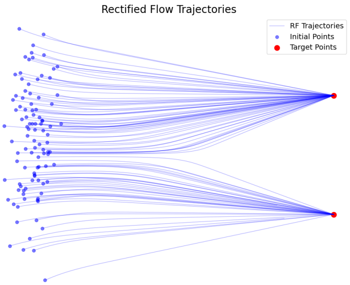
Diffusion = ODE + Langevin

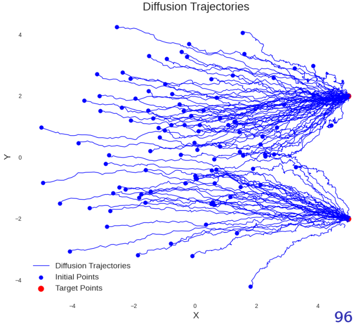
$$\mathrm{d}Z_t = \underbrace{v_t(Z_t)\,\mathrm{d}t}_{\text{Rectified Flow}} + \underbrace{\sigma_t^2 \nabla \log \rho_t(Z_t)\,\mathrm{d}t + \sqrt{2}\,\sigma_t\,\mathrm{d}W_t}_{\text{Langevin Dynamics}}$$

Training Free Conversion: ODE → SDE

- **ODE (Flow)**: Generate data Z_1 by solving
$$\mathrm{d}Z_t = v(Z_t, t)\,\mathrm{d}t.$$

- **SDE (Diffusion)**: Generate data Z_1 by solving
$$\mathrm{d}Z_t = v(Z_t, t)\,\mathrm{d}t + \sigma_t\,\mathrm{d}W_t,$$
where W_t is standard Brownian motion.





Diffusion = ODE + Langevin

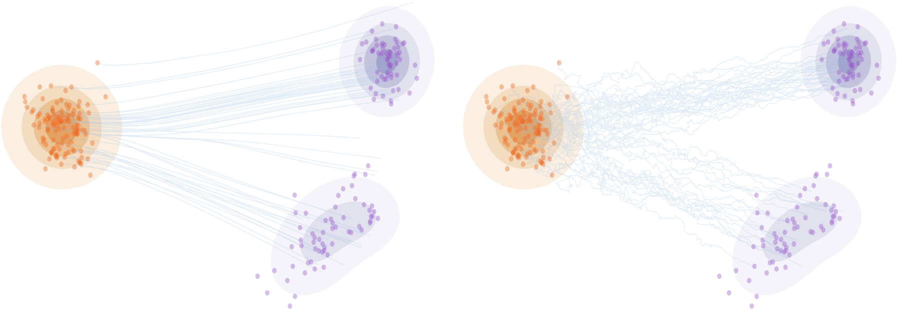
- Assume ρ_t is the density of Z_t following ODE $\dot{Z}_t = v_t(Z_t)$.
- We can convert ODE into SDE while **preserving marginals**:

$$\mathrm{d}Z_t = \underbrace{v_t(Z_t)\,\mathrm{d}t}_{\text{Rectified Flow}} + \underbrace{\sigma_t^2 \nabla \log \rho_t(Z_t)\,\mathrm{d}t + \sqrt{2}\,\sigma_t\,\mathrm{d}W_t}_{\text{Langevin Dynamics}}$$

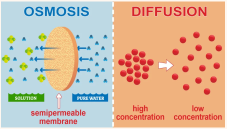
- Langevin term is
  - Self consistent: $\rho_t = \texttt{density of } Z_t$
  - hence in equilibrium all time
  - changes the dynamics, but does not modify the marginals.

Diffusion = ODE + Langevin

$$\mathrm{d}Z_t = \underbrace{v_t(Z_t)\,\mathrm{d}t}_{\text{Rectified Flow}} + \underbrace{\sigma_t^2 \nabla \log \rho_t(Z_t)\,\mathrm{d}t + \sqrt{2}\,\sigma_t\,\mathrm{d}W_t}_{\text{Langevin Dynamics}}$$



Fokker–Planck Equation.



For an Itô SDE of form

$$\mathrm{d}Z_t = v_t(Z_t)\,\mathrm{d}t + \sigma_t\,\mathrm{d}W_t\,,$$

the density ρ_t of Z_t follows

$$\partial_t \rho_t(x) = -\nabla\cdot\left(v_t^{\text{eff}}(x)\,\rho_t(x)\right),$$

$$v_t^{\text{eff}}(x) = v_t(x) - \underbrace{\frac{\sigma_t^2}{2}\nabla \log \rho_t(x)}_{\text{``osmosis'' force}}.$$

- Therefore, $\mathrm{d}Z_t = v_t(Z_t)\mathrm{d}t + \sigma_t \mathrm{d}W_t$ and $\mathrm{d}\tilde{Z}_t = v_t^{eff}(\tilde{Z}_t)\mathrm{d}t$ share the same marginals.

De-randomization of Ito SDEs

- In the literature, the ODE methods were first obtained by de-randomizing SDEs:
  - DDPM $\to$ DDIM
  - Time-reverse SDEs $\to$ Probability Flow ODEs
  - This direction feels more mysterious.

$$\mathrm{d}X_t = v_t(X_t)\mathrm{d}t + \sigma_t \mathrm{d}W_t \quad \stackrel{\text{de-randomization}}{\Longrightarrow} \quad \mathrm{d}Z_t^* = v_t^{\mathit{eff}}(Z_t^*)\mathrm{d}t.$$

$\{Z_t^*\}$ yields "best" deterministic approximation that preserves marginals:

$$\min_{\{Z_t\}} \mathbb{E}\left[\left\|v_t(Z_t) - \dot{Z}_t\right\|\right]^2 \quad s.t. \quad \mathrm{Law}(X_t) = \mathrm{Law}(Z_t), \quad \forall t \in [0,1].$$

It minimizes the actional functional, while preserving the marginal distributions.

Proof. [Use Orthogonality of Helmholtz decomposition]. First, the optimal $\{Z_t\}$ must be an ODE using Jensen's inequality. Hence, assume $\dot{Z}_t = \mu_t(Z_t)$.

To ensure that $\mathrm{Law}(Z_t) = \mathrm{Law}(X_t)$, we must have

$$\mu_t(x) = v_t^{eff}(x) + r_t(x), \qquad \nabla \cdot (r_t(x)\rho_t(x)) = 0,$$

where $r_t(x)$ is a divergence-free field that does not contribute to the change of marginals.

With integration by parts,

$$\mathbb{E}\left[r_t(X_t)\nabla f(X_t)\right] - \int f(x)\nabla \cdot (\rho_t(x) r_t(x))\mathrm{d}x = 0, \quad \forall f.$$

Hence, $$\|\mu_t(X_t) - v_t(X_t)\|^2 = \left\| v_t^{eff}(X_t) - v_t(X_t) \right\|^2 + \|r_t(X_t)\|^2$$

Hence $r_t = 0$ achieves minimum.

Marginal Preservation: Three Approaches

Three marginal preservation approaches so far:

- **Derandomization**:
  - removing or adding **diffusion** noise
- **Markovization**:
  - removing or adding **memory**
- **Helmholtz decomposition**:
  - removing or adding **rotational** fields

A Marginal Preserving Chain:

`Non-Markov SDE` $\rightarrow$ `Non-Markov ODE` $\rightarrow$ `Markov` $\rightarrow$ `Gradient flow.`

- Transport cost decreases $\searrow$ along the chain.

Diffusion = ODE + Langevin

$$\mathrm{d}Z_t = \underbrace{v_t(Z_t)\,\mathrm{d}t}_{\text{Rectified Flow}} + \underbrace{\sigma_t^2 \nabla \log \rho_t(Z_t)\,\mathrm{d}t + \sqrt{2}\,\sigma_t\,\mathrm{d}W_t}_{\text{Langevin Dynamics}}$$

For Gaussian noise X_0, independent coupling $X_0 \perp\!\!\!\perp X_1$, using Tweedie's formula,

$$\nabla \log \rho_t(x) = \frac{t v_t(x) - x}{1 - t},$$

the SDE can be fully determined by v_t (without estimating ρ_t):

$$\mathrm{d}Z_t = \left(v_t(Z_t) + \frac{\sigma_t^2}{1-t}(t v_t(Z_t) - Z_t) \right) \mathrm{d}t + \sqrt{2}\sigma_t \mathrm{d}W_t.$$

Diffusion = ODE + Langevin

$$\mathrm{d}Z_t = \underbrace{v_t(Z_t)\,\mathrm{d}t}_{\text{Rectified Flow}} + \underbrace{\sigma_t^2 \nabla \log \rho_t(Z_t)\,\mathrm{d}t + \sqrt{2}\,\sigma_t\,\mathrm{d}W_t}_{\text{Langevin Dynamics}}$$

For Gaussian noise X_0, independent coupling $X_0 \perp\!\!\!\perp X_1$, using Tweedie's formula,

$$\nabla \log \rho_t(x) = \frac{t v_t(x) - x}{1 - t},$$

the SDE can be fully determined by v_t (without estimating ρ_t):

$$\mathrm{d}Z_t = \left(v_t(Z_t) + \frac{\sigma_t^2}{1-t}(t v_t(Z_t) - Z_t) \right) \mathrm{d}t + \sqrt{2}\sigma_t \mathrm{d}W_t.$$

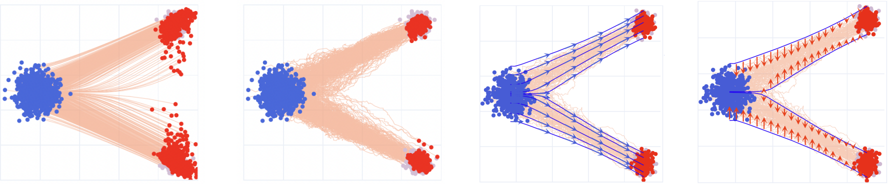
DDPM (upto time scaling) chooses $\sigma_t^2 = \frac{1-t}{t}$:

$$\mathrm{d}Z_t = \left(2 v_t(Z_t) - \frac{1}{t} Z_t \right) \mathrm{d}t + \sqrt{2\frac{1-t}{t}}\mathrm{d}W_t.$$

Why Add Noise? Langevin as Error Corrector

$$\mathrm{d}Z_t = \underbrace{v_t(Z_t)\,\mathrm{d}t}_{\text{Rectified Flow}} + \underbrace{\sigma_t^2 \nabla \log \rho_t(Z_t)\,\mathrm{d}t + \sqrt{2}\,\sigma_t\,\mathrm{d}W_t}_{\text{Langevin Dynamics}}$$

- Langevin dynamics can act as an error correction mechanism, nudging trajectories back toward high-density regions.



Deterministic ODE (with large v_t error) | ODE + Langevin | Velocity v_t (evolving forward) | Score $\nabla \log \rho_t$ (mode seeking)

- The mode-seeking behavior of $\nabla \log \rho_t$ is balanced by diffusion noise, preserving marginals.

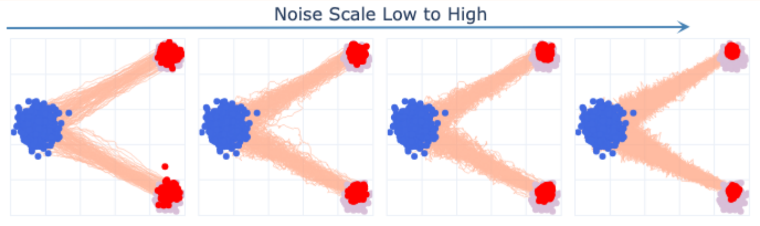
Artifacts of Diffusion: Over-Concentration

- But no free lunch: when the estimated velocity $\hat{v}_t$ is wrong, the score $\nabla \log \hat{\rho}_t$ is also off. Errors amplify with large σ_t.
- Singularity shows up:

$$\nabla \log \hat{\rho}_t(x) = \frac{t\hat{v}_t(x) - x}{1 - t}.$$

- $\nabla \log \hat{\rho}_t(x) \to \infty$ as $t \to 1$, unless the boundary condition $\hat{v}_t(x) \to x$ as $t \to 1$ holds.

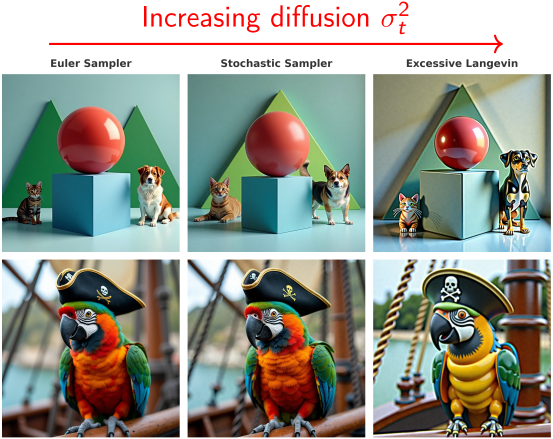
Paradox: More diffusion ($\sigma_t^2 \nearrow$) lead to more mode seeking.



As noise increases (left to right), samples cluster more tightly.

The Dark Side of Diffusion: Over-Concentration

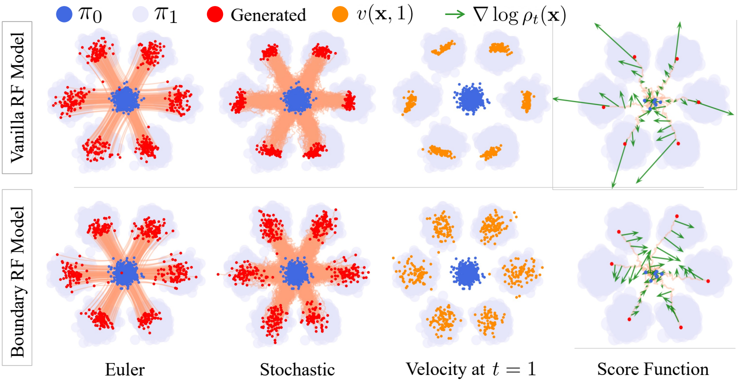
- Excessive Diffusion magnitude lead to cartoon like images. [HXL+25]
- Cartoonish images often lie in high-density regions [KAAL22, KHG25]



Figures from [HLX+25].

- Stable parameterization helps

$$v_t(x) = x + (t-1)m_t(x) \implies \nabla \log \rho_t(x) = -tm_t(x) - x.$$



Open Questions

- How to determine diffusion coefficients optimally?
- How to better limit errors in ODE/Langevin?
- Better understanding on cartoonish effect?

Rectified Flow Recap: Smooth Interpolations

- **Coupling**: Sample a noise data pair (X_0, X_1).
- **Interpolation**: Build a smooth interpolation:

$$X_t = \Phi_t(X_0, X_1, \omega), \qquad \omega : \text{any randomness}$$

with time-derivative: $\dot{X}_t = \partial_t \Phi_t(X_0, X_1, \omega)$.

- **Learning**: Fit ODE velocity field v_t with

$$L(v) = \mathbb{E}_{(X_0, X_1, t)} \left[\| \dot{X}_t - v_t(X_t, t) \|^2 \right]$$

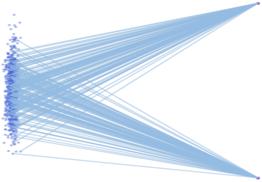
- **Inference**: solve ODE: $\dot{Z}_t = v_t(Z_t) \qquad Z_0 \sim \pi_0$.
  Optional: estimate $\nabla \log \rho_t$ and solve ODE+Langevin.
- **Further: Reflow, distillation, control, align, etc.**
- Concurrent to 1-rectified flow: stochastic interpolants [ABVE23], flow matching [LCBH+22]

Rectified Flow: Rough Interpolations

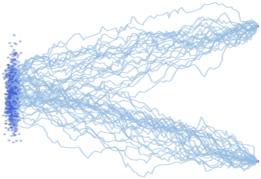
- **Coupling**: Sample a noise data pair (X_0, X_1), such that $X_0 \sim \pi_0, \quad X_1 \sim \pi_1$.
- **Interpolation**: Build a smooth interpolation:

$$X_t = \Phi_t(X_0, X_1, \omega), \qquad \omega : \text{any randomness}$$

- Besides smooth trajectories, the interpolation X_t can also have **rough paths** like Brownian motion.



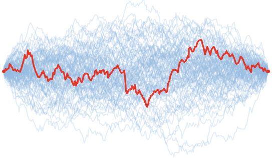
Smooth Interpolation
($\dot{X}_t = \partial_t \Phi_t(X_0, X_1, \omega)$ exists)



Rough Interpolation
(X_t is not time-differentiable)

Diffusion Bridges

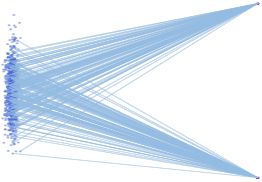
- **Brownian bridge:**



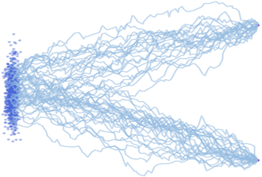
$$\mathrm{d}X_t = \frac{x_1 - X_t}{1-t}\mathrm{d}t + \sigma \mathrm{d}W_t.$$

- Grantees to arrive at $X_1 = x_1$ from any $X_0 = x_0$.
- Marginals satisfy

$$X_t \overset{\text{law}}{=} t x_1 + (1-t)X_0 + \sigma\sqrt{t(1-t)}\xi_t, \qquad \xi_t \sim \texttt{Normal}(0, I).$$



Straight Interpolation



Brownian Interpolation

Diffusion Bridges

- **Coupling**: Sample a noise data pair (X_0, X_1).
- **Interpolation**: Build interpolation by Brownian bridges:

$$\mathrm{d}X_t = \frac{X_1 - X_t}{1 - t}\mathrm{d}t + \sigma \mathrm{d}W_t.$$

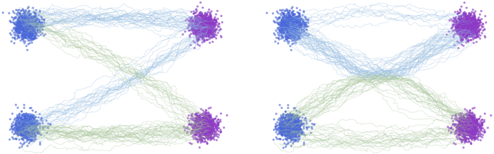
- **Learning**: learn an SDE generative model

$$\mathrm{d}Z_t = v_t(Z_t)\mathrm{d}t + \sigma \mathrm{d}W_t,$$

by fitting velocity field v_t with

$$L(v) = \mathbb{E}_{(X_0, X_1, t)}\left[\|\frac{X_1 - X_t}{1 - t} - v_t(X_t)\|^2\right].$$

Also preserve the marginal distributions.



Diffusion Bridges

- **Coupling**: Sample a noise data pair (X_0, X_1).
- **Interpolation**: Build interpolation by Brownian bridges:

$$\mathrm{d}X_t = \frac{X_1 - X_t}{1 - t}\mathrm{d}t + \sigma \mathrm{d}W_t.$$

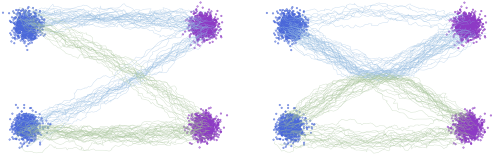
- **Learning**: learn an SDE generative model

$$\mathrm{d}Z_t = v_t(Z_t)\mathrm{d}t + \sigma \mathrm{d}W_t,$$

by fitting velocity field v_t with

$$L(v) = \mathbb{E}_{(X_0, X_1, t)}\left[\left\| \frac{X_1 - X_t}{1 - t} - v_t(X_t) \right\|^2 \right].$$

Also preserve the marginal distributions.



Diffusion Bridges

- **Interpolation**: Build an Itô interpolation (or bridge) between (X_0, X_1):

$$\mathrm{d}X_t = \mu_t(\boldsymbol{X})\mathrm{d}t + \sigma \mathrm{d}W_t, \qquad \boldsymbol{X} = \{X_t\}_{t=0}^{1}.$$

μ_t depends on the whole path $\boldsymbol{X}$ (non-causal):

$$Q^{aug}(\boldsymbol{X}) = Q^{bridge}(\boldsymbol{X} \mid X_1)\, Q^{data}(X_1).$$

- **Learning forward process**: Consider the forward process:

$$\mathrm{d}Z_t = v_t(Z_t)\mathrm{d}t + \sigma_t \mathrm{d}W_t, \quad \text{with} \quad v_t(x) = \mathbb{E}\left[\mu_t(\boldsymbol{X}) \mid X_t = x\right],$$

Solved by fitting:

$$\min_v \frac{1}{2} \int \mathbb{E}\left[\|\mu_t(\boldsymbol{X}) - v_t(X_t)\|^2\right] \mathrm{d}t$$

Diffusion Bridges

- **Interpolation**: Build an Itô interpolation (or bridge) between (X_0, X_1):

$$\mathrm{d}X_t = \mu_t(\boldsymbol{X})\mathrm{d}t + \sigma \mathrm{d}W_t, \qquad \boldsymbol{X} = \{X_t\}_{t=0}^{1}.$$

  μ_t depends on the whole path $\boldsymbol{X}$ (non-causal):

$$\mathrm{Q}^{aug}(\boldsymbol{X}) = Q^{bridge}(\boldsymbol{X} \mid X_1)\, Q^{data}(X_1).$$

- **Learning forward process**: Consider the forward process:

$$\mathrm{d}Z_t = v_t(Z_t)\mathrm{d}t + \sigma_t \mathrm{d}W_t, \quad \text{with} \quad v_t(x) = \mathbb{E}\left[\mu_t(\boldsymbol{X}) \mid X_t = x\right],$$

  Solved by fitting:

$$\min_v \underbrace{\frac{1}{2}\int \mathbb{E}\left[\left\|\mu_t(\boldsymbol{X} \mid X_1) - v_t(X_t)\right\|^2\right]\mathrm{d}t}_{\mathrm{KL}(\mathrm{Q}^{aug} \,\|\, \mathrm{P})}$$

Markovization: $\mathbb{P}_Z$ yields the Markovization of $\mathbb{P}_X$:

$$\min_{\mathrm{P}} \mathrm{KL}(\mathrm{Q}^{aug} \,\|\, \mathrm{P}) \quad \text{s.t.} \quad \mathrm{P} \text{ is Markovian.}$$

Diffusion Bridges

- Related to a set of deep mathematics:
  - Gyongy projection (1986), Reverse-time diffusion (Anderson, 1982), Doob's h transform, Follmer process.
- Time-reversal diffusion (Song et al. 20), diffusion bridge mixture (Peluchetti 21), Diffusion bridges via *h*-transform (Liu et al. 22), Schrodinger bridge matching (De Bortoli et.al, 2021, Shi et.al. 23), stochastic interpolant (Albergo et.al. 23), etc.
- Further extension:
  - Diffusion with **random stopping time** (Ye et al. 22).
  - **Prior informed** design of diffusion bridges: (Liu et al. 23).

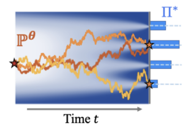
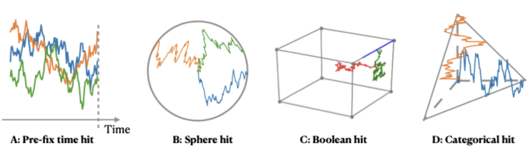


Figure 1: An Ω-Bridge on discrete domain $\Omega = \{1, 2, 3, 4\}$.



Jump Bridges for Discrete Data

- **Coupling**: Sample a noise data pair (X_0, X_1).
- **Interpolation**: Build interpolation by a non-Markov jump process:

$$X_t = U_t \circ X_1 + (1 - U_t) \circ X_0 \ \sim \ \mathtt{Jump}(R_t(\boldsymbol{X})),$$

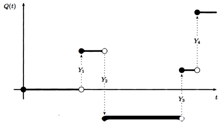
U_t: a binary jump process from $U_0 = 0$ to $U_1 = 1$.

- **Learning**: learn a Markov jump process

$$Z_t = \mathtt{Jump}(R_t(Z_t)), \qquad R_t\text{: Jump rate function,}$$

by fitting with

$$\mathbb{E}_{(X_0, X_1, t)} \left[\mathrm{KL}(R_t(\boldsymbol{X}) \parallel R_t(X_t)) \right].$$



- D3PM [AJH+21], CTMC [CBDB+22], RADD [ONX+24], MDLM [SAS+24], LLaDA [NZY+25], discrete flow matching [GRS+24].