_{\theta,t}(x) - u_t(x)\|_2^2\right]$$

Flow matching (Lipman et al. 2022)

Analogously to score matching, we don't have access to $u_t(x)$:

$$\mathscr{L}_{FM}(\theta) = \frac{1}{2}\mathbb{E}_{t\sim U[0,T],\ x_t\sim p_t(x)}\left[\|v_t(x_t) - u_t(x_t)\|_2^2\right]$$

Instead, consider a conditional field $u_t(x \mid x_0)$ giving rise to marginal distributions $p_t(x \mid x_0)$ such that

$$p_T(x \mid x_0) = p_T(x), \quad p_0(x \mid x_0) \approx \delta(x - x_0)$$

Conditional flow matching (Lipman et al., 2022): It holds that

$$L_{CFM}(\theta) := \frac{1}{2}\mathbb{E}_{t\sim U[0,T],\ x_0\sim p_0(x),\ x_t\sim p_t(x|x_0)}\left[\|v_t(x_t) - u_t(x_t \mid x_0)\|_2^2\right] = L_{FM}(\theta) + \text{const.}$$

This can be easily estimated, just like the DSM loss. But how should we pick $u_t(x \mid x_0)$?

Flow matching (Lipman et al. 2022)

We want: Field $u_t(x \mid x_0)$ which gives rise to

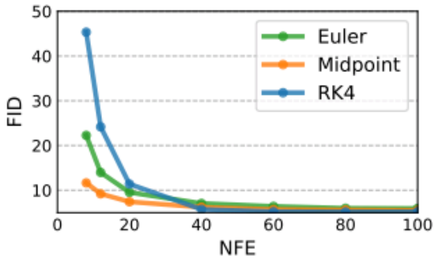
$$p_T(x \mid x_0) = p_T(x), \quad p_0(x \mid x_0) \approx \delta(x - x_0)$$

Pick a conditional flow $\phi_t(x \mid x_0)$, to satisfy the above. For example, consider the *optimal transport* flow

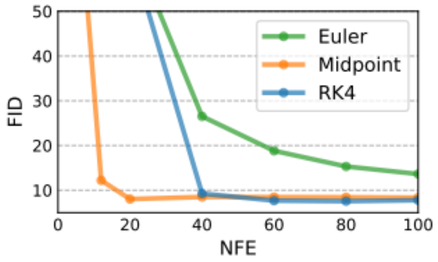
$$\phi_t(x \mid x_0) = 1 - (1 - \sigma_{\min})t'x + t'x_0, \quad t' = 1 - \frac{t}{T}$$

Flow matching formulation includes the VPSDE and VESDE probability flow ODEs as special cases.

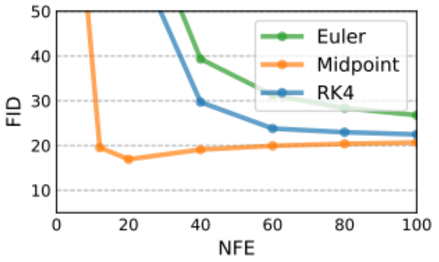
	CIFAR-10			ImageNet 32×32			ImageNet 64×64		
Model	NLL↓	FID↓	NFE↓	NLL↓	FID↓	NFE↓	NLL↓	FID↓	NFE↓
Ablations									
DDPM	3.12	7.48	274	3.54	6.99	262	3.32	17.36	264
Score Matching	3.16	19.94	242	3.56	5.68	178	3.40	19.74	441
ScoreFlow	3.09	20.78	428	3.55	14.14	195	3.36	24.95	601
Ours									
FM ^{w}/ Diffusion	3.10	8.06	183	3.54	6.37	193	3.33	16.88	187
FM ^{w}/ OT	**2.99**	**6.35**	**142**	**3.53**	**5.02**	**122**	**3.31**	**14.45**	**138**



Flow matching ^{w}/ OT



Flow matching ^{w}/ Diffusion



Score matching ^{w}/ Diffusion

Denoising diffusion probabilistic models (Ho et al 2021)

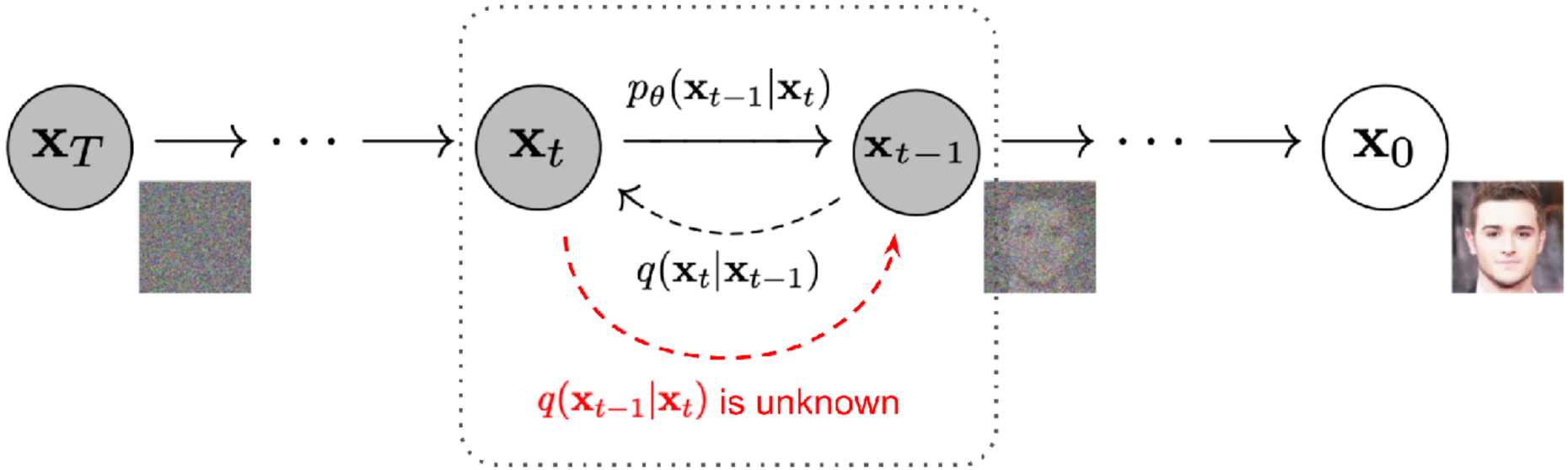
Idea: Given a *forward* diffusion process that adds noise to points $x_0 \sim q_0(x)$ until Gaussian noise at $x_T \sim (0,I)$

1. Forward diffusion process is defined by $q(x_t \mid x_{t-1}) = \mathcal{N}\left(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t \mathbf{I}\right)$
2. Learn the reverse diffusion process from data by approximating $q(x_{t-1} \mid x_t)$ with a variational $p_\theta(x_{t-1} \mid x_t)$

Loss Function: Minimise the cross-entropy (maximise ELBO)

$$L_{\text{CE}} = -\,\mathbb{E}_{q(x_0)} \log p_\theta\left(x_0\right) \leq \mathbb{E}_{q(x_{0:T})}\left[\log \frac{q\left(x_{1:T} \mid x_0\right)}{p_\theta\left(x_{0:T}\right)}\right] = L_{\text{VLB}}$$

Use variational lower bound



DDPM is a discretised SDE

The forward diffusion process is $q(x_t \mid x_{t-1}) = \mathcal{N}\left(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t \mathbf{I}\right)$

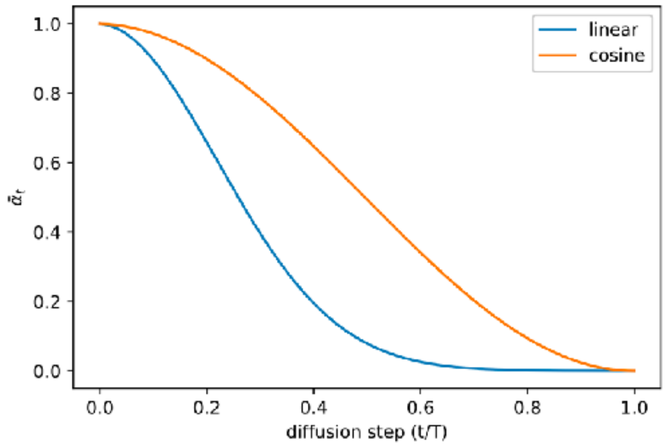
$$x_t = \sqrt{1-\beta_t}x_{t-1} + \sqrt{\beta_t}\epsilon_{t-1}, \qquad t = 1, \ldots, T$$

As $T \to \infty$, the discrete Markov process converges to the following SDE

$$\mathrm{d}x = -\frac{1}{2}\beta(t)x\mathrm{d}t + \sqrt{\beta(t)}\mathrm{d}w, \qquad \text{which is the variance-preserving SDE [Song et al 2020] !}$$

Minimising the cross-entropy L_{CE} is equivalent to score-matching from the Song formulation.

Noise Schedules: [Nichol and Dhariwal 2021] picking different forms of $\beta(t)$ can improve training and sampling performance.



Generative Modeling through a Coupling Perspective (Vargas et al 2023)

Consider the generating process $x_T \sim p_T(x_T), \quad x_0 \mid x_T \sim p_\theta(x_0 \mid x_T)$ to obtain $x_0 \sim \int p_\theta(x_0 \mid x_T) p_T(x_T) \mathrm{d}x_T$

and a "reversed" process $x_0 \sim p_0(x_0), \quad x_T \mid x_0 \sim q_\phi(x_T \mid x_0)$.

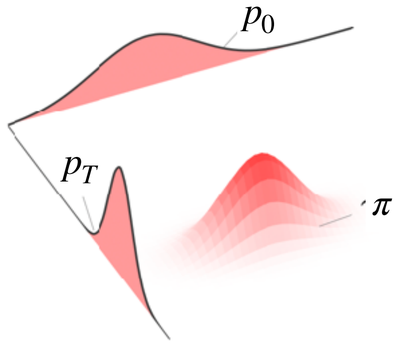
These processes are "reversals" if the joint distribution is equal, i.e. $p_T(x_T)p_\theta(x_0 \mid x_T) = p_0(x_0)q_\phi(x_T \mid x_0) = \pi(x_0, x_T)$

Furthermore $p_0(x_0) = \int p_\theta(x_0 \mid x_T) p_T(x_T) \mathrm{d}x_T$, and $p_T(x_T) = \int q_\phi(x_T \mid x_0) p_0(x_0) \mathrm{d}x_0$

For training, minimise a divergence b/w the joints, $L_D(\theta, \phi) = D\left(p_0(x_0)q_\phi(x_T \mid x_0) \,||\, p_T(x_T)p_\theta(x_0 \mid x_T)\right)$

The set of minimisers associated with $L_D(\theta, \phi)$ is the set of all probabilistic couplings between p_T and p_0

$$\pi \in \mathscr{P} = \left\{ \pi \geq 0, \int \pi(x_0, x_T)\mathrm{d}x_T = p_0(x_0), \int \pi(x_0, x_T)\mathrm{d}x_0 = p_T(x_T) \right\}$$



Product Coupling $\pi = p_0 \otimes p_T$

DDPM from a Variational Perspective

$$L_D(\theta, \phi) = D\left(p_0(x_0)q_\phi(x_T \mid x_0) \,||\, p_T(x_T)p_\theta(x_0 \mid x_T)\right)$$

1. Hierarchical Variational Distributions with discrete latent variables

Let us restrict the family of distributions we consider to hierarchical models with intermediate latent variables $\{x_1, x_2, \ldots, x_{T-1}\}$.

Assume $q_\phi(x_T, x_{T-1}, \ldots, x_1 \mid x_0) = \prod_{t=1}^{T} q_{\phi_{t-1}}\left(x_t \mid x_{t-1}\right)$

And $p_\theta(x_0, x_1, \ldots, x_{T-1} \mid x_T) = \prod_{t=1}^{T} p_{\theta_t}\left(x_{t-1} \mid x_t\right)$

Assume Gaussian

Fix q_{ϕ_t} and ϕ_t to some choice of Markov process, for example $q_{\phi_t}(x_t \mid x_{t-1}) = \mathcal{N}\left(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t \mathbf{I}\right)$

We can't obtain $q_\phi(x_T \mid x_0), p_\theta(x_0 \mid x_T)$ in closed-form without marginalising out $\{x_1, \ldots, x_{T-1}\}$ (Data-processing inequality)

Instead, consider $D_{KL}\left(p_0(x_0)q_\phi(x_{1:T} \mid x_0) \,||\, p_T(x_T)p_\theta(x_{0:T-1} \mid x_T)\right) \geq D\left(p_0(x_0)q_\phi(x_T \mid x_0) \,||\, p_T(x_T)p_\theta(x_0 \mid x_T)\right)$

Then we have $\theta = \arg\min L_D(\theta) = \arg\min D_{KL}\left(p_0(x_0)q_\phi(x_{1:T} \mid x_0) \,||\, p_T(x_T)p_\theta(x_{0:T-1} \mid x_T)\right)$ (This is exactly DDPM)

SDEs from a Variational Perspective (Vargas et al 2023)

2. Hierarchical Variational Distributions with infinite latents

Consider the extension from finite latent variables to an infinite number of latent variables.

The discrete $\{x_{0:T}\}$ now result in continuous paths x_t, and with the Gaussian assumptions we made before, the forward and reverse path dynamics of x_t from $[0,T]$ and $[T,0]$ are now given by SDEs -

$$\mathrm{d}x_t = f_\phi(x_t)\mathrm{d}t + \sigma \mathrm{d}w_t, \qquad x_0 \sim p_0(x_0) \qquad\qquad p_0(x_0)q_\phi(x_{1:T} \mid x_0) \to \mathbb{Q}_\phi$$

$$\mathrm{d}x_t = g_\theta(x_t)\mathrm{d}t + \sigma \mathrm{d}\bar{w}_t, \qquad x_T \sim p_T(x_T) \qquad\qquad p_T(x_T)p_\theta(x_{0:T-1} \mid x_T) \to \mathbb{P}_\theta$$

If we assume $f_\phi(x_t) = -\alpha x_t$, and impose the time-reversal property for SDEs, $g_\theta(x_t) = -\alpha x_t - \sigma^2 \nabla p_t(x)$.

For $\alpha = 0$, $g_\theta(x_t)$ a score-matching network $\to$ VE-SDEs (Song et al 2021)

For $\alpha > 0$, $g_\theta(x_t)$ a score-matching network $\to$ VP-SDEs (Ho et al 2020, Song et al 2021)

$$D_{KL}\left(p_0(x_0)q_\phi(x_{1:T} \mid x_0) \,||\, p_T(x_T)p_\theta(x_{0:T-1} \mid x_T)\right) \to \arg\min_\theta D_{KL}(\mathbb{Q}_\phi \,||\, \mathbb{P}_\theta) \quad = \quad \text{score-matching}$$

Theoretical Guarantees of Convergence for SDEs

- Consider the SDE $\mathrm{d}x = -\alpha x \mathrm{d}t + \sigma \mathrm{d}w, \quad x_0 \sim p_{\text{data}}, \quad x_T \sim p_{\text{noise}}$
- Assume we use a parametric model $s_\theta(x, t)$ to fit the score, and that the error is bounded, i.e.
  - $\left\| s_{\theta\star}(t, x) - \nabla \log p_t(x) \right\| \leq \mathrm{M} \quad \text{for some } M \geq 0$
  - During sampling, we solve the reverse SDE by discretising $[T,0] \rightarrow [\gamma_T, \gamma_{T-1}, \ldots, \gamma_0]$
- Then the following bounds hold in total variation [De Bortoli et. al. 2021] -
  - For $\alpha > 0$, we have $\left\| \mathscr{L}\left(X_0\right) - p_{\text{data}} \right\|_{\text{TV}} \leq C_\alpha \left(\mathrm{M} + \bar{\gamma}^{1/2}\right) \exp\left[D_\alpha T\right] + B_\alpha \exp\left[-\alpha^{1/2} T\right]$
  - For $\alpha = 0$, we have $\left\| \mathscr{L}\left(X_0\right) - p_{\text{data}} \right\|_{\text{TV}} \leq C_0 \left(\mathrm{M} + \bar{\gamma}^{1/2}\right) \exp\left[D_0 T\right] + B_0 \left(T^{-1} + T^{-1/2}\right)$
- Where $\mathscr{L}(X_0)$ is the obtained empirical data distribution, and T is the total time interval that the SDE is solved for, $\bar{\gamma} = \max \gamma_k$, and $B_\alpha, C_\alpha, D_\alpha \rightarrow \infty$ as $\alpha \rightarrow \infty$.

Schrödinger Bridges

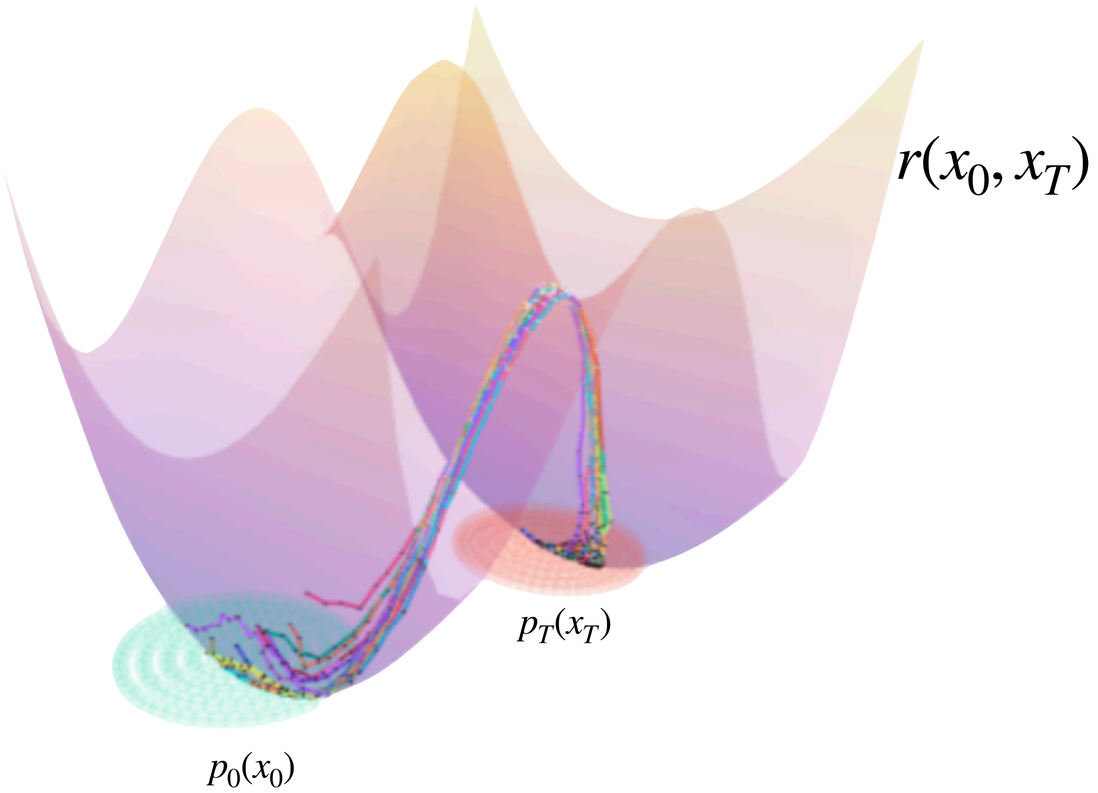
We wish to find a coupling density $\pi^\star$ such that

$$\pi^*(x_0, x_T) \in \underset{\pi(x_0, x_T)}{\arg\min} \left\{ D_{\mathrm{KL}}(\pi(x_0, x_T) \| r(x_0, x_T)) : \pi_{x_0} = p_0(x_0), \pi_{x_T} = p_T(x_T) \right\}$$

Where $r(x_0, x_T)$ is a physically or biologically motivated, reference process.

Diffusion Schrodinger Bridges [Vargas et al. 2021, de Bortoli et al. 2021] -

- The reference density is path measure $\mathbb{Q}$ corresponding to an SDE, i.e. drift-augmented Brownian motion, OU Process
- Discrete analogue is also possible, where reference density is the Markov process $q_\phi(x_{0:T}) = p_0(x_0) q_\phi(x_{1:T} \mid x_0)$

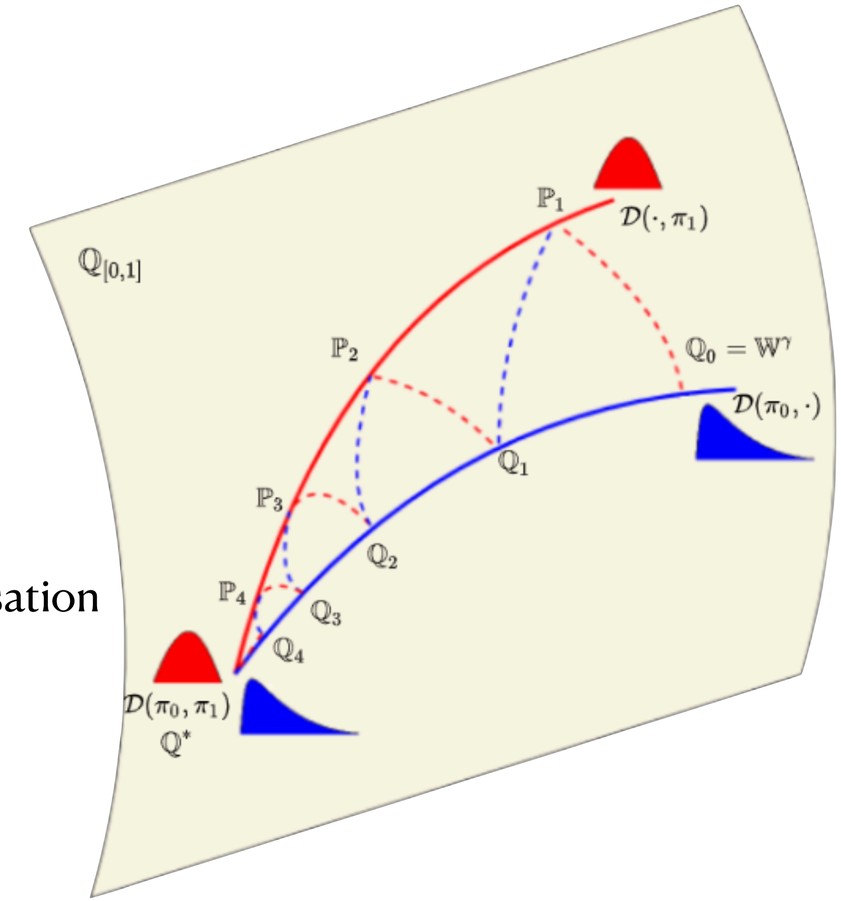


Solving Schrödinger Bridges

The Iterative Proportional Fitting (IPF) algorithm solves

$$\pi^*(x_0, x_T) \in \underset{\pi(x_0,x_T)}{\arg\min} \left\{ D_{\mathrm{KL}}(\pi(x_0, x_T) \| r(x_0, x_T)) : \pi_{x_0} = p_0(x_0), \pi_{x_T} = p_T(x_T) \right\}$$

1. Choose $\pi_0 = r(x_0, x_t)$
2. Perform $\pi^{2n+1} = \arg\min \left\{ \mathrm{KL}\left(\pi \mid \pi^{2n}\right) : \pi_{x_T} = p_T \right\}$
3. Perform $\pi^{2n+2} = \arg\min \left\{ \mathrm{KL}\left(\pi \mid \pi^{2n+1}\right) : \pi_{x_0} = p_0 \right\}$



Remember the coupling loss $L_D(\theta, \phi) = D\left(p_0(x_0)q_\phi(x_T \mid x_0) \,||\, p_T(x_T)p_\theta(x_0 \mid x_T)\right)$

Proposition [Vargas et. al 2023]: If the coupling loss is solved using Expectation-Maximisation (EM) with a KL divergence as follows:

$$\theta_{n+1} = \underset{\theta}{\arg\min} \mathscr{L}_{D_{\mathrm{KL}}}\left(\phi_n, \theta\right),$$

$$\phi_{n+1} = \underset{\phi}{\arg\min} \mathscr{L}_{D_{\mathrm{KL}}}\left(\phi, \theta_{n+1}\right)$$

Then, for a suitable initialisation of IPF, the IPF iterates agree with the EM iterations

$$\pi^n = q^{\phi_{(n-1)/2}}(x_T \mid x_0)p_0(x_0), \quad \text{for } n \text{ odd}$$

$$\pi^n = p^{\theta_{n/2}}(x_0 \mid x_T)p_T(x_T), \quad \text{for } n \text{ even}$$

Solving Schrodinger's Bridges

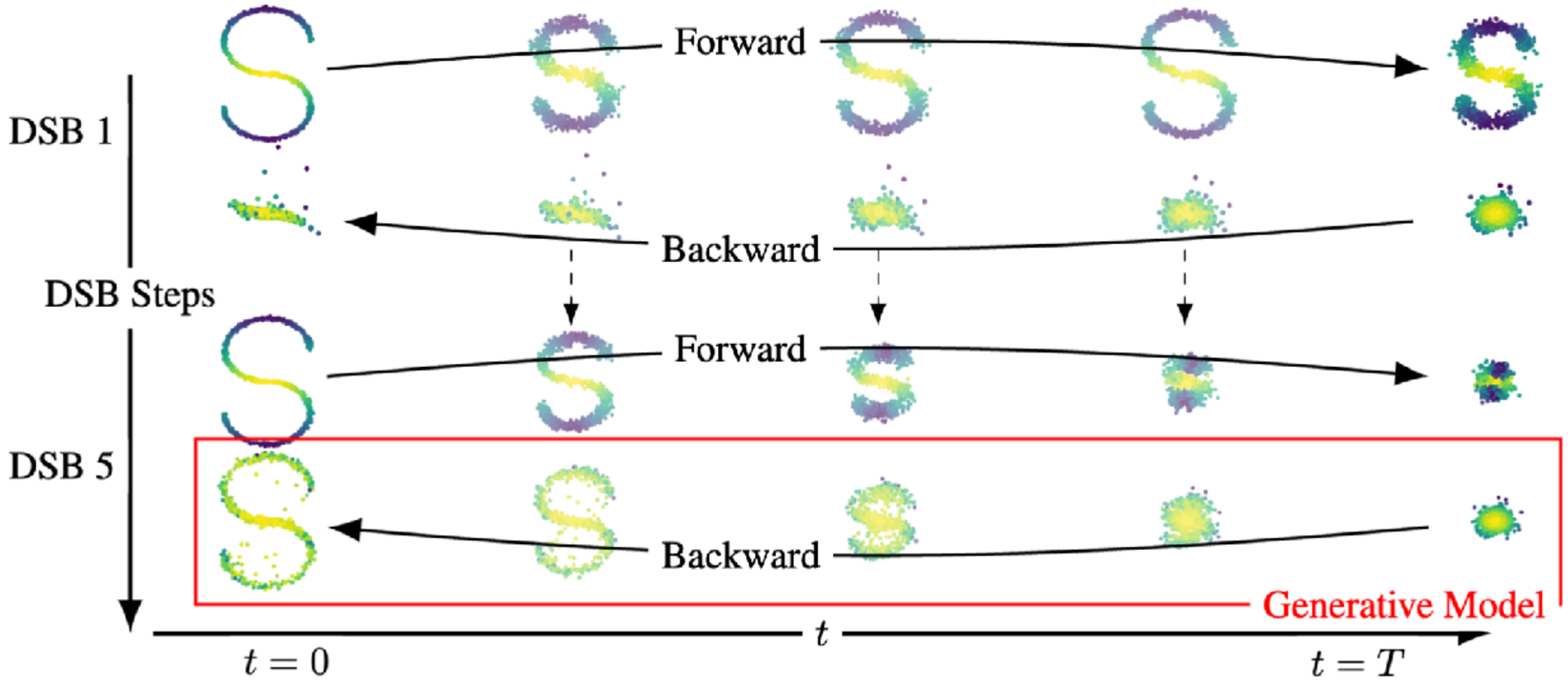
1.The first step of a SB solves the score-matching objective exactly, i.e. $\theta^\star = \arg\min_\theta \mathscr{L}_{D_{\mathrm{KL}}}(\phi_n, \theta)$

2.The second step "fixes" the denoising process so that the constraint $\pi_{x_0} = p_0(x_0)$ is enforced.

3.The third step solves the score-matching objective to the "fixed" denoising process and enforces $\pi_{x_T} = p_T(x_T)$

4.The fourth step "fixes" the denoising process again so that the constraint $\pi_{x_0} = p_0(x_0)$ is enforced.

- Each "score-matching" step doesn't need to run for long T or high mixing time
- During generation, samples are obtained from p_0 in finite T



A unifying perspective of variational inference, diffusions, and Optimal Transport

- [Vargas et al 2023] unify many formulations under the variational inference view
  - Score-based generative modelling [Song et al 2021, Ho et al 2021]
  - Score-based sampling
    - ergodic drift [Vargas et al 2023, Berner et al 2022]
    - Follmer drift [Follmer, 1984; Vargas et al., 2021a; Zhang and Chen, 2021a; Huang et al., 2021b]
  - Domain adaptation and stochastic filtering [Reich and Cotter, 2015]
  - Score-based annealed flows [Heng et al., 2015; 2020; Arbel et al., 2021; Doucet et al., 2022]
  - Schrodinger Bridges & Entropic Optimal Transport [de Bortoli et al 2021; Vargas et al 2021]

References

References I

[1] Song, Yang, et al. “Score-based generative modeling through stochastic differential equations.” *arXiv preprint arXiv:2011.13456* (2020).

[2] Vargas, Francisco, et al. “Transport, Variational Inference and Diffusions”, ICML 2023 Workshop Frontiers4LCD.

[3] De Bortoli, Valentin, et al. “Diffusion Schrödinger bridge with applications to score-based generative modeling.” *Advances in Neural Information Processing Systems* 34 (2021): 17695-17709.”

[4] Vargas, F., Thodoroff, P., Lamacraft, A., & Lawrence, N. (2021). Solving Schrödinger bridges via maximum likelihood. *Entropy*, *23*(9), 1134.

[5] Chen, R. T., Rubanova, Y., Bettencourt, J., & Duvenaud, D. K. (2018). Neural ordinary differential equations. *Advances in neural information processing systems*, *31*.

[6] Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, *33*, 6840-6851.

[7] Pavon, M., Tabak, E. G., & Trigila, G. (2018). The data-driven Schroedinger bridge. In *arXiv [math.OC]*. arXiv. http://arxiv.org/abs/1806.01364

[8] Song, Y., & Ermon, S. (2019). Generative Modeling by Estimating Gradients of the Data Distribution. In *arXiv [cs.LG]*. arXiv. http://arxiv.org/abs/1907.05600

[9] Nichol, A., & Dhariwal, P. (2021). Improved Denoising Diffusion Probabilistic Models. In *arXiv [cs.LG]*. arXiv. http://arxiv.org/abs/2102.09672

[10] Maoutsa, D., Reich, S., & Opper, M. (2020). Interacting particle solutions of fokker–planck equations through gradient–log–density estimation. *Entropy*, *22*(8), 802.

References II

[1] Hyvärinen, A., & Dayan, P. (2005). Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, *6*(4).

[2] Vincent, P. (2011). A connection between score matching and denoising autoencoders. *Neural computation*, *23*(7), 1661-1674.

[3] Dhariwal, P., & Nichol, A. (2021). Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, *34*, 8780-8794.

[4] Ho, J., & Salimans, T. (2022). Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*.

[5] Lipman, Y., Chen, R. T., Ben-Hamu, H., Nickel, M., & Le, M. (2022). Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*.

[6] Berner, J., Richter, L., & Ullrich, K. (2022). An optimal control perspective on diffusion-based generative modeling. *arXiv preprint arXiv:2211.01364*.

[7] Föllmer, H., & Wakolbinger, A. (1986). Time reversal of infinite-dimensional diffusions. *Stochastic processes and their applications*, *22*(1), 59-77.

[8] Vargas, F., Ovsianas, A., Fernandes, D., Girolami, M., Lawrence, N. D., & Nüsken, N. (2023). Bayesian learning via neural Schrödinger–Föllmer flows. *Statistics and Computing*, *33*(1), 3.

[9] Zhang, Q., & Chen, Y. (2021). Path integral sampler: a stochastic control approach for sampling. *arXiv preprint arXiv:2111.15141*.

[10] Huang, J., Jiao, Y., Kang, L., Liao, X., Liu, J., & Liu, Y. (2021). Schrödinger-Föllmer sampler: sampling without ergodicity. *arXiv preprint arXiv:2106.10880*.

References III

[1] Huang, J., Jiao, Y., Kang, L., Liao, X., Liu, J., & Liu, Y. (2021). Schrödinger-Föllmer sampler: sampling without ergodicity. *arXiv preprint arXiv:2106.10880*.

[2] Reich, S., & Cotter, C. (2015). Probabilistic forecasting and Bayesian data assimilation. Cambridge University Press.

[3] Heng, J., Doucet, A., & Pokern, Y. (2021). Gibbs flow for approximate transport with applications to Bayesian computation. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *83*(1), 156-187.

[4] Heng, J., Bishop, A. N., Deligiannidis, G., & Doucet, A. (2020). Controlled sequential monte carlo.

[5] Arbel, M., Matthews, A., & Doucet, A. (2021, July). Annealed flow transport monte carlo. In *International Conference on Machine Learning* (pp. 318-330). PMLR.

[6] Vargas, F., Grathwohl, W., & Doucet, A. (2023). Denoising diffusion samplers. *arXiv preprint arXiv:2302.13834*.

Appendix

Proving the Existence of the Inverse SDE

Idea: Given a *forward* SDE that mixes the data distribution $p_0(x)$ into a simple distribution $p_T(x)$ (e.g. Gaussian)

$$dx = f(x_t, t)\ dt + g(t)\ dw_t \qquad\qquad dx = [f(x_t, t) - g(t)^2 \nabla \log p_t(x)]\ dt + g(t)\ d\bar{w}_t$$

Proof Sketch: Let us assume a discretised form of the forward SDE, $x_{t_{k+1}} = x_{t_k} + f(x_{t_k}, t_k)\delta t + g(t_k)\epsilon_{t_k}\sqrt{\delta t}$

We wish to enforce $p_{t|t+\delta}(x \mid y)p_{t+\delta}(y) = p_{t+\delta|t}(y \mid x)p_t(x)$

$$p_{t|t+\delta}(x \mid y) = \mathcal{N}\left(x \mid y + f^+(y)\delta, \delta\sigma^2\right)$$
$$p_{t+\delta|t}(y \mid x) = \mathcal{N}\left(x \mid y + f^-(y)\delta, \delta\sigma^2\right)$$

Take the log on both sides and rearrange

$$\left(f^+(x) + f^-(y)\right)^\top (y - x) + \delta\left(\left\| f^+(x) \right\|^2 + \left\| f^-(y) \right\|^2\right) = \sigma^2\left(\ln p_{t+\delta}(y) - \ln p_t(x)\right)$$

Taking the limit $\delta \to 0$

$$\left(f^+(x) + f^-(y)\right)^\top (y - x) = \sigma^2\left(\ln p_t(y) - \ln p_t(x)\right)$$

Proving the Existence of the Inverse SDE

Proof Sketch:

Taking the limit $\delta \to 0$

$$\left(f^{+}(x) + f^{-}(y)\right)^{\top}(y - x) = \sigma^{2}\left(\ln p_t(y) - \ln p_t(x)\right)$$

Applying Taylor's Expansion

$$\begin{aligned}\left(f^{+}(x) + f^{-}(y)\right)^{\top}(y - x) &= \ln p_t(y) - \ln p_t(x) \\ &= \sigma^{2}\nabla \ln p_t(x)^{\top}(y - x) + h(y)^{\top}(y - x)\end{aligned}$$

Simplifying, we get

$$f^{+}(y) + f^{-}(y) = \sigma^{2}\nabla \ln p_t(y)$$

Therefore, we get

$$\mathrm{d}Y_t = \left(\sigma_t^{2}\nabla \ln p_{T-t}\left(Y_t\right) - f^{+}\left(Y_t, T - t\right)\right)\mathrm{d}t + \sigma_t\ \mathrm{d}W_t$$