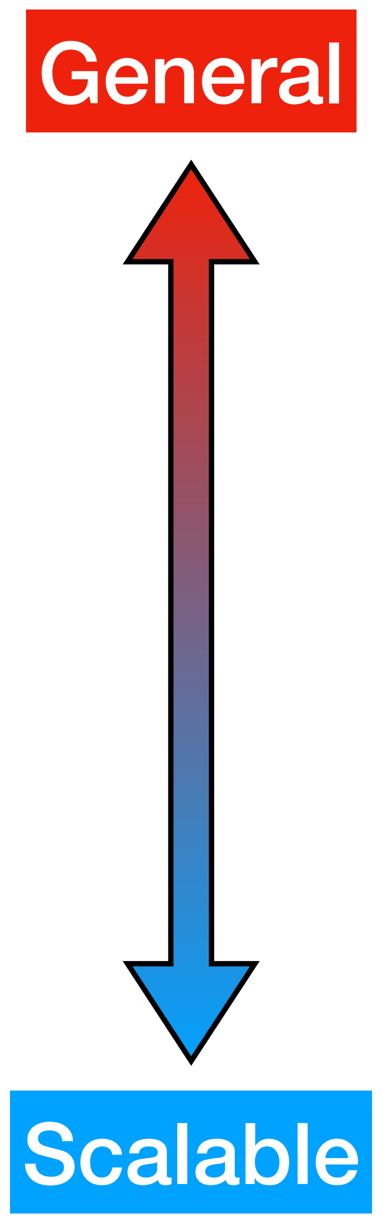
Flow Matching

Simplifying and Generalizing Diffusion Models

Yaron Lipman

Meta

מכון ויצמן למדע
WEIZMANN INSTITUTE OF SCIENCE



Generative models

$\cup$

Flows

$\cup$

Flow Matching

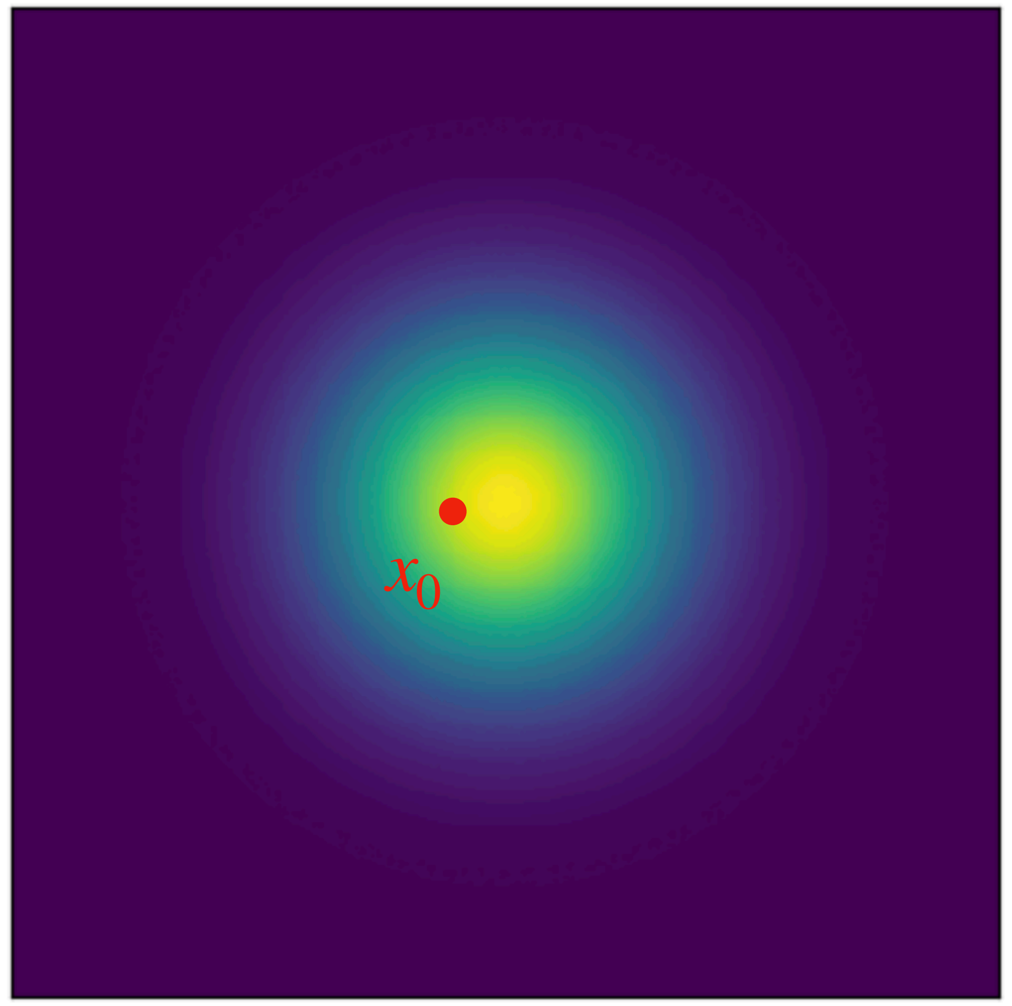
$\cup$

Diffusion Models

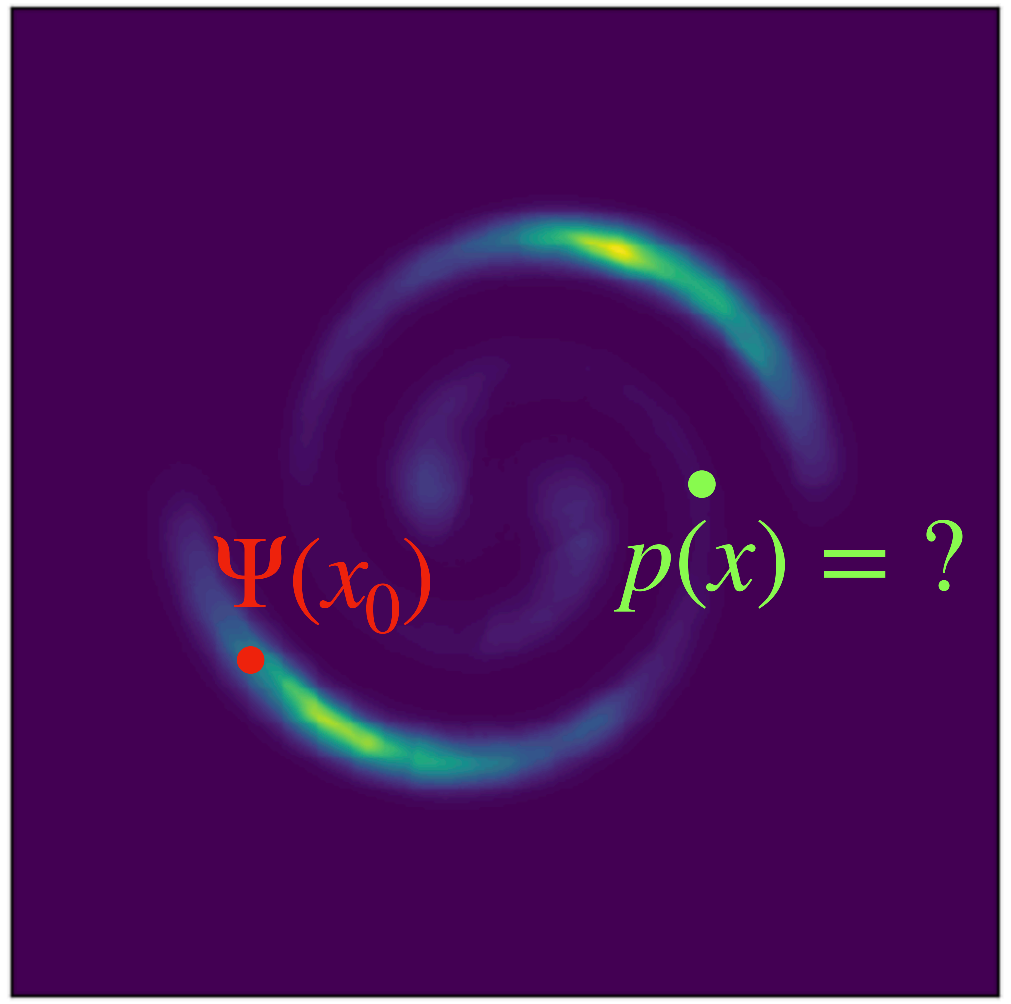
Generative models

$p(x)$ $\quad$ $q(x)$

$\mathbb{R}^d$

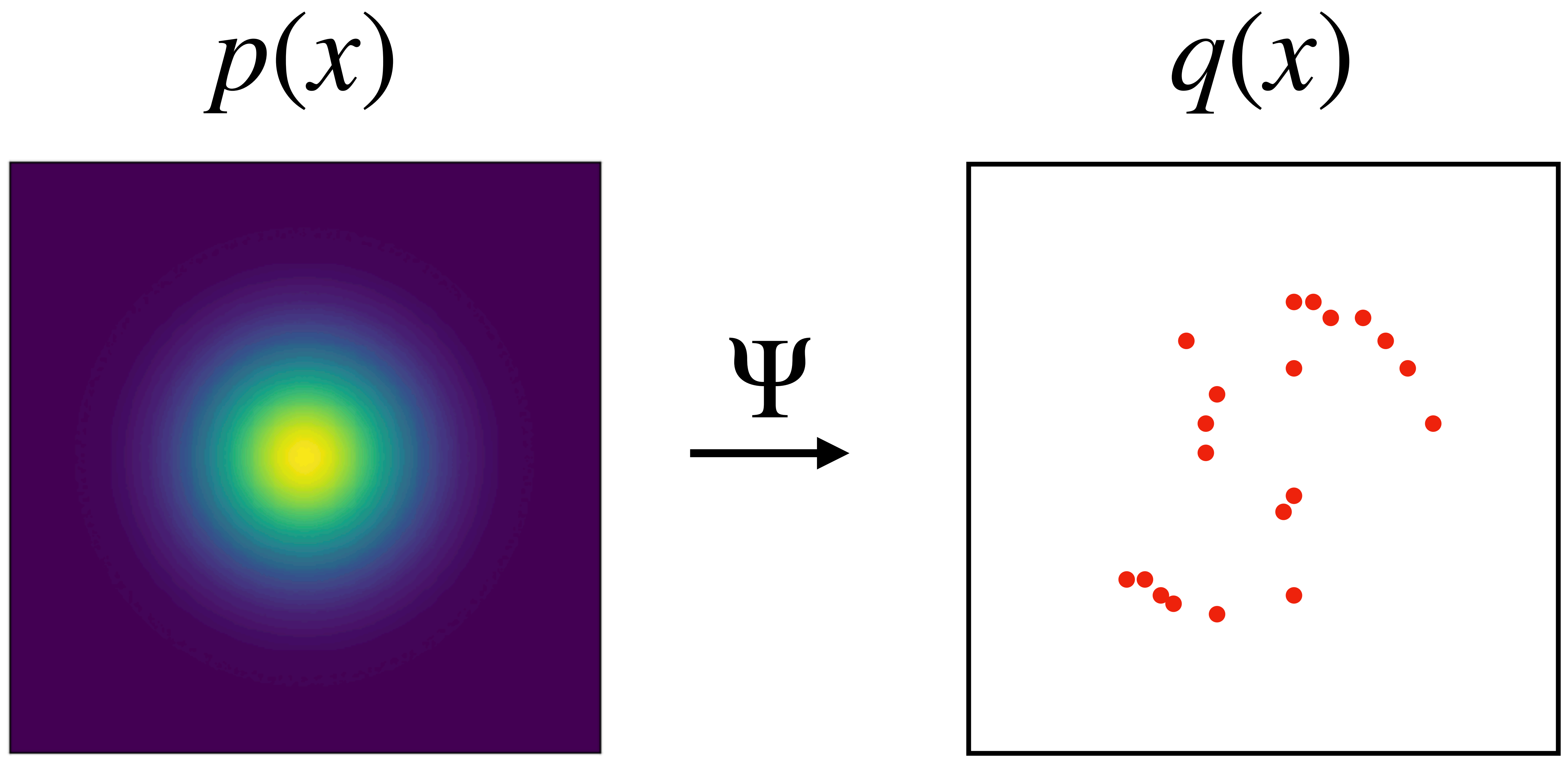


$\xrightarrow{\Psi}$



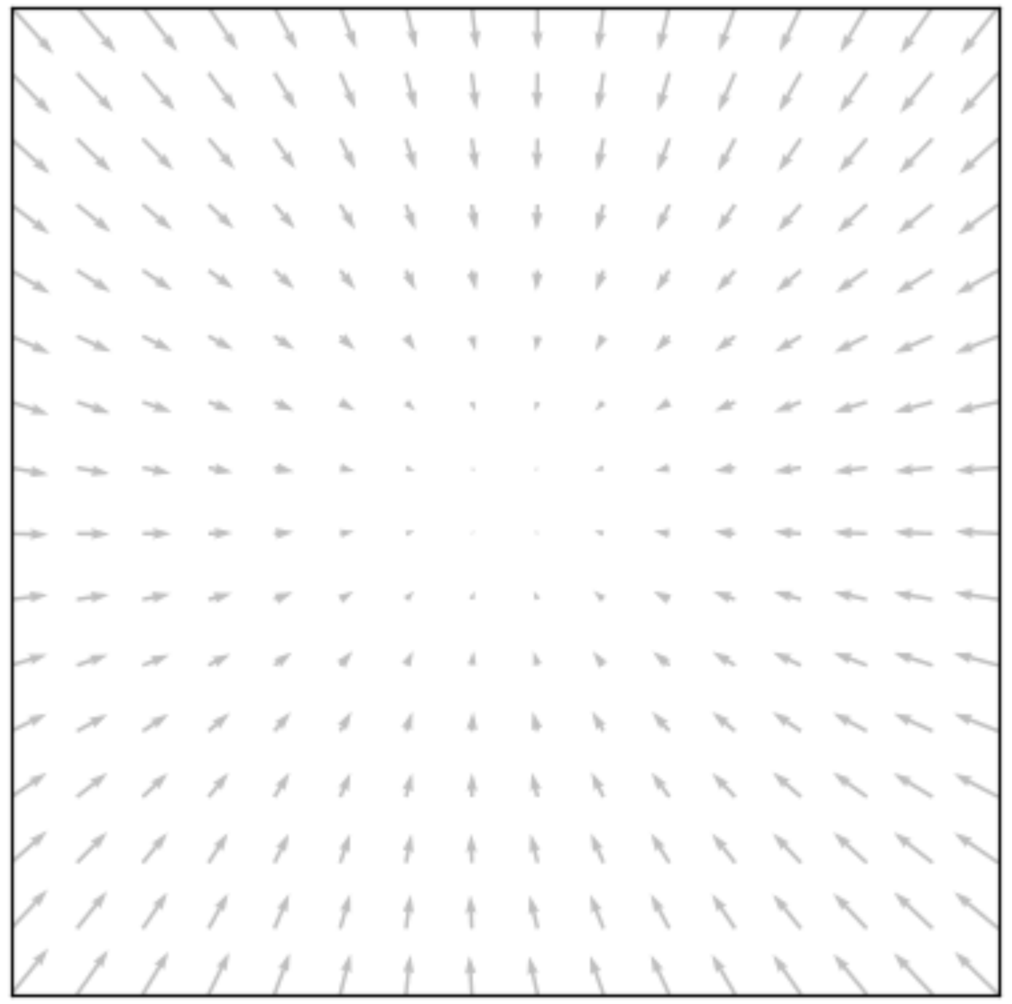
$x_0 \sim p$ $\quad$ $\Psi(x_0) \sim q$

Generative models



Flows as Generative Models

$v_t(x)$

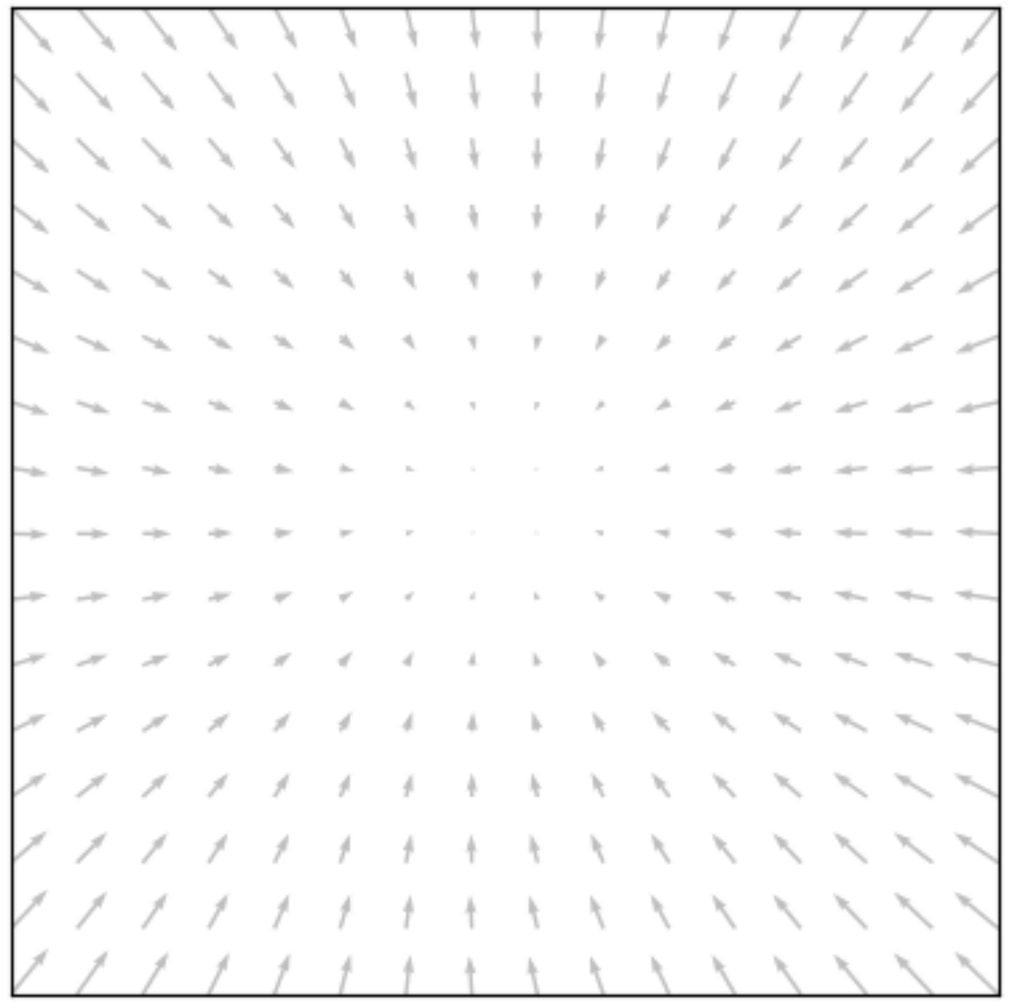


$v : [0,1] \times \mathbb{R}^d \to \mathbb{R}^d$

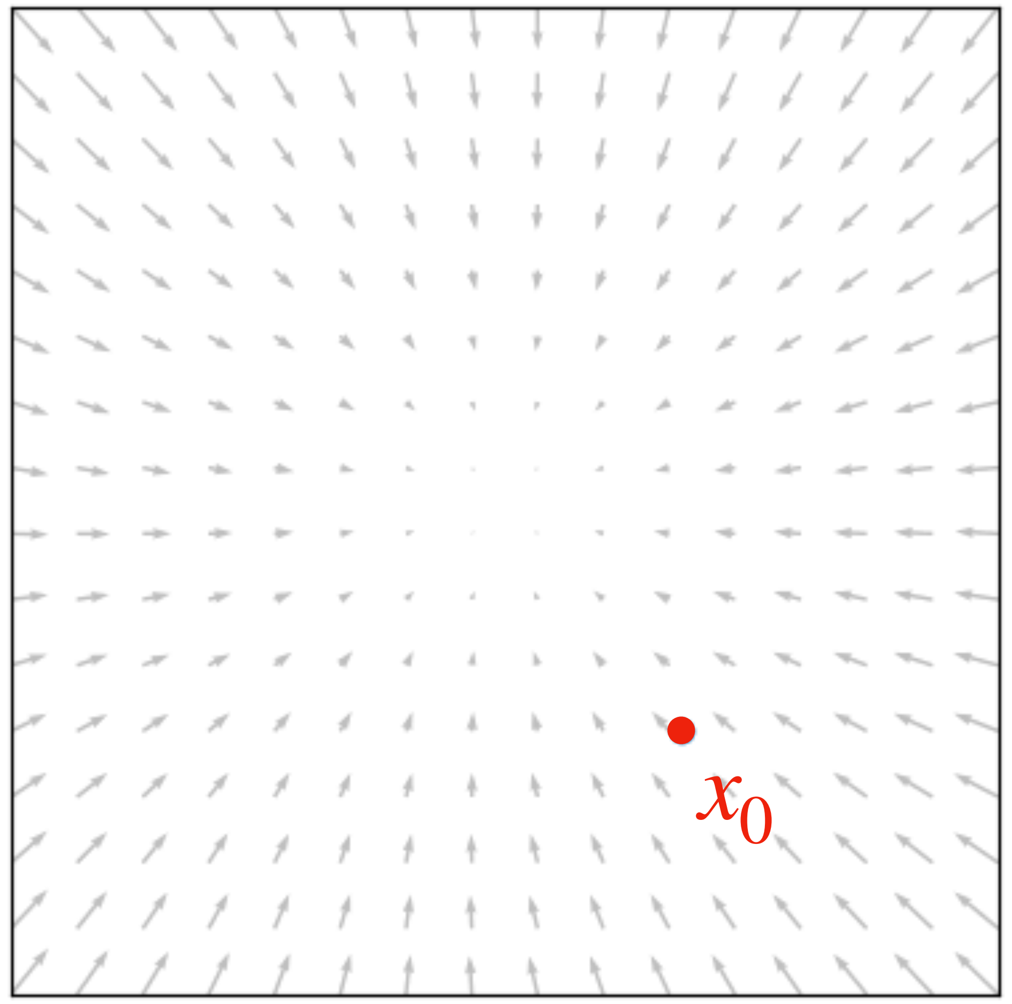
[Chen et al. 2018]

Flows as Generative Models

$$v_t(x)$$



$$x_t = \Psi_t(x_0)$$



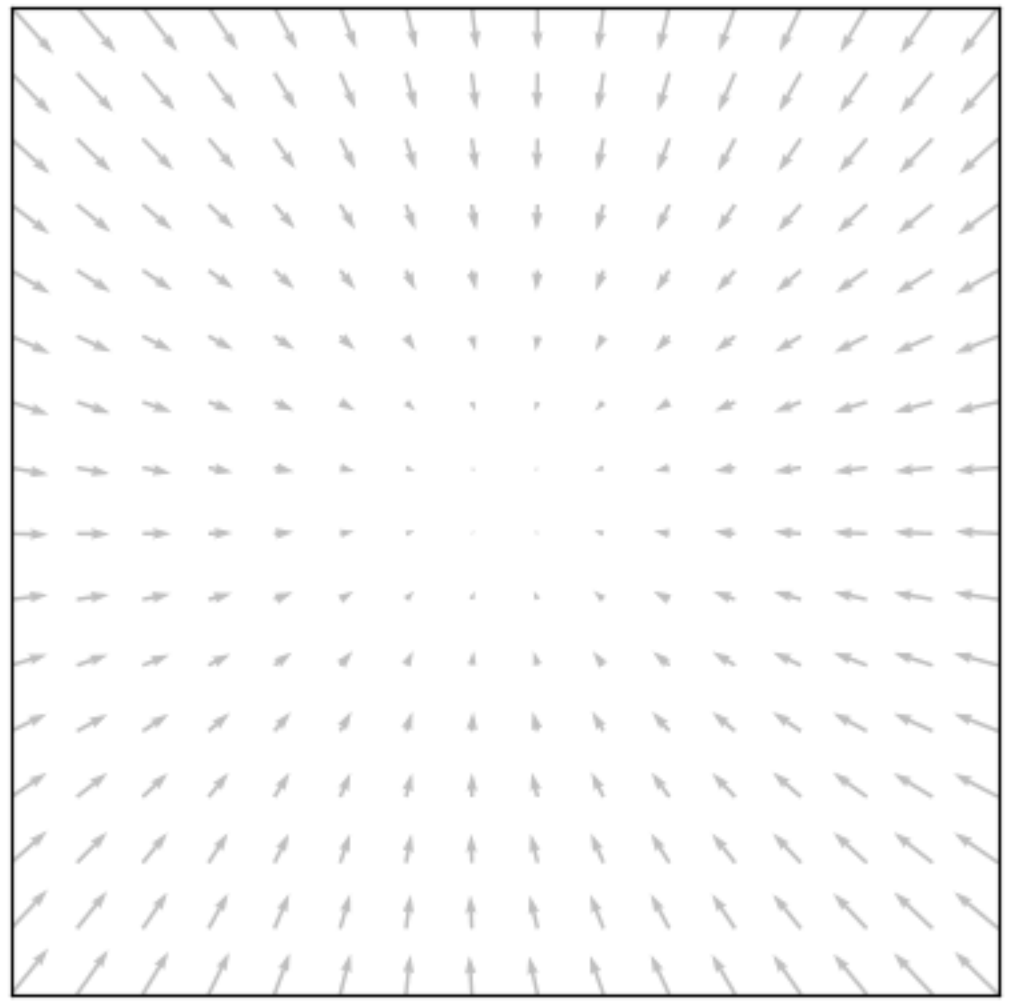
Flow ODE

$$\dot{x}_t = v_t(x_t)$$

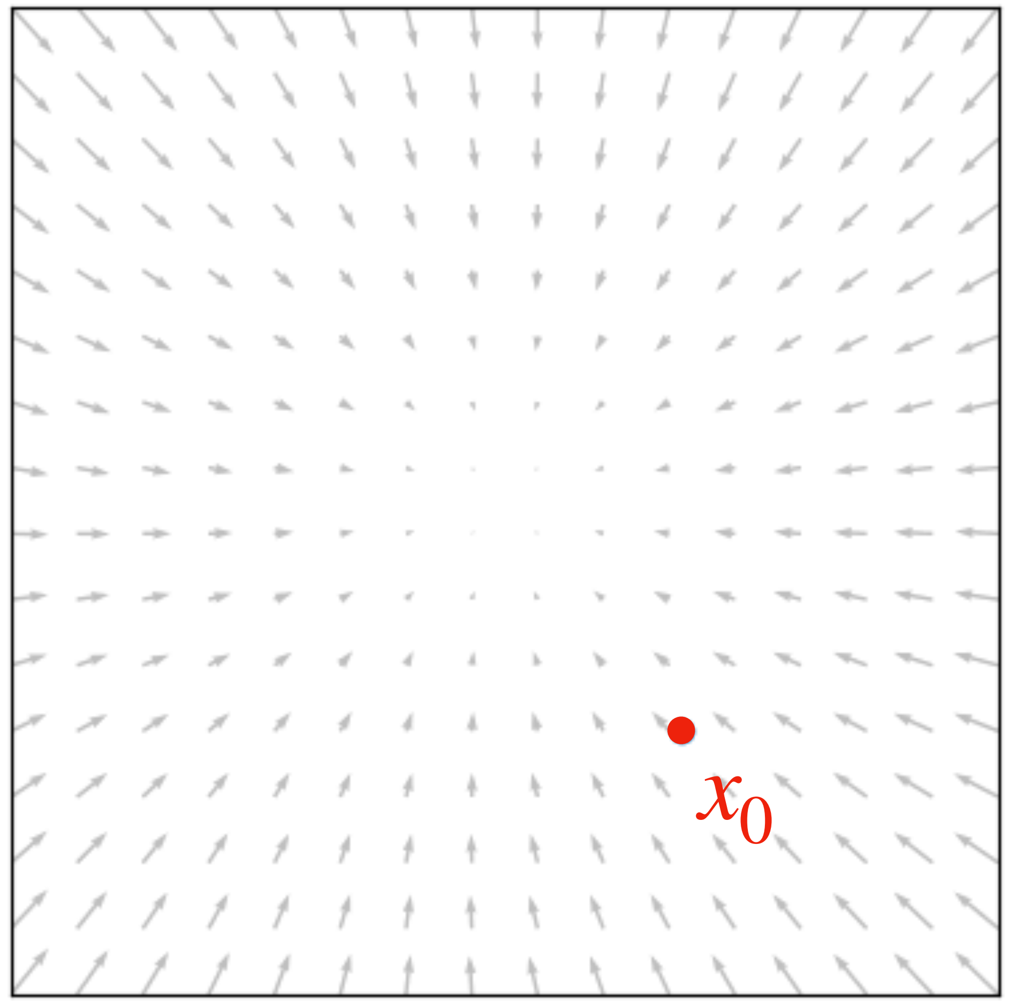
[Chen et al. 2018]

Flows as Generative Models

$$v_t(x)$$



$$x_t = \Psi_t(x_0)$$

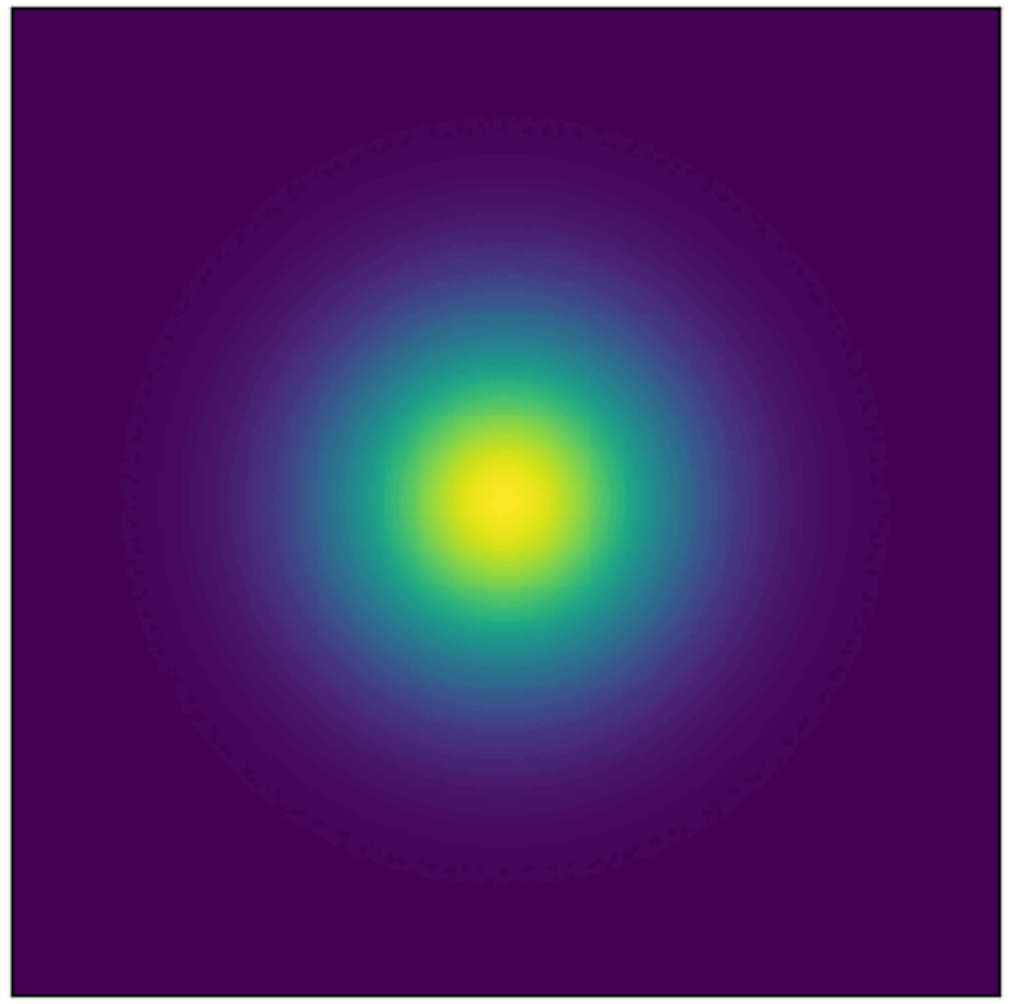


Flow ODE

$$\dot{x}_t = v_t(x_t)$$

$$x_0 \sim p$$

$$x_t \sim p_t$$

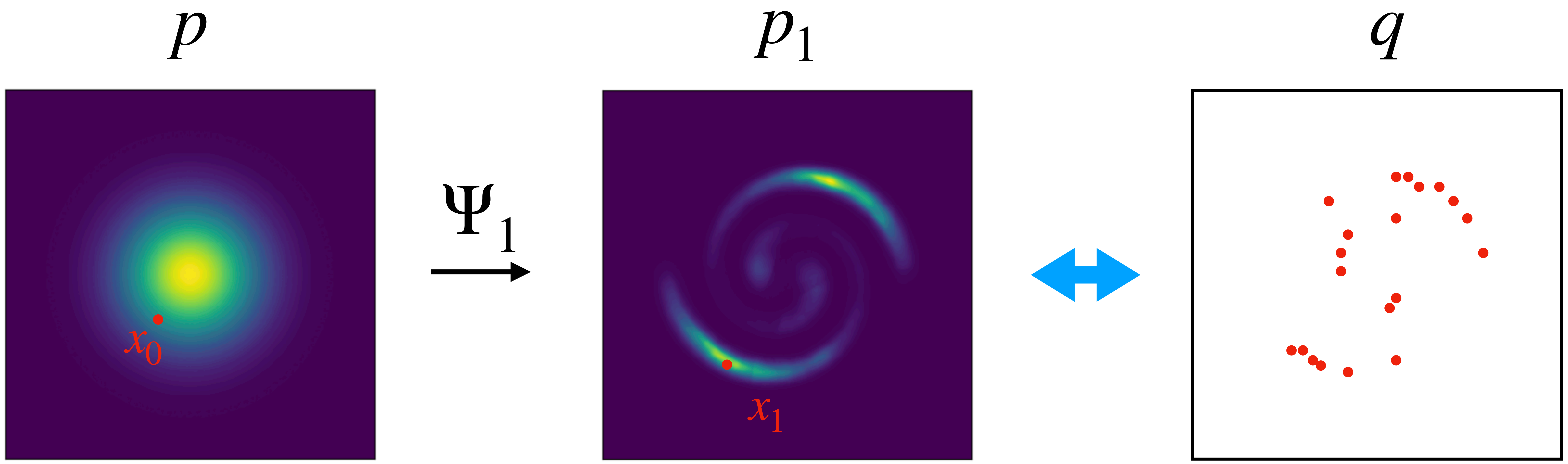


Continuity Equation PDE

$$\dot{p}_t = -\operatorname{div}(p_t v_t)$$

[Chen et al. 2018]

Flows as Generative Models



[Chen et al. 2018]

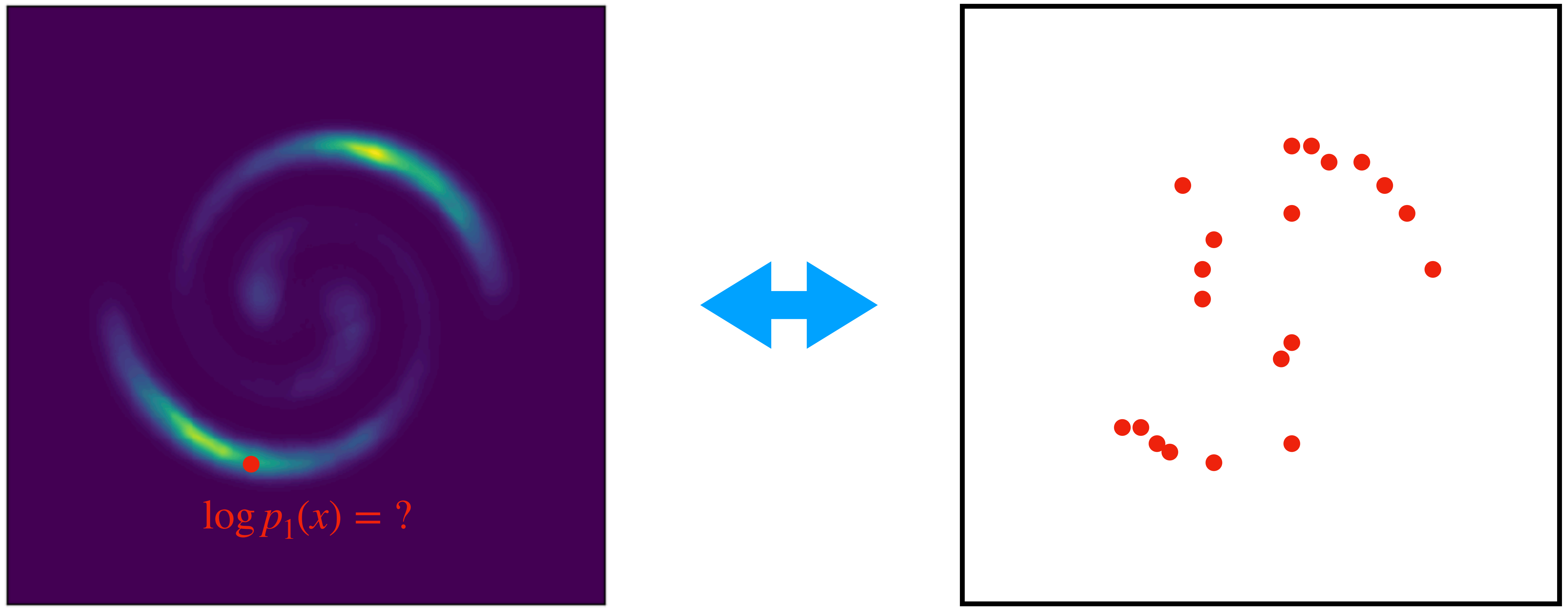
Training Flows

Minimize:

$$D_{\mathrm{KL}}(q \,\|\, p_1) = -\,\mathbb{E}_{x \sim q} \log p_1(x) + c$$

p_1

q

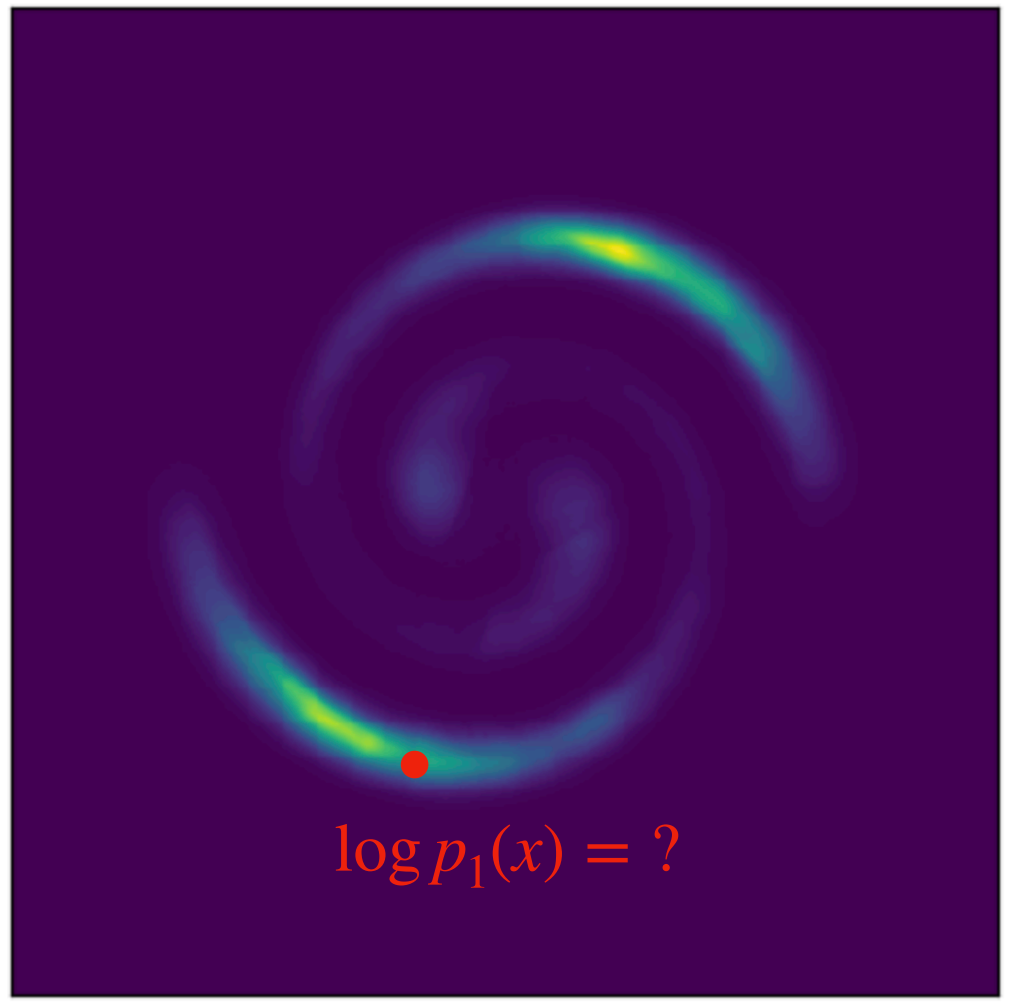


[Chen et al. 2018]

Training Flows

Minimize: $$D_{\mathrm{KL}}(q \,\|\, p_1) = -\mathbb{E}_{x\sim q} \log p_1(x) + c$$

p_1



Continuity Equation (*fixed x*)

$$\dot{p}_t = -\operatorname{div}(p_t v_t)$$

Instantaneous change of coordinates (*trajectory x_t*)

$$\frac{d}{dt}\log p_t(x_t) = -\operatorname{div}(v_t(x_t))$$

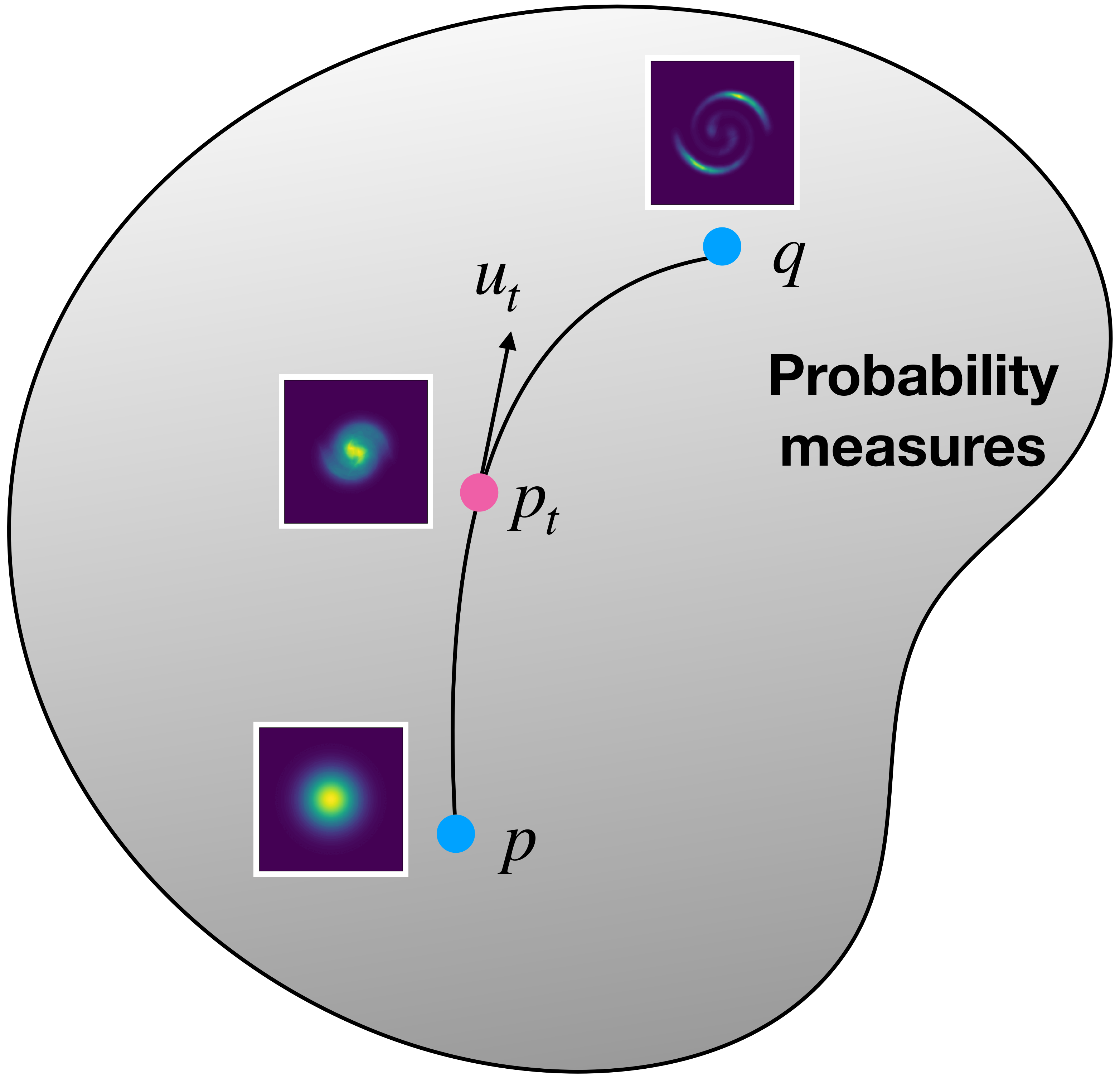
Requires simulation x_t

[Chen et al. 2018]

Flow Matching

Continuity Equation

$$\dot{p}_t = -\operatorname{div}(p_t u_t)$$



Flow Matching

Continuity Equation

$$\dot{p}_t = -\operatorname{div}(p_t u_t)$$

Compare **velocities** instead of probabilities:

Neural network

$$L_{\mathrm{FM}}(q \| p_1) = \min \mathbb{E}_{t,\, p_t(x)} \| v_t(x) - u_t(x) \|^2$$

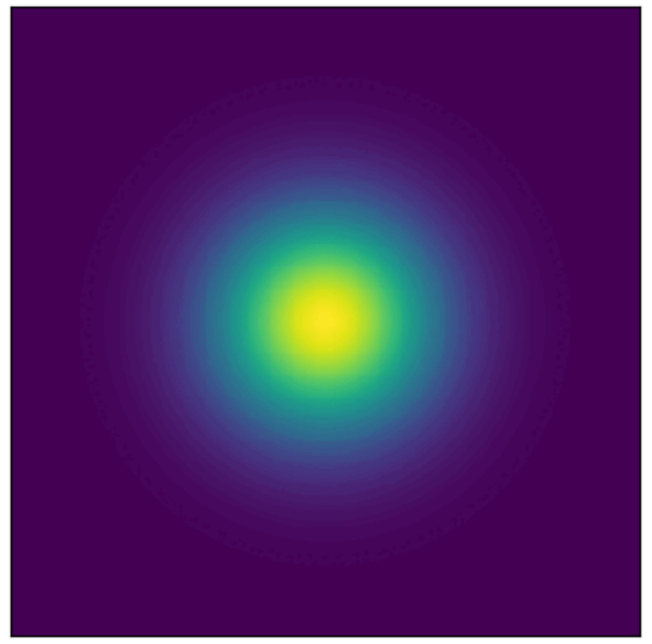
★ Marginal supervision p_t, u_t
★ Conditional FM loss

Flow Matching

Supervision p_t, u_t

Marginal path

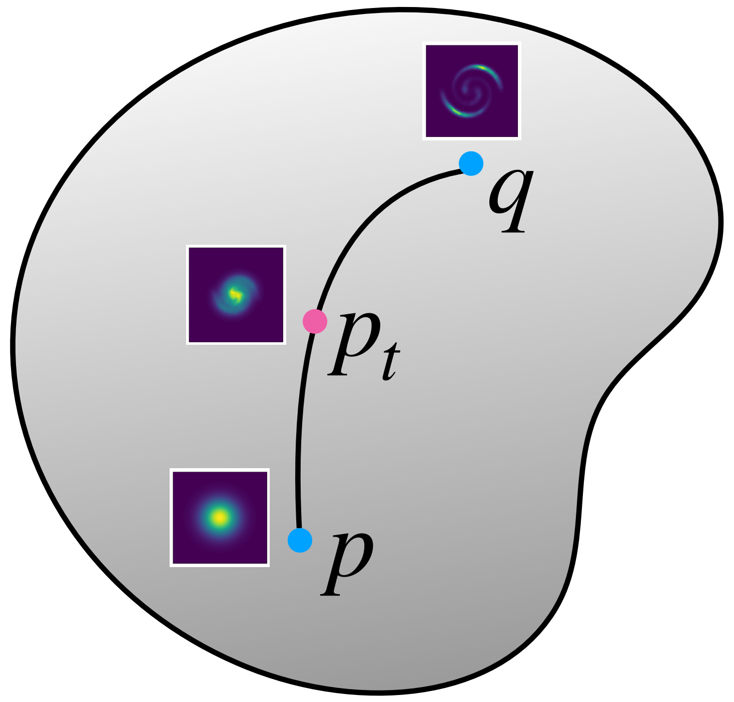
$$p_t(x) = \int p_t(x \mid x_1) q(x_1) dx_1$$



Boundary conditions:

$$p_0 = p$$

$$p_1 = q$$

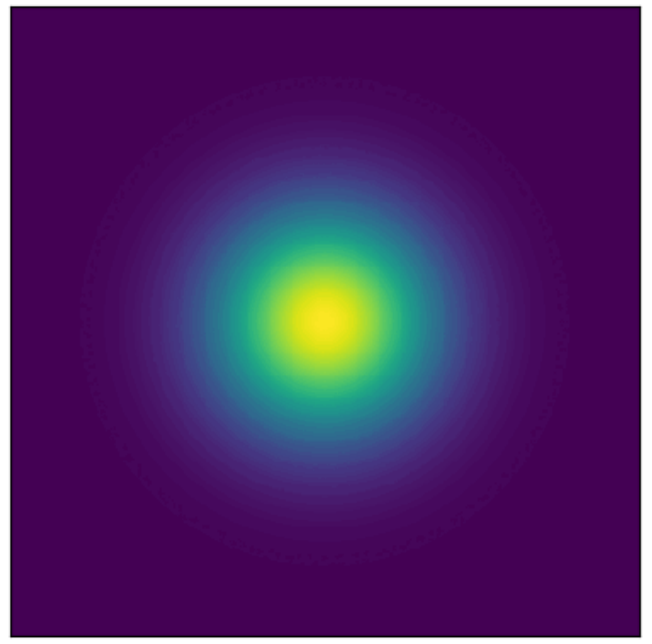


Flow Matching

Supervision p_t, u_t

Marginal path

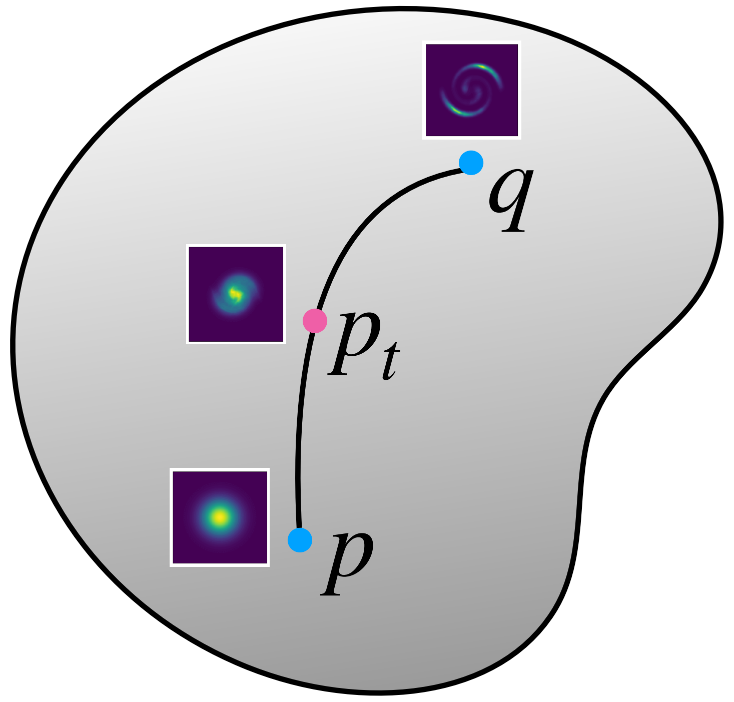
$$p_t(x) = \int p_t(x|x_1)q(x_1)dx_1$$



Boundary conditions:

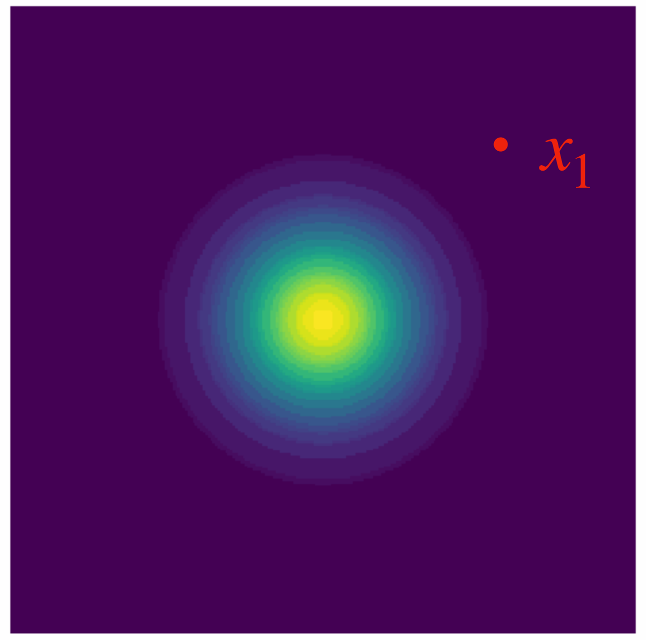
$$p_0 = p$$

$$p_1 = q$$



Conditional path

$$p_t(x|x_1)$$



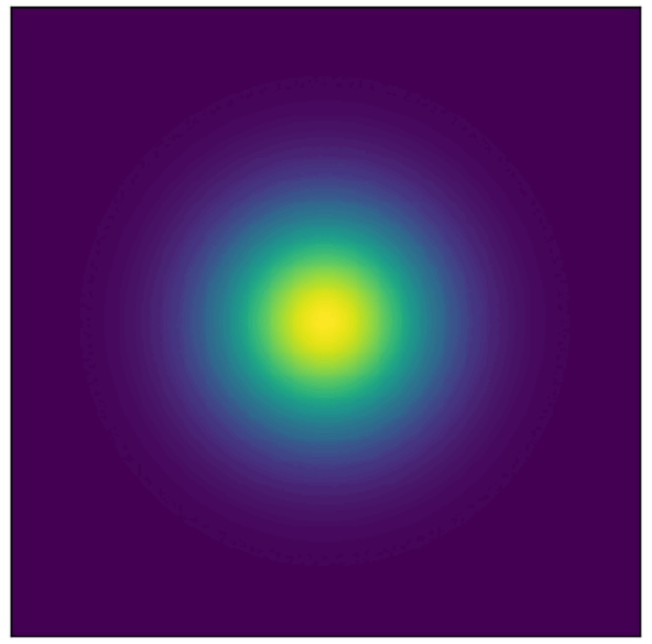
$$p_0(\cdot|x_1) = p$$

$$p_1(\cdot|x_1) = \delta_{x_1}$$

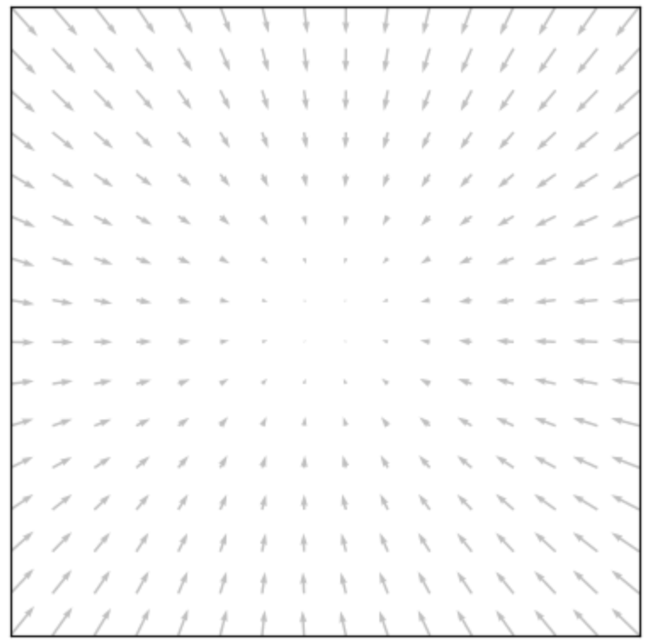
Flow Matching

Supervision p_t, u_t

$$p_t(x) = \int p_t(x|x_1)q(x_1)dx_1$$

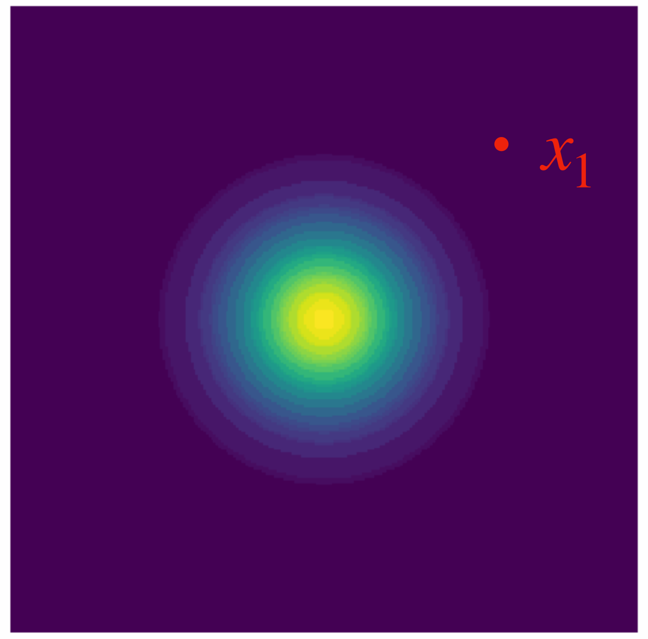


$$u_t(x) = \int u_t(x|x_1)\frac{p_t(x|x_1)q(x_1)}{p_t(x)}dx_1$$

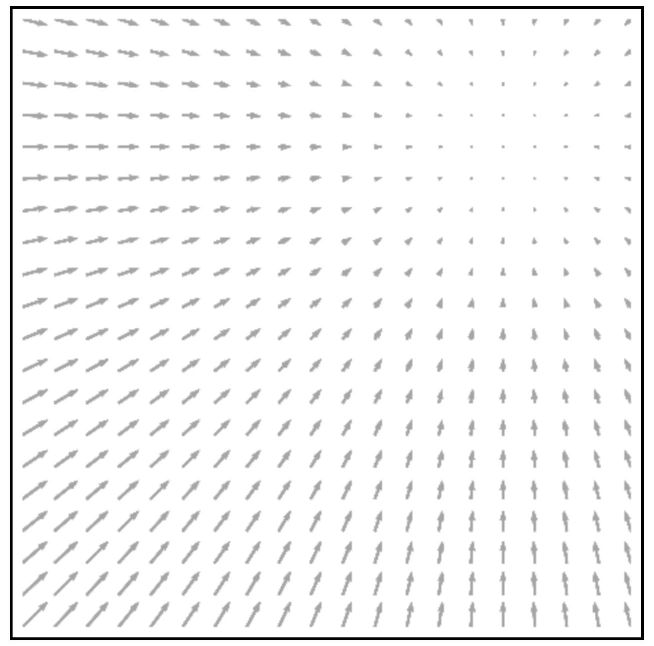


Marginal path

$p_t(x|x_1)$



$u_t(x|x_1)$



Conditional path

$$x_t = \Psi_t(x_0|x_1)$$

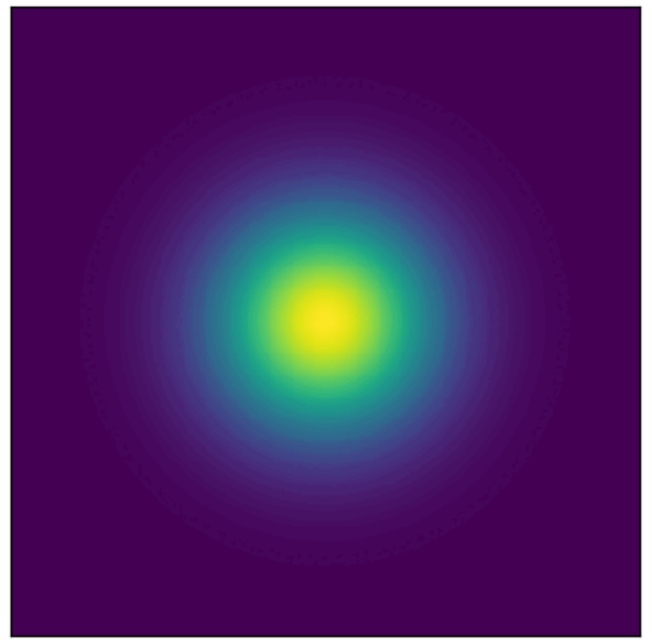
Flow Matching

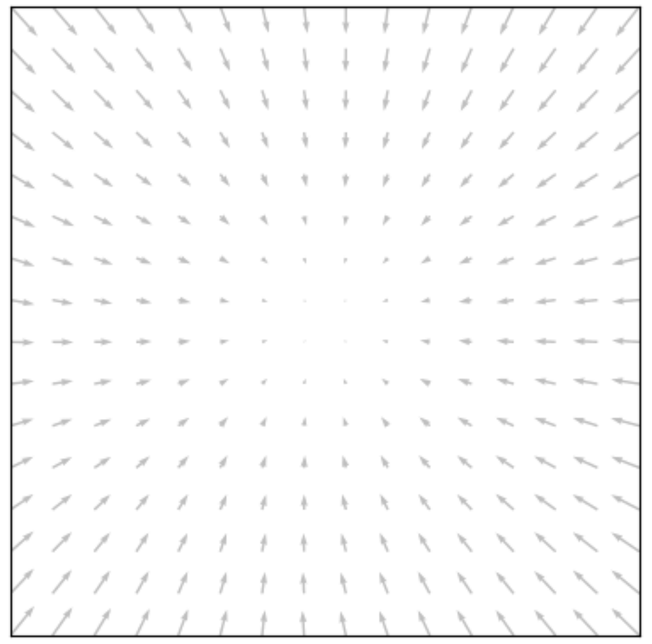
Supervision p_t, u_t

$$p_t(x) = \int p_t(x|x_1)q(x_1)dx_1$$

$$u_t(x) = \int u_t(x|x_1)\frac{p_t(x|x_1)q(x_1)}{p_t(x)}dx_1$$

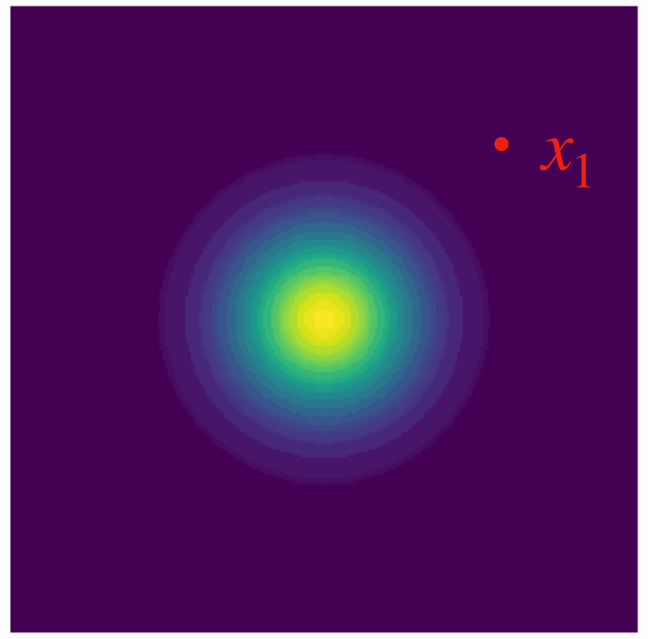
Marginal path



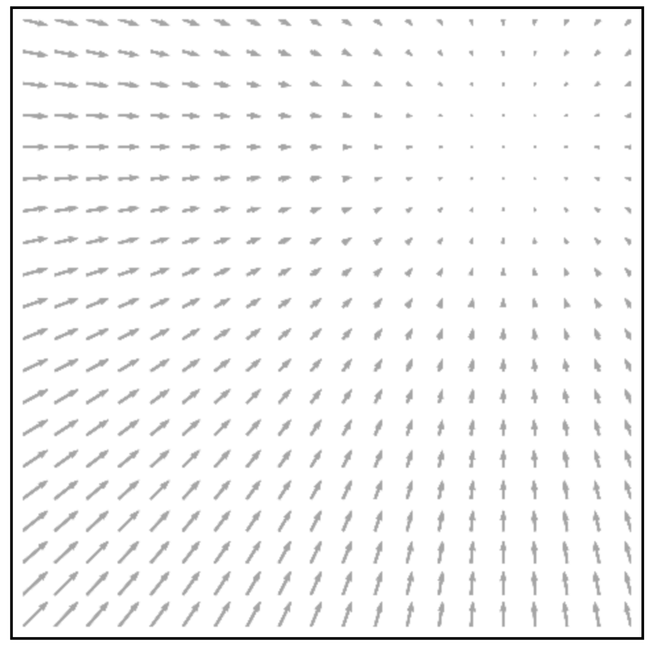


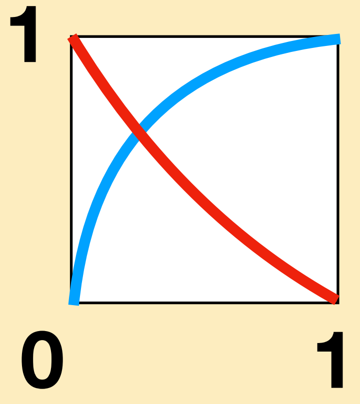
Conditional path

$$p_t(x|x_1)$$



$$u_t(x|x_1)$$





$$x_t = \Psi_t(x_0|x_1) = \sigma_t x_0 + \alpha_t x_1$$

$$x_0 \sim p \Rightarrow x_t \propto p\left(\frac{x - \alpha_t x_1}{\sigma_t}\right)$$

Flow Matching

Tractable loss

$$L_{\text{FM}}(q||p_1) = \min \mathbb{E}_{t, p_t(x)} ||v_t(x) - u_t(x)||^2$$

The gradients of losses coincide:

$\nabla_\theta L_{\text{FM}} = \nabla_\theta L_{\text{CFM}}$

$$L_{\text{CFM}}(q||p_1) = \min \mathbb{E}_{t, q(x_1), p_t(x|x_1)} ||v_t(x) - u_t(x\,|\,x_1)||^2$$

$$= \min \mathbb{E}_{t, q(x_1), p(x_0)} ||v_t(x_t) - \dot{x}_t||^2$$

$x_t \sim p_t(x\,|\,x_1)$

$\dot{x}_t = u_t(x_t\,|\,x_1)$

[L et al. 2022]

Flow Matching

Recap

Algorithm 1: Flow Matching training.

Input : dataset q, noise p
Initialize v^θ
while *not converged* **do**
  $t \sim \mathcal{U}([0,1])$ ▷ sample time
  $x_1 \sim q(x_1)$ ▷ sample data
  $x_0 \sim \boxed{p(x_0)}$ ▷ sample noise
  $x_t = \boxed{\Psi_t(x_0|x_1)}$ ▷ conditional flow
  Gradient step with $\nabla_\theta \| v_t^\theta(x_t) - \dot{x}_t \|^2$
Output: v^θ

Algorithm 2: Flow Matching sampling.

Input : trained model v^θ
$x_0 \sim p(x_0)$ ▷ sample "noise"
Numerically solve ODE $\dot{x}_t = v_t^\theta(x_t)$
Output: x_1

[L. et al. 2022]

[Liu et al. 2022]

[Albergo & Vanden-Eijnden 2022]

Flow Matching vs. Diffusion

Algorithm 1: Flow Matching training.

Input : dataset q, noise p
Initialize v^θ
while *not converged* **do**
  $t \sim \mathcal{U}([0,1])$ ▷ sample time
  $x_1 \sim q(x_1)$ ▷ sample data
  $x_0 \sim p(x_0)$ ▷ sample noise
  $x_t = \Psi_t(x_0|x_1)$ ▷ conditional flow
  Gradient step with $\nabla_\theta \| v_t^\theta(x_t) - \dot{x}_t \|^2$
Output: v^θ

Algorithm 2: Diffusion training.

Input : dataset q, noise p
Initialize s^θ
while *not converged* **do**
  $t \sim \mathcal{U}([0,1])$ ▷ sample time
  $x_1 \sim q(x_1)$ ▷ sample data
  $x_t = p_t(x_t|x_1)$ ▷ sample conditional prob
  Gradient step with
  $\nabla_\theta \| s_t^\theta(x_t) - \nabla_{x_t} \log p_t(x_t|x_1) \|^2$
Output: v^θ

$p_t(x_t | x_1)$ general
$p(x_0)$ is general

$p_t(x_t | x_1)$ closed-form from of SDE $dx_t = f_t dt + g_t dw$

- *Variance Exploding*: $p_t(x | x_1) = \mathcal{N}(x | x_1, \sigma_{1-t}^2 I)$
- *Variance Preserving*: $p_t(x | x_1) = \mathcal{N}(x | \alpha_{1-t} x_1, (1 - \alpha_{1-t}^2) I)$
  $\alpha_t = e^{-\frac{1}{2}T(t)}$

$p(x_0)$ is Gaussian
$p_0(\cdot | x_1) \approx p$

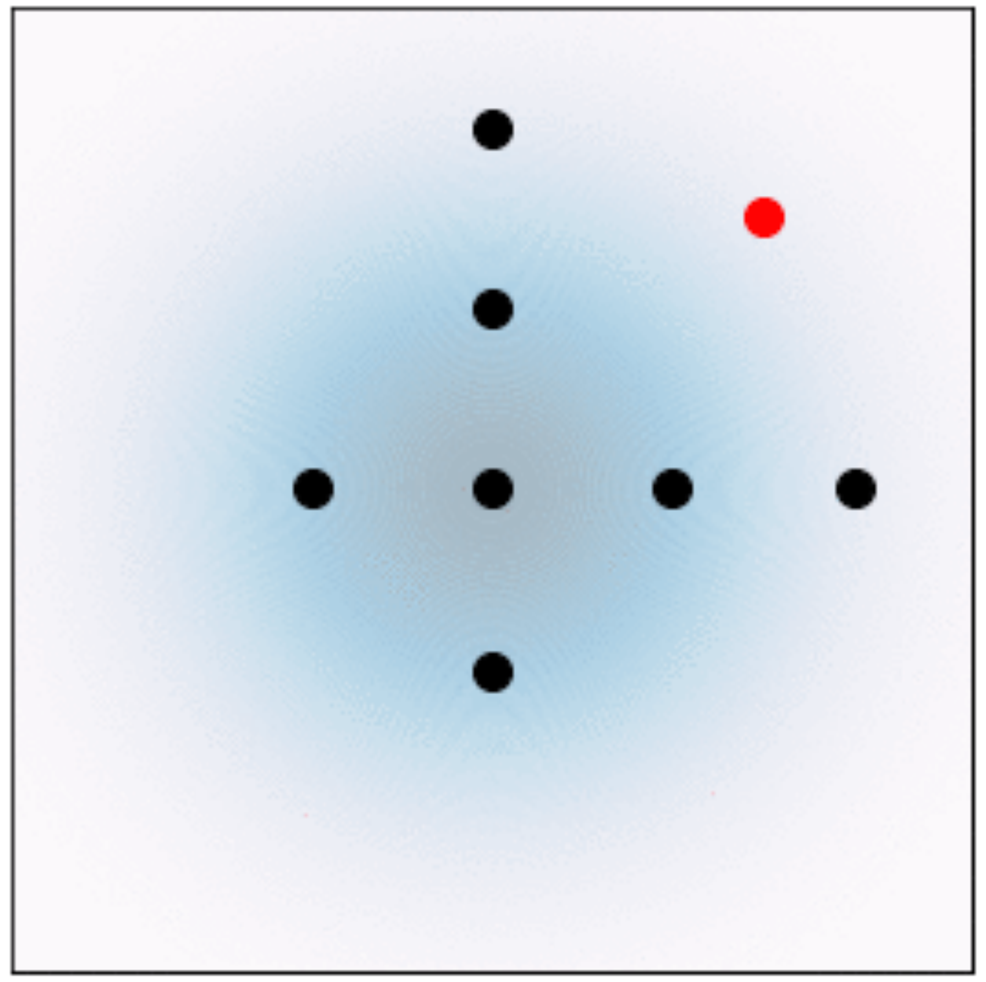
Conditional Optimal Transport Paths

(Cond-OT)

- Example of non-diffusion choice

$$p_t(x|x_1), u_t(x|x_1) \quad \Leftrightarrow \quad \Psi_t(x_0|x_1)$$

$$\mathcal{N}(tx_1, (1-t)^2 I), \frac{x_1 - x}{1-t} \qquad \Psi_t(x_0|x_1) = (1-t)x_0 + tx_1$$



Cond-OT

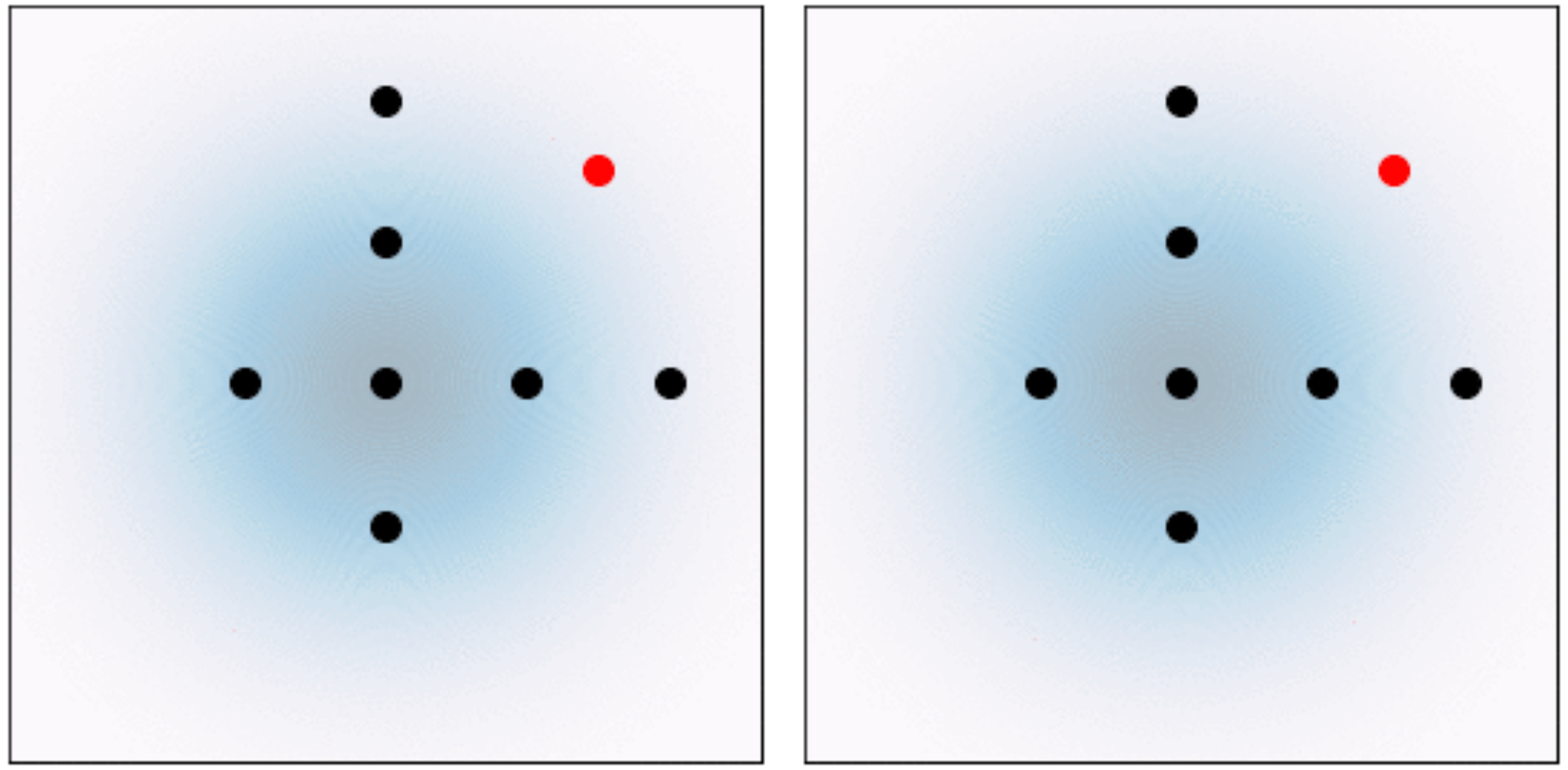
Conditional Optimal Transport Paths

(Cond-OT)

- Example of non-diffusion choice

$$p_t(x \mid x_1),\ u_t(x \mid x_1) \quad \Leftrightarrow \quad \Psi_t(x_0 \mid x_1)$$

$$\mathcal{N}(tx_1, (1-t)^2 I),\ \frac{x_1 - x}{1-t} \qquad \Psi_t(x_0 \mid x_1) = (1-t)x_0 + tx_1$$



Diffusion (VP) **Cond-OT**

Conditional Optimal Transport Paths

(Cond-OT)

- Example of non-diffusion choice

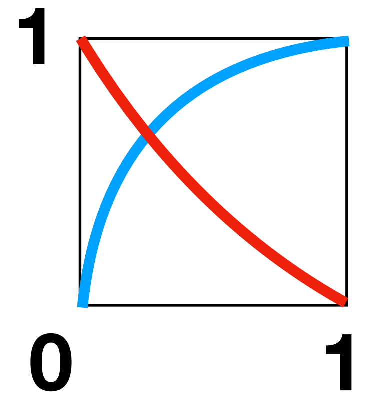
$$p_t(x \mid x_1), u_t(x \mid x_1) \quad \Leftrightarrow \quad \Psi_t(x_0 \mid x_1)$$

$$\mathcal{N}(tx_1, (1-t)^2 I), \frac{x_1 - x}{1-t} \qquad \Psi_t(x_0 \mid x_1) = (1-t)x_0 + tx_1$$

- *Note*: the marginals $p_t(x), u_t(x)$ are **NOT** optimal transport

Conditional Optimal Transport Paths

- *Theorem (informal)*: Among all schedulers Cond-OT is **Kinetic Optimal** for **sparse data in high dimension** $(n/d^{1/2} \to 0)$.



$$x_t = \sigma_t x_0 + \alpha_t x_1$$

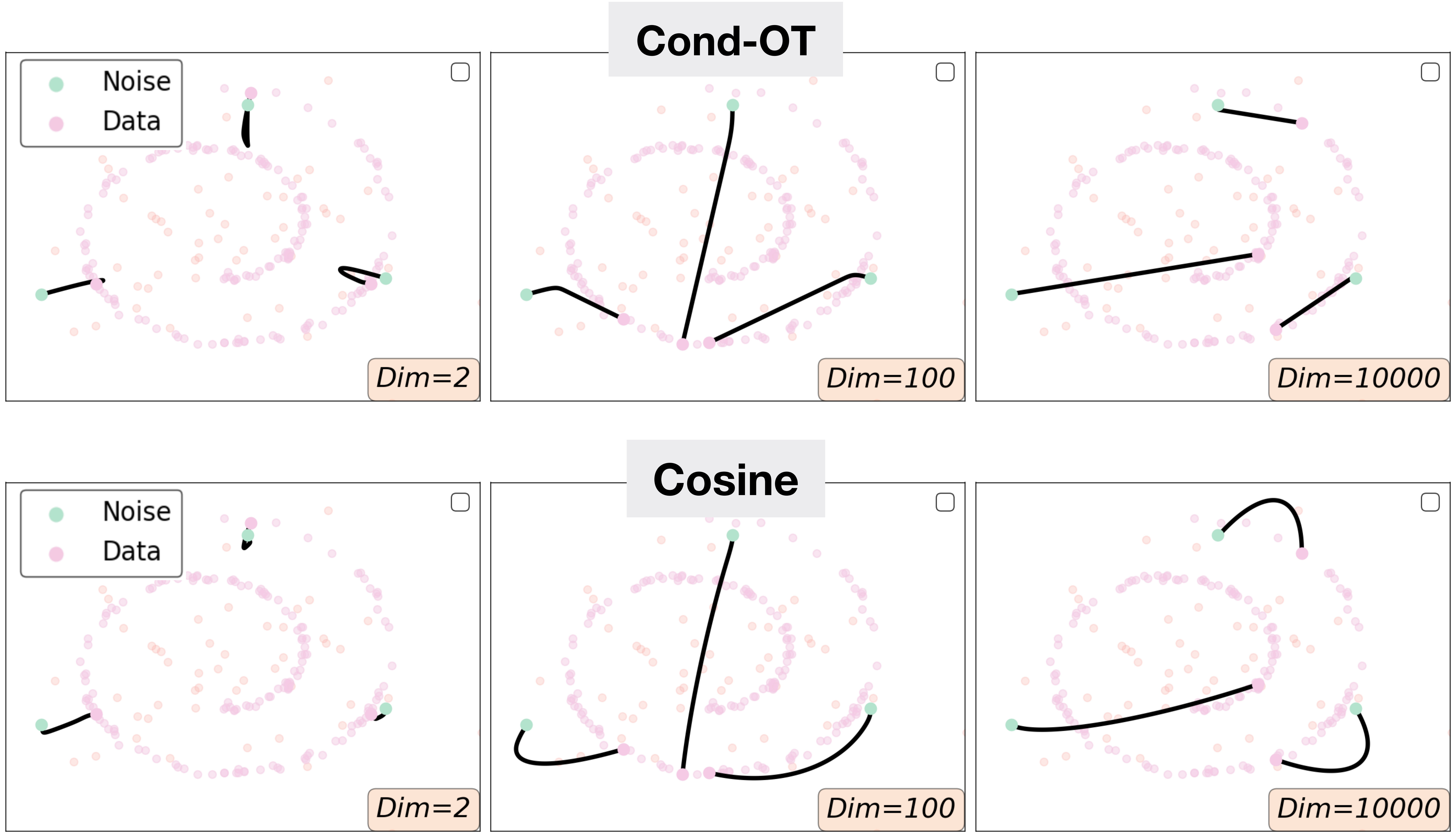
Kinetic Energy

$$E_K = \mathbb{E}_{t,p_t(x)} \|u_t(x)\|^2$$

[Shaul et al. 2023]

Conditional Optimal Transport Paths

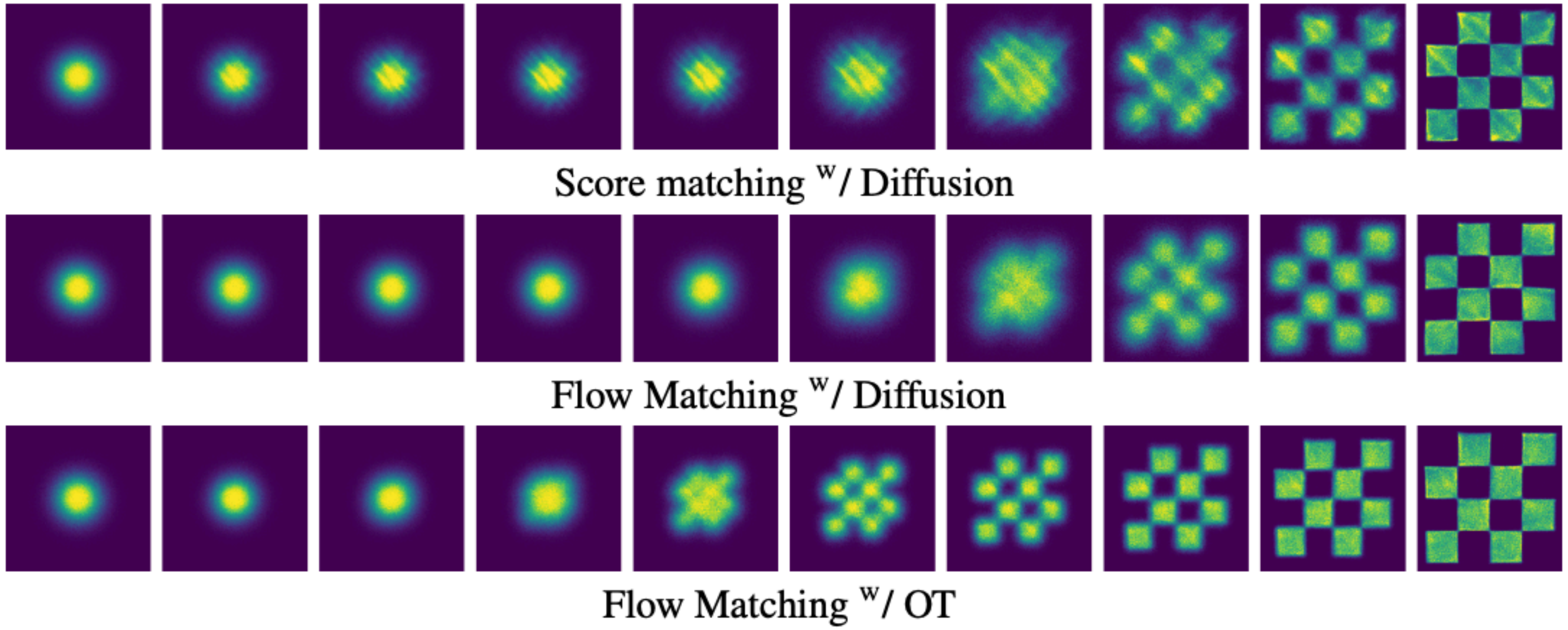
- *Theorem (informal)*: Among all schedulers Cond-OT is **Kinetic Optimal** for **sparse data in high dimension** ($n/d^{1/2} \to 0$).



[Shaul et al. 2023]

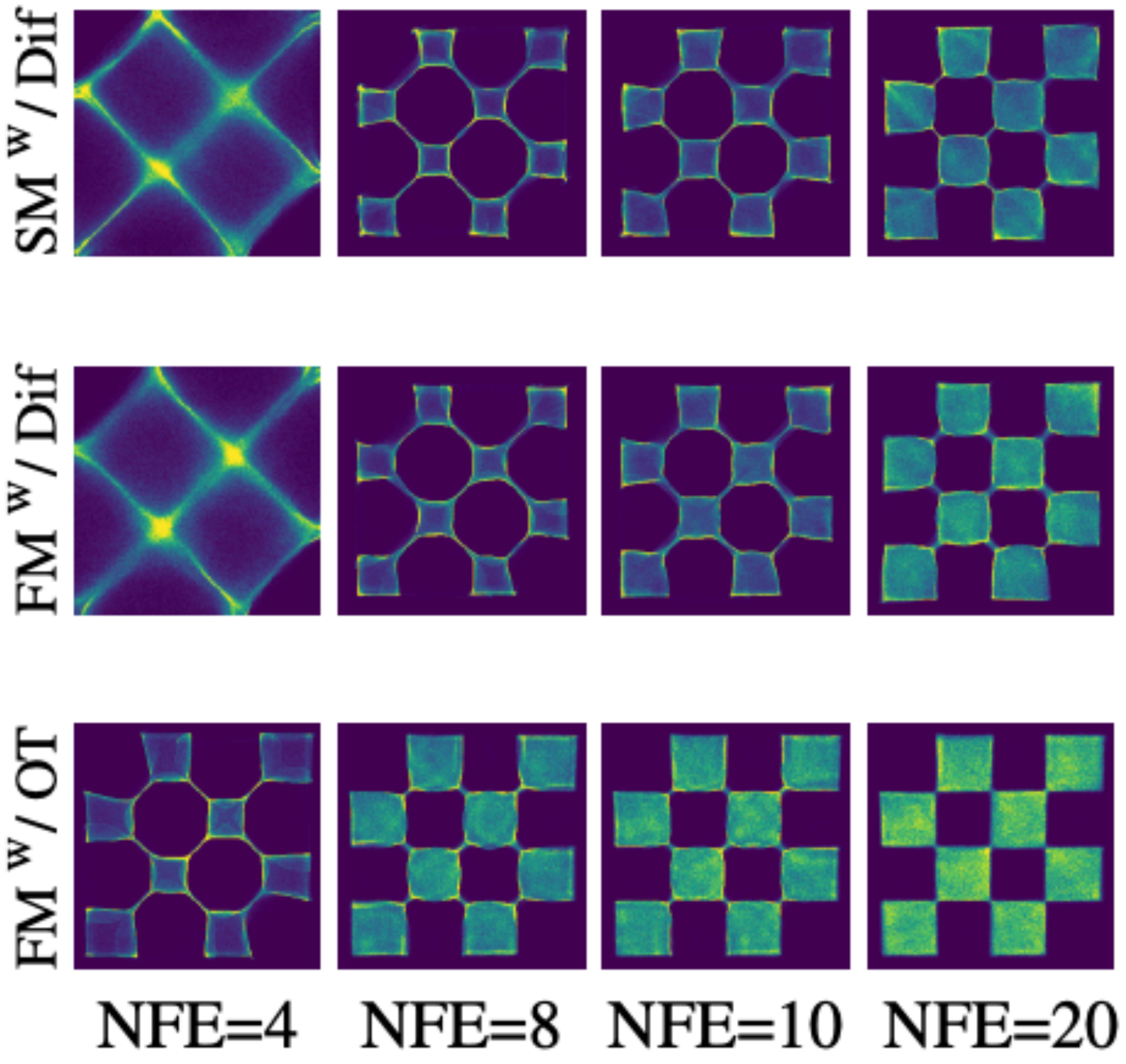
Experiments with Cond-OT

- FM-CondOT is faster/easier to sample

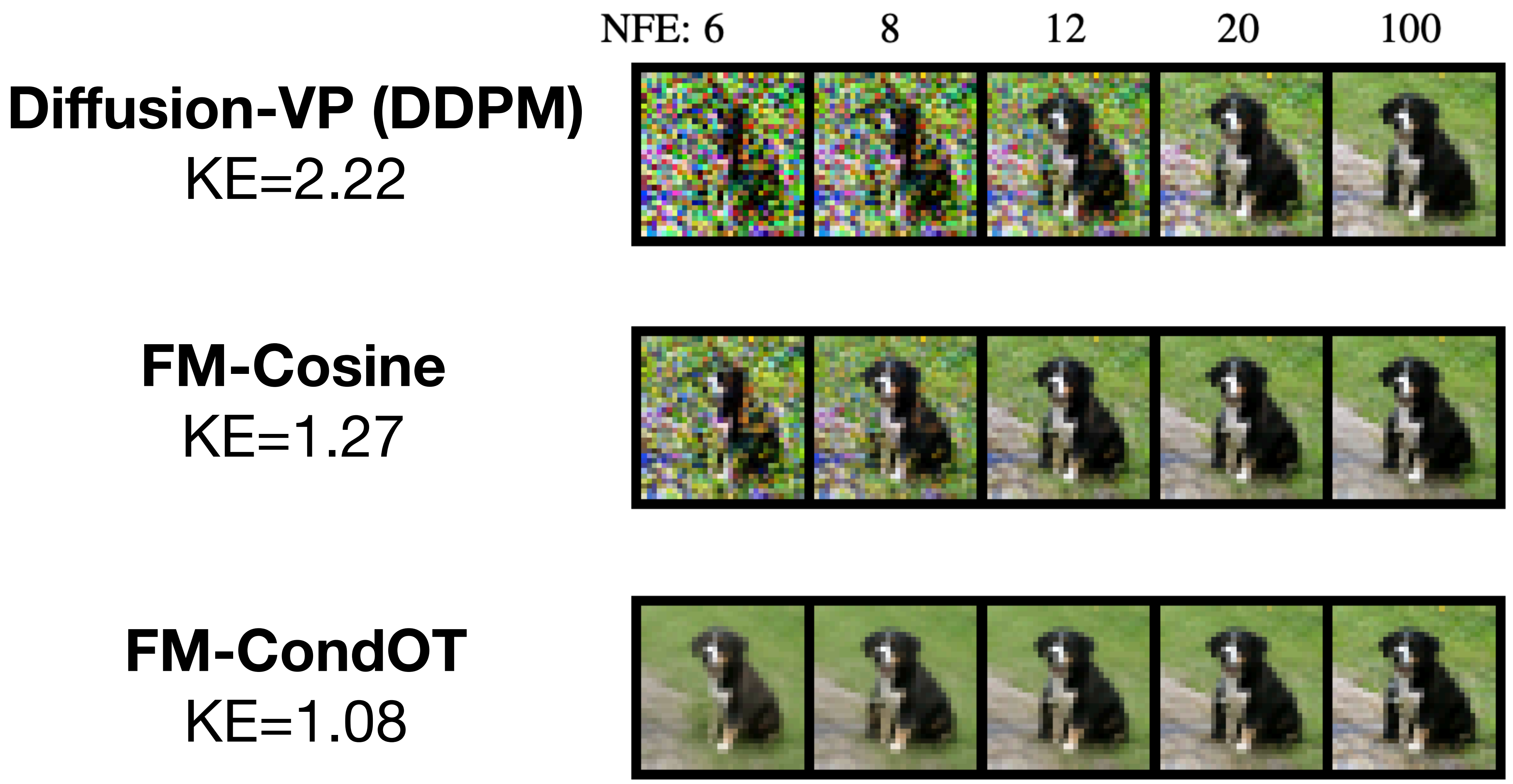


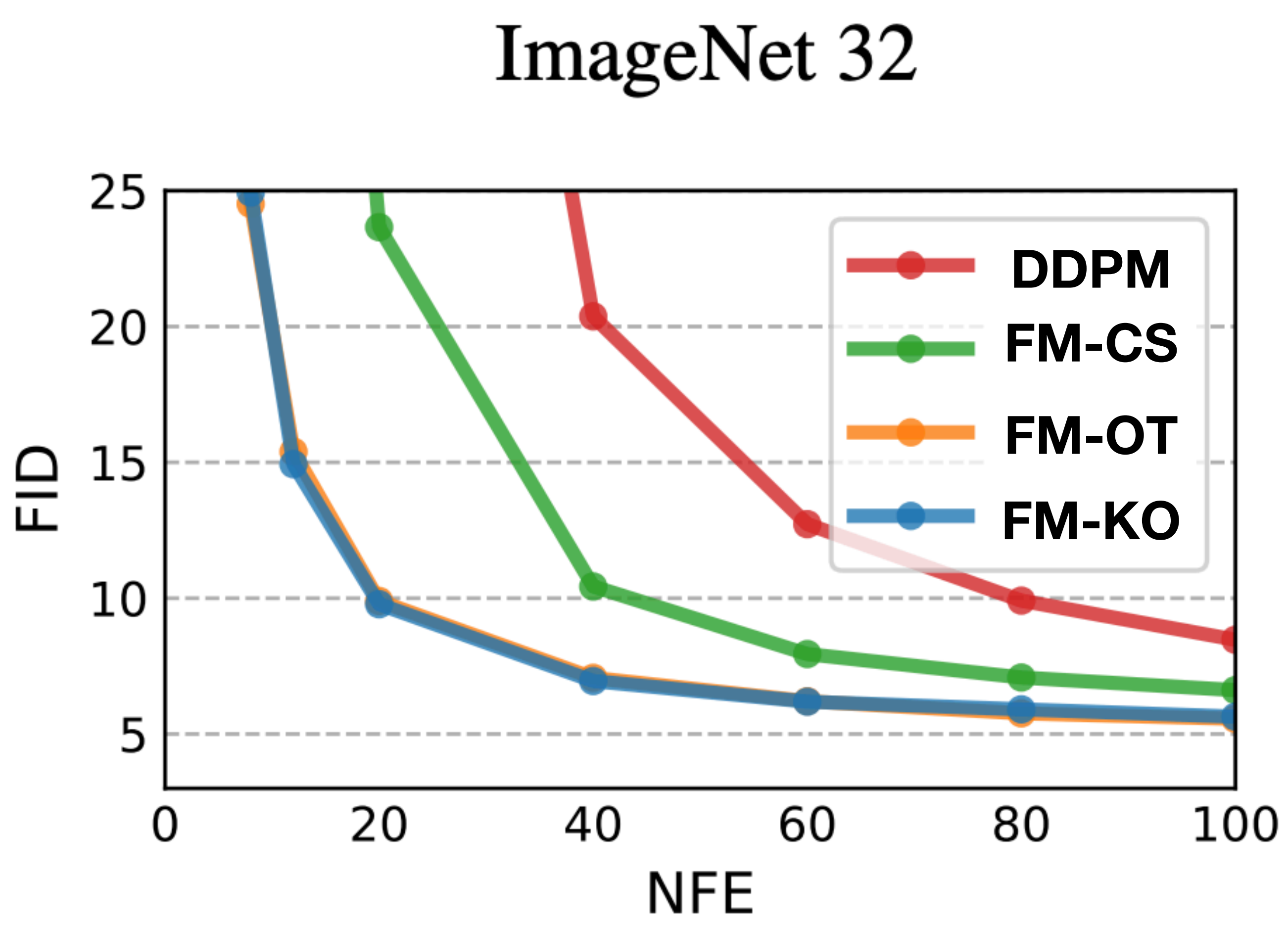
Experiments with Cond-OT

- FM-CondOT is faster/easier to sample



Experiments with Cond-OT





Generalizations

offered by FM

Generative models

$\cup$

Flows

$\cup$

Flow Matching

$\cup$

Diffusion Models

- **Arbitrary source p, general marginal paths.**
- General source-data couplings $q(x_0, x_1)$
- Non-euclidean spaces

Generalizations

offered by FM

Generative models

$\cup$

Flows

$\cup$

Flow Matching

$\cup$

Diffusion Models

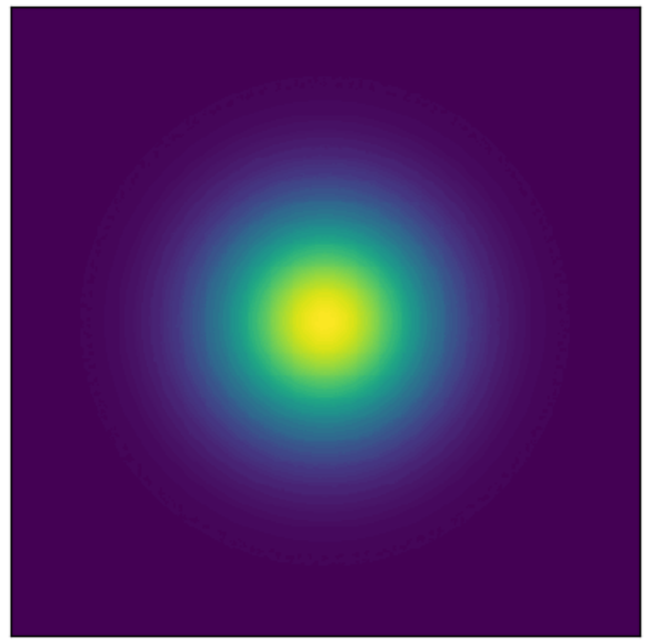
- Arbitrary source p, general marginal paths.
- **General source-data couplings** $q(x_0, x_1)$
- Non-euclidean spaces

Flow Matching

Supervision p_t, u_t

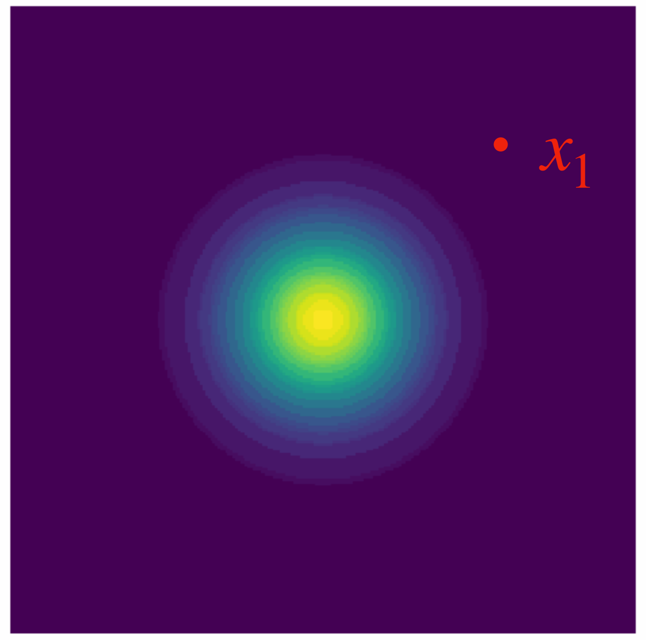
Marginal path

$$p_t(x) = \int p_t(x|x_1)q(x_1)dx_1$$



Conditional path

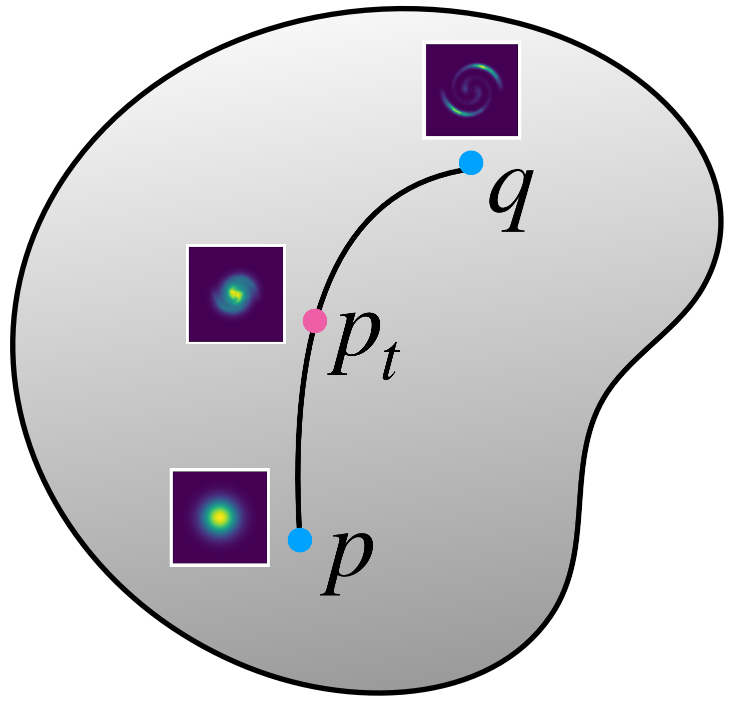
$$p_t(x|x_1)$$



Boundary conditions:

$$p_0 = p$$

$$p_1 = q$$



$$p_0(\cdot|x_1) = p$$

$$p_1(\cdot|x_1) = \delta_{x_1}$$

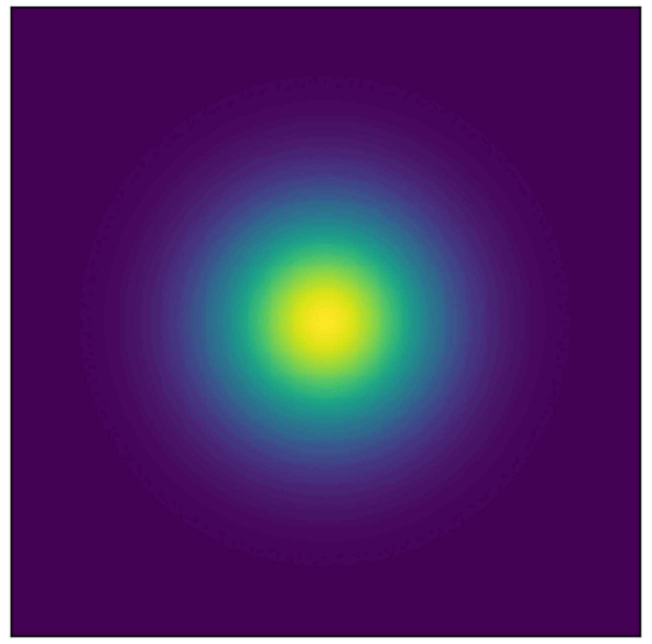
[Pooladian et al. 2023]
[Tong et al. 2023]

Flow Matching

Supervision p_t, u_t

Marginal path

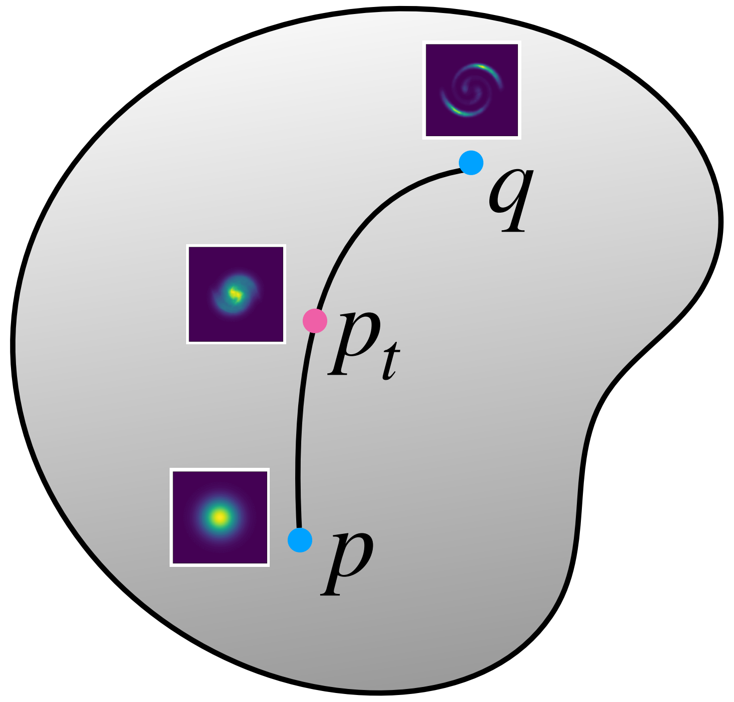
$$p_t(x) = \int p_t(x|x_1)q(x_1)dx_1$$



Boundary conditions:

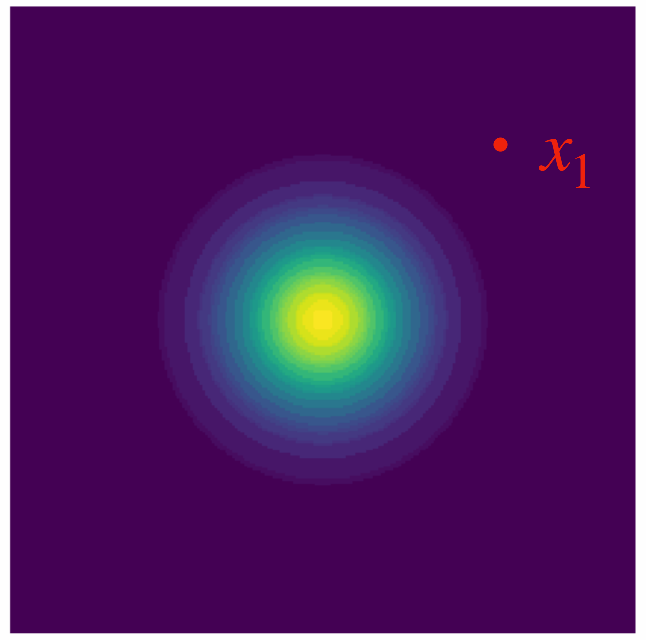
$$p_0 = p$$

$$p_1 = q$$



Conditional path

$$p_t(x|x_1)$$



$$\cancel{p_0(\cdot|x_1) = p} \rightarrow \int \underbrace{p_0(x_0|x_1)q(x_1)}_{q(x_0, x_1)}dx_1 = p(x_0)$$

$$p_1(\cdot|x_1) = \delta_{x_1}$$

[Pooladian et al. 2023]
[Tong et al. 2023]

Flow Matching

Arbitrary noise-data coupling

Algorithm 3: Flow Matching training.

Input : dataset and noise joint q

Initialize v^θ

while *not converged* **do**

- $t \sim \mathcal{U}([0,1])$ ▷ sample time
- $\boxed{(x_0, x_1) \sim q(x_0, x_1)}$ ▷ sample noise-data
- $x_t = \Psi_t(x_0|x_1)$ ▷ conditional flow
- Gradient step with $\nabla_\theta \|v_t^\theta(x_t) - \dot{x}_t\|^2$

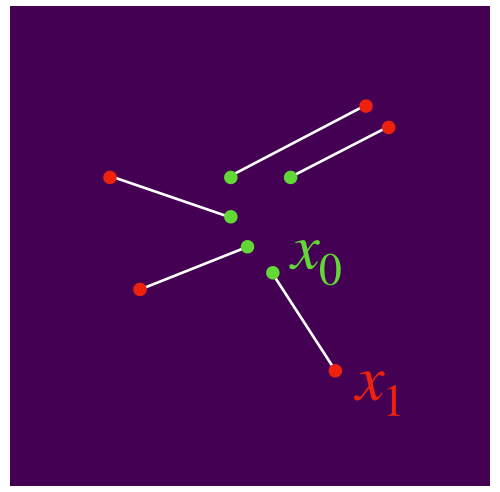
Output: v^θ

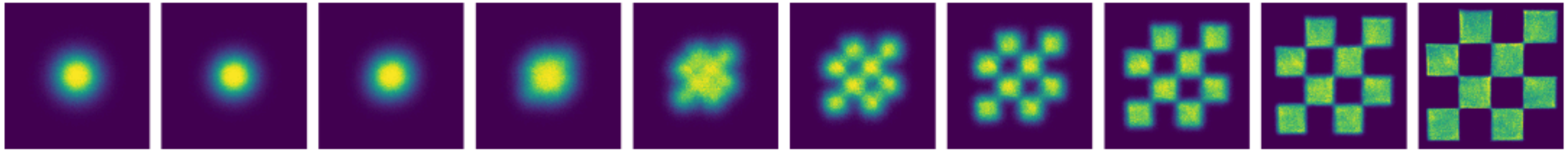
[Pooladian et al. 2023]
[Tong et al. 2023]

Flow Matching

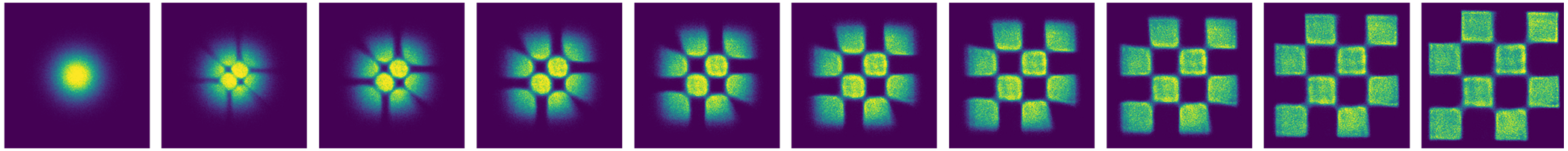
Batch-OT

$$\min \mathbb{E}_{t,\, q(x_0,x_1)} \| v_t(x_t) - \dot{x}_t \|^2$$





Flow Matching / Cond-OT



Flow Matching / Batch-OT

[Pooladian et al. 2023]

Generalizations

offered by FM

Generative models

$\cup$

Flows

$\cup$

Flow Matching

$\cup$

Diffusion Models

- Arbitrary source p, general marginal paths.
- **General source-data couplings** $q(x_0, x_1)$
- Non-euclidean spaces

Generalizations

offered by FM

Generative models

$\cup$

Flows

$\cup$

Flow Matching

$\cup$

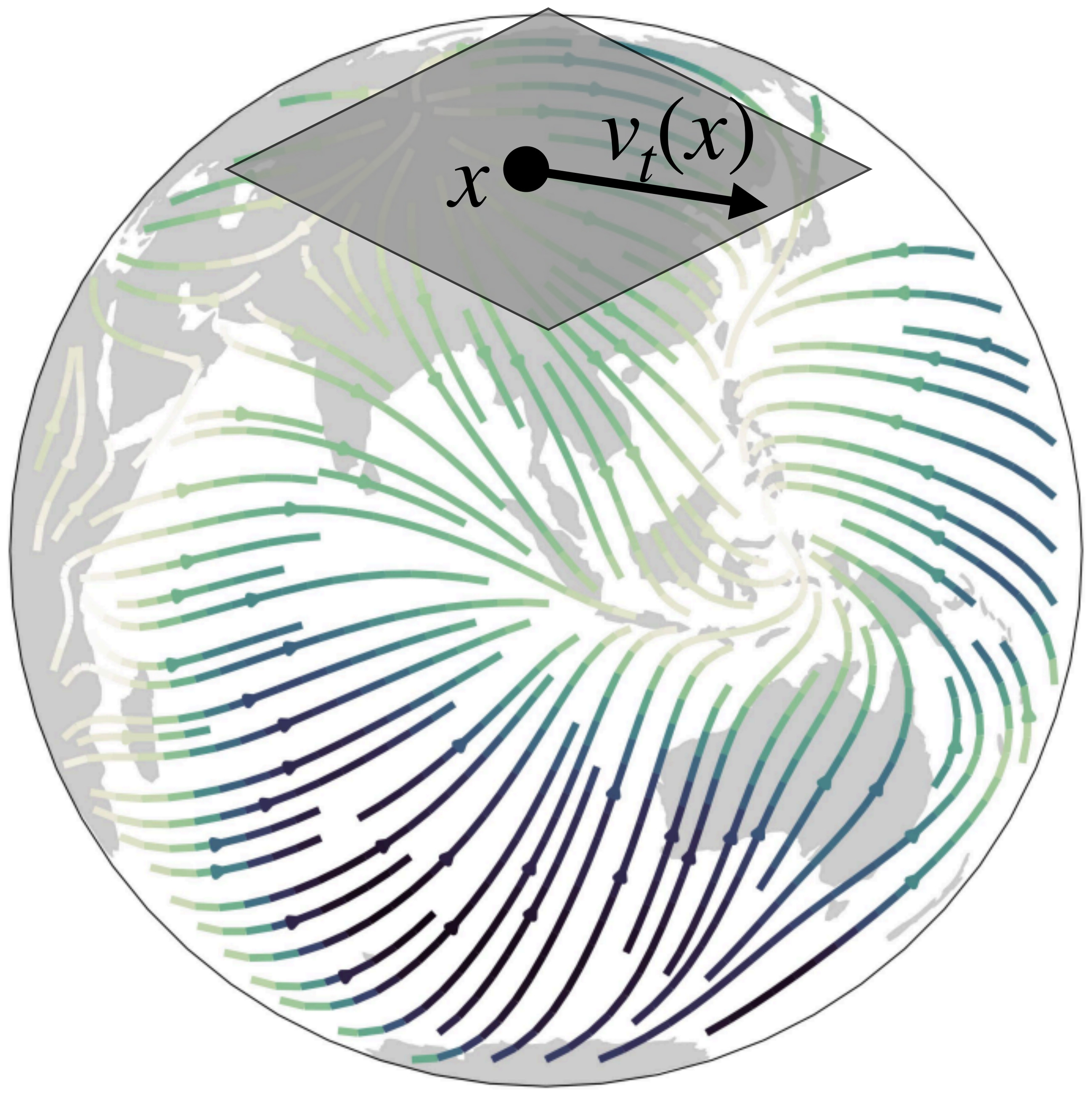
Diffusion Models

- Arbitrary source p, general marginal paths.
- General source-data couplings $q(x_0, x_1)$
- **Non-euclidean spaces**

Flows on General Geometries

$$v : [0,1] \times \mathcal{M} \to T\mathcal{M}$$

$\mathcal{M}$



Flow ODE

$$\dot{x}_t = v_t(x_t)$$

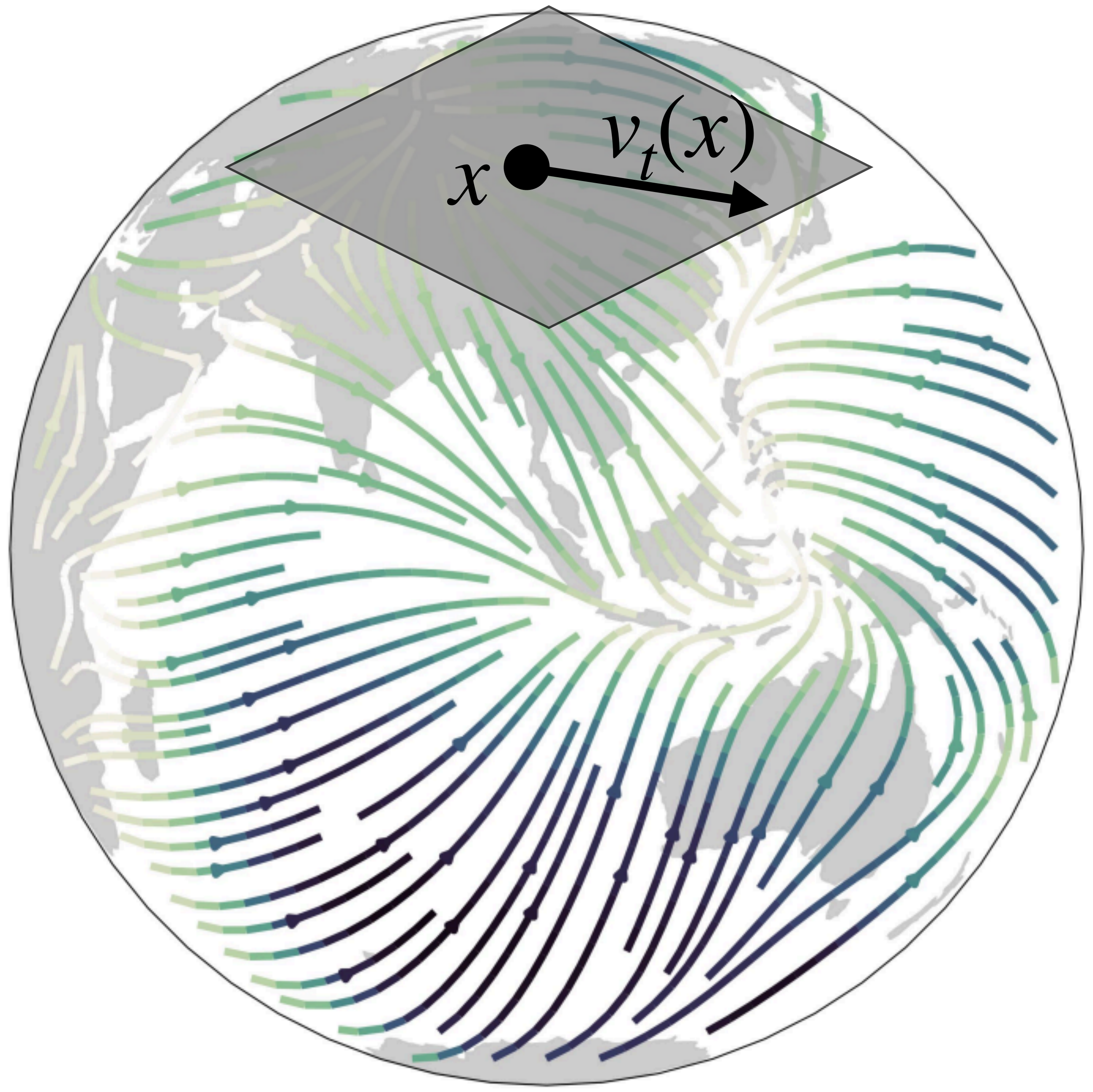
[Mathieu and Nickel 2020]

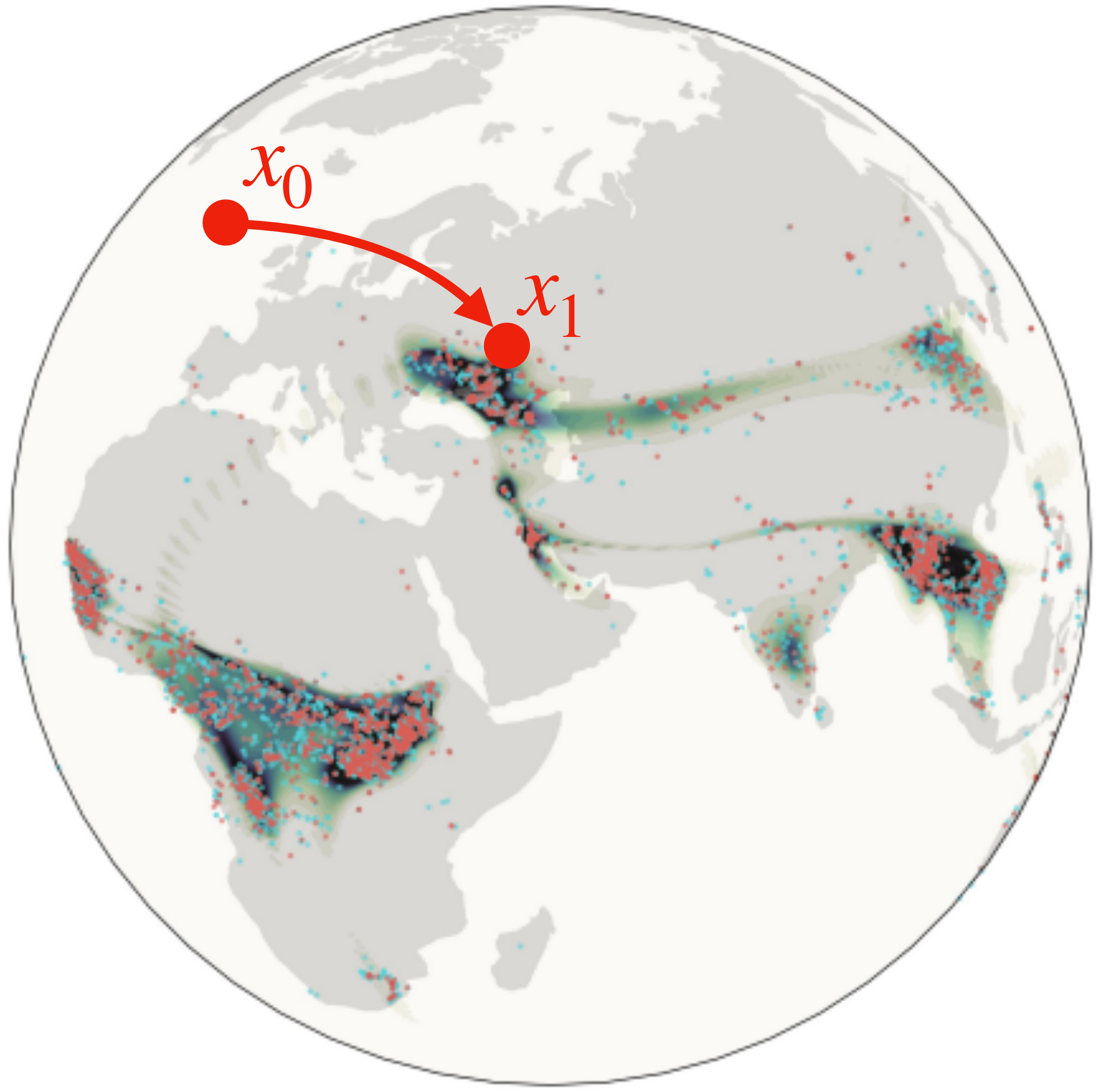
Flows on General Geometries

$v : [0,1] \times \mathcal{M} \rightarrow T\mathcal{M}$

$x_t = \Psi_t(x_0)$

$\mathcal{M}$





Flow ODE

$$\dot{x}_t = v_t(x_t)$$

Continuity Equation PDE

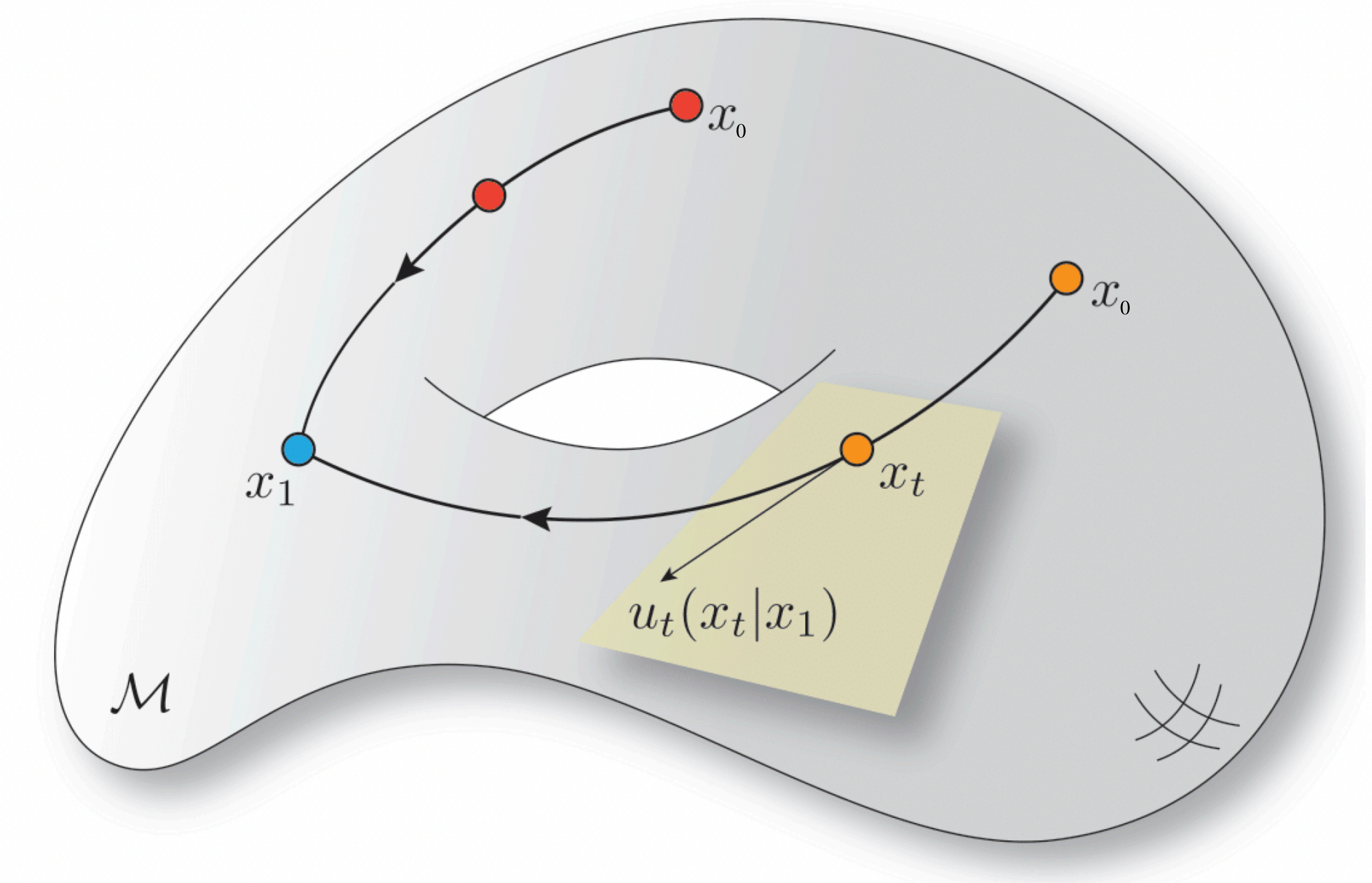
$$\dot{p}_t = -\operatorname{div}_g(p_t v_t)$$

[Mathieu and Nickel 2020]

Flow Matching on Manifolds

$$L_{\mathrm{RCFM}}(q \| p_1) = \min \mathbb{E}_{t,\, q(x_0, x_1)} \| v_t(x_t) - \dot{x}_t \|_{g}^2$$

$$x_t = \Psi_t(x_0 \,|\, x_1)$$



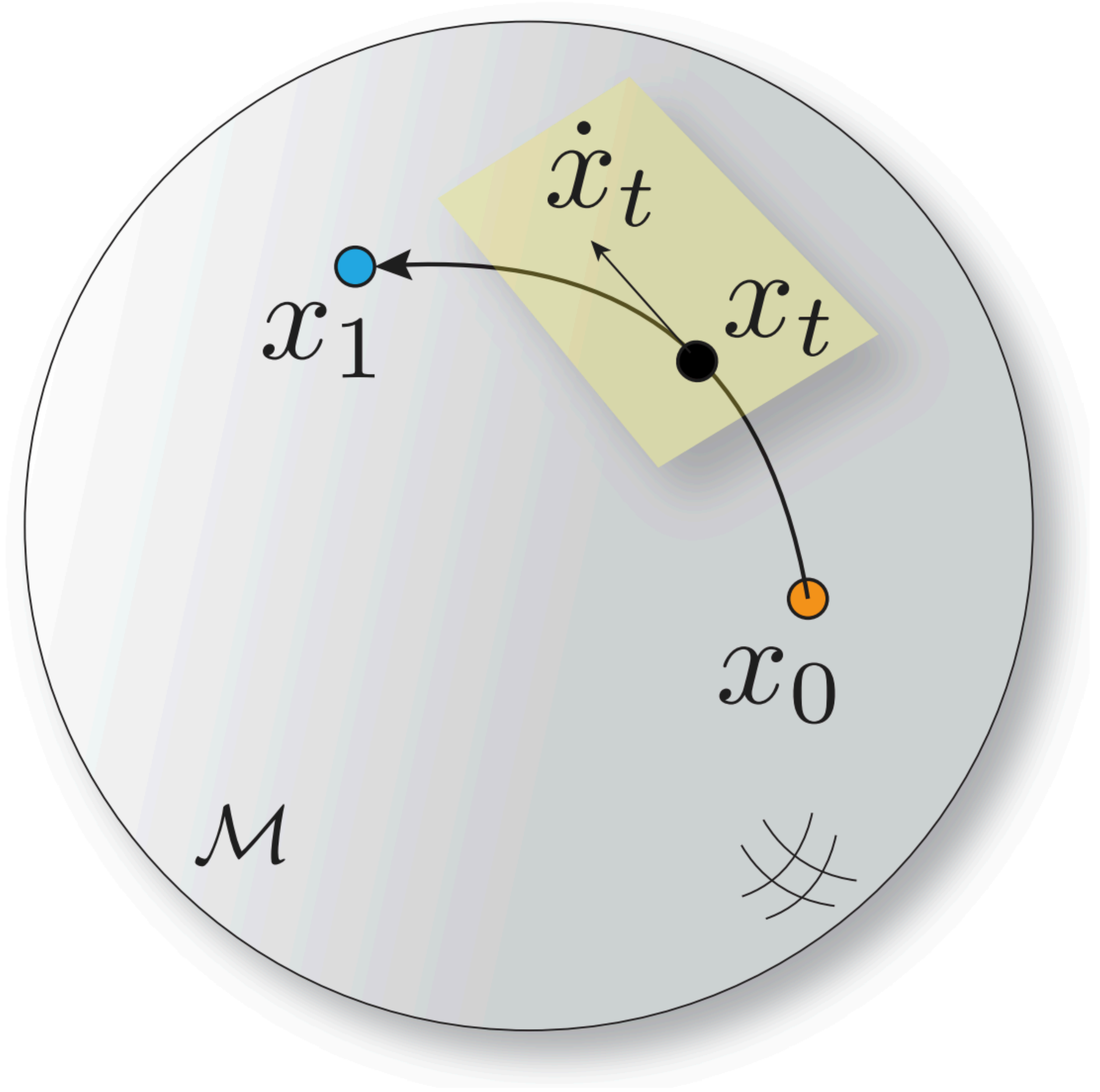
[Chen and L 2023]

Flow Matching on (simple) Manifolds

In Euclidean space:

$$d(x_0, x_1) = \|x_0 - x_1\|$$

$$x_t = (1 - t)x_0 + tx_1$$



Closed form geodesics:

$$d(x_0, x_1)$$

$$x_t = \exp_{x_0}\left(t \log_{x_0} x_1\right)$$

Hypersphere, Hyperbolic Space, Torus, Transformation Groups

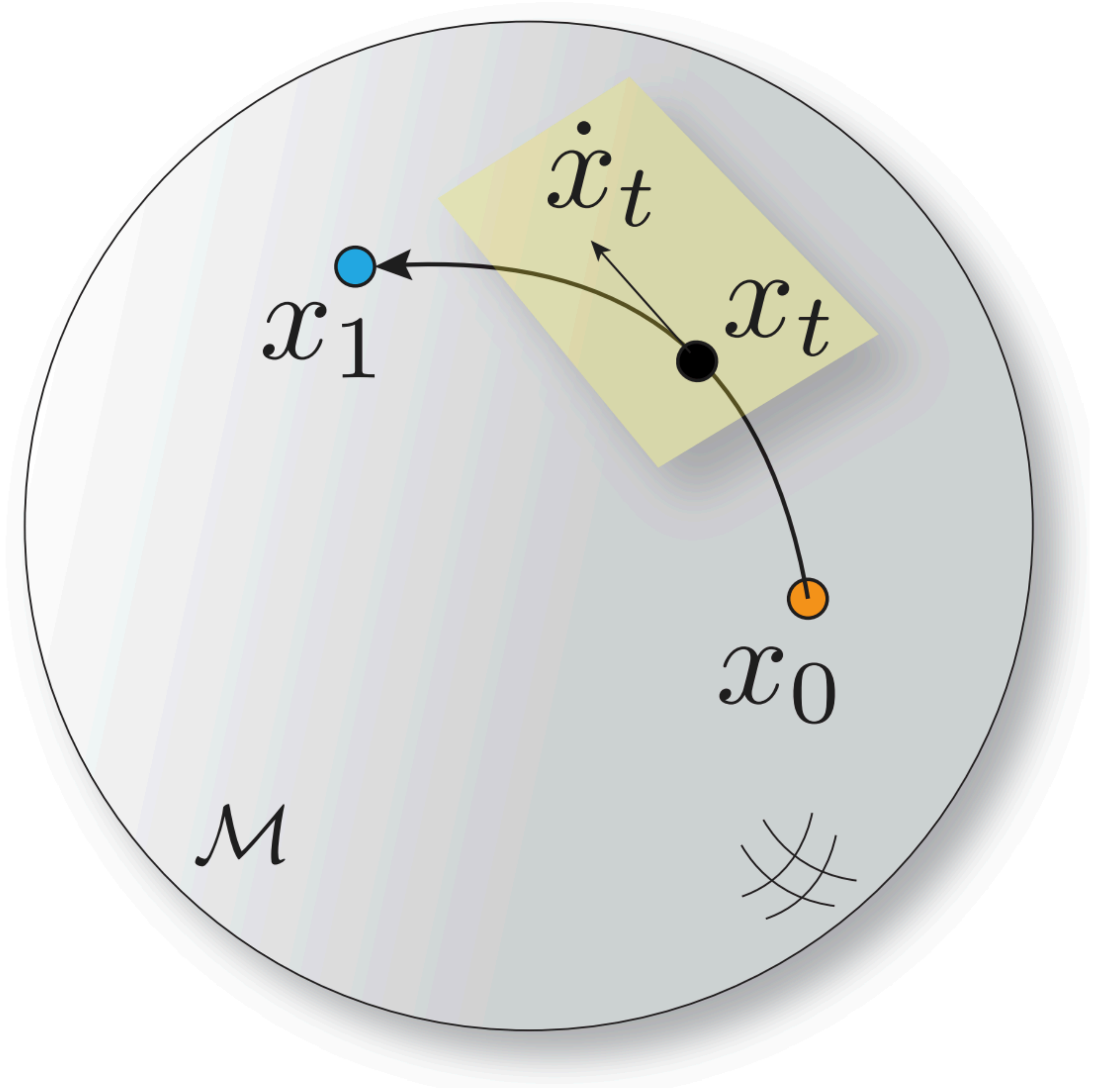
[Chen and L 2023]

Flow Matching on (simple) Manifolds

In Euclidean space:

$$d(x_0, x_1) = \|x_0 - x_1\|$$

$$x_t = (1-t)x_0 + tx_1$$

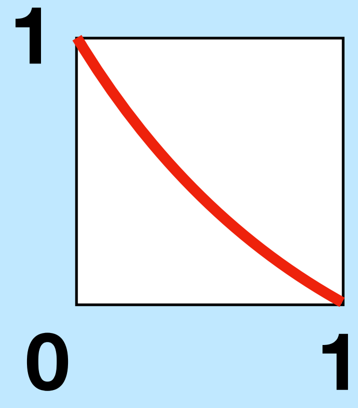


Closed form geodesics:

$$d(x_0, x_1)$$

$$x_t = \exp_{x_0}\left(t \log_{x_0} x_1\right)$$

$$d(x_t, x_1) = \kappa(t) d(x_0, x_1)$$



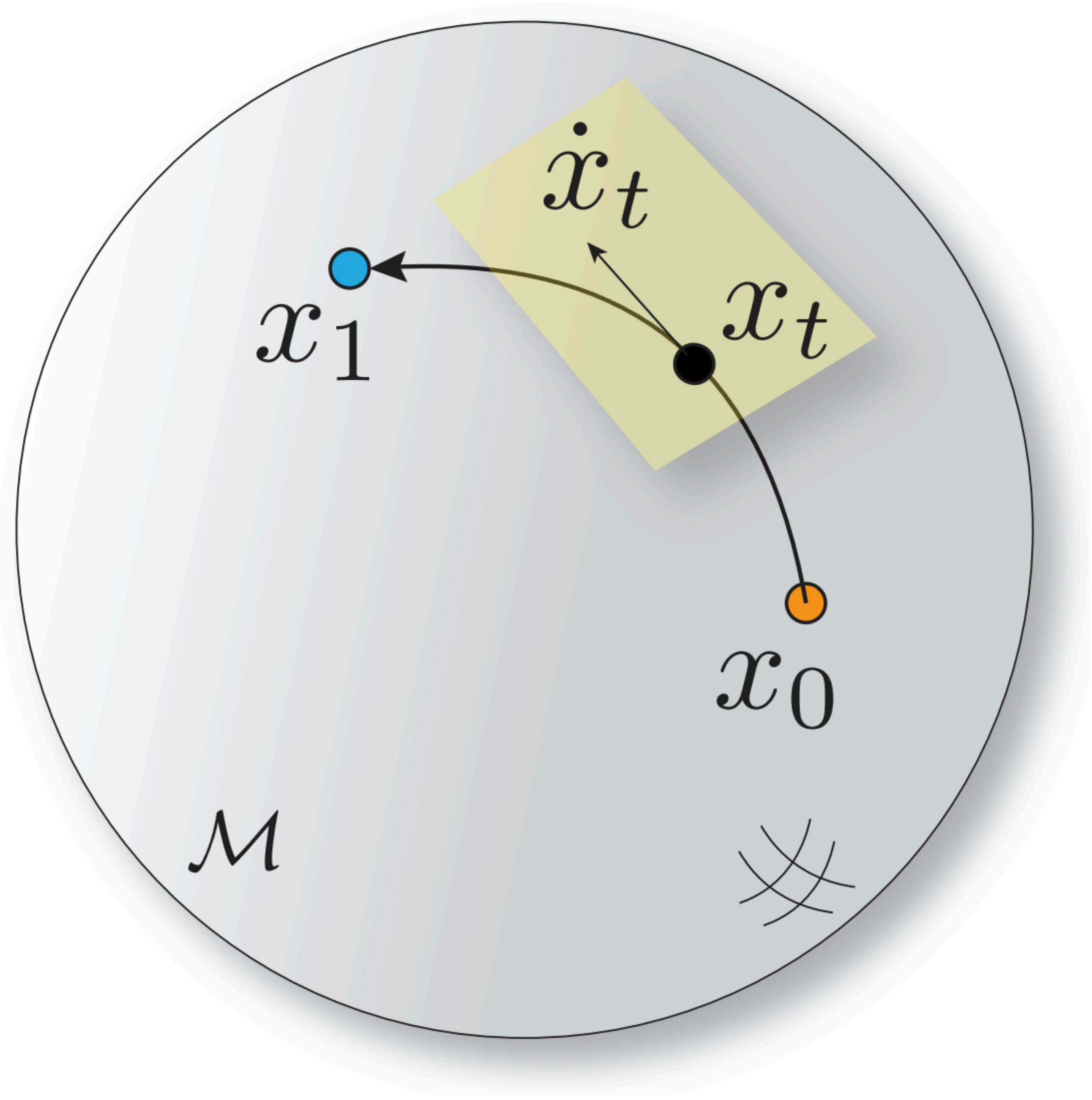
[Chen and L 2023]

Flow Matching on (simple) Manifolds

In Euclidean space:

$$d(x_0, x_1) = \|x_0 - x_1\|$$

$$x_t = (1 - t)x_0 + tx_1$$

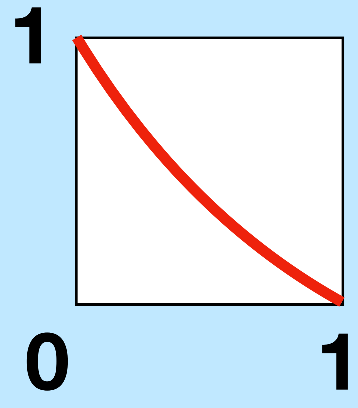


Closed form geodesics:

$$d(x_0, x_1)$$

$$x_t = \exp_{x_0}\left(t \log_{x_0} x_1\right)$$

$$d(x_t, x_1) = \kappa(t) d(x_0, x_1)$$



$$x_t = \exp_{x_0}\left(\kappa(t) \log_{x_0} x_1\right)$$

[Chen and L 2023]

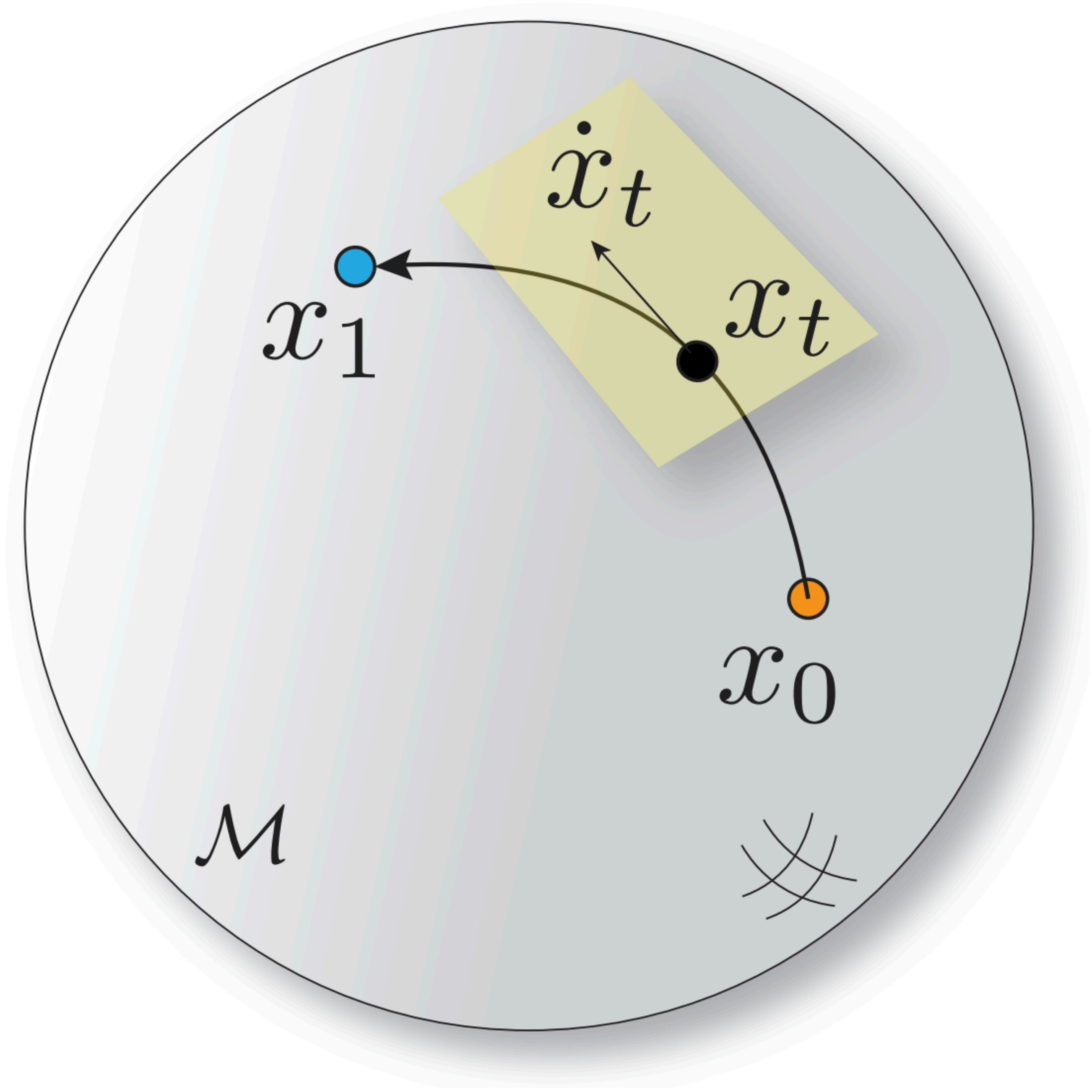
Flow Matching on (simple) Manifolds

In Euclidean space:

$$d(x_0, x_1) = \|x_0 - x_1\|$$

$$x_t = (1-t)x_0 + tx_1$$

$$x_t = (1 - \kappa(t))x_0 + \kappa(t)x_1$$

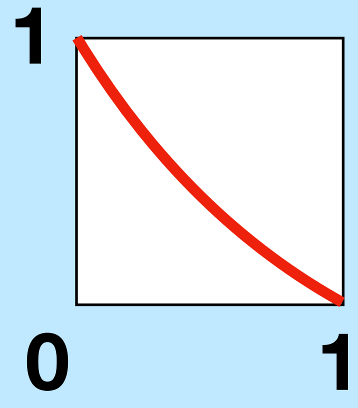


Closed form geodesics:

$$d(x_0, x_1)$$

$$x_t = \exp_{x_0}\left(t \log_{x_0} x_1\right)$$

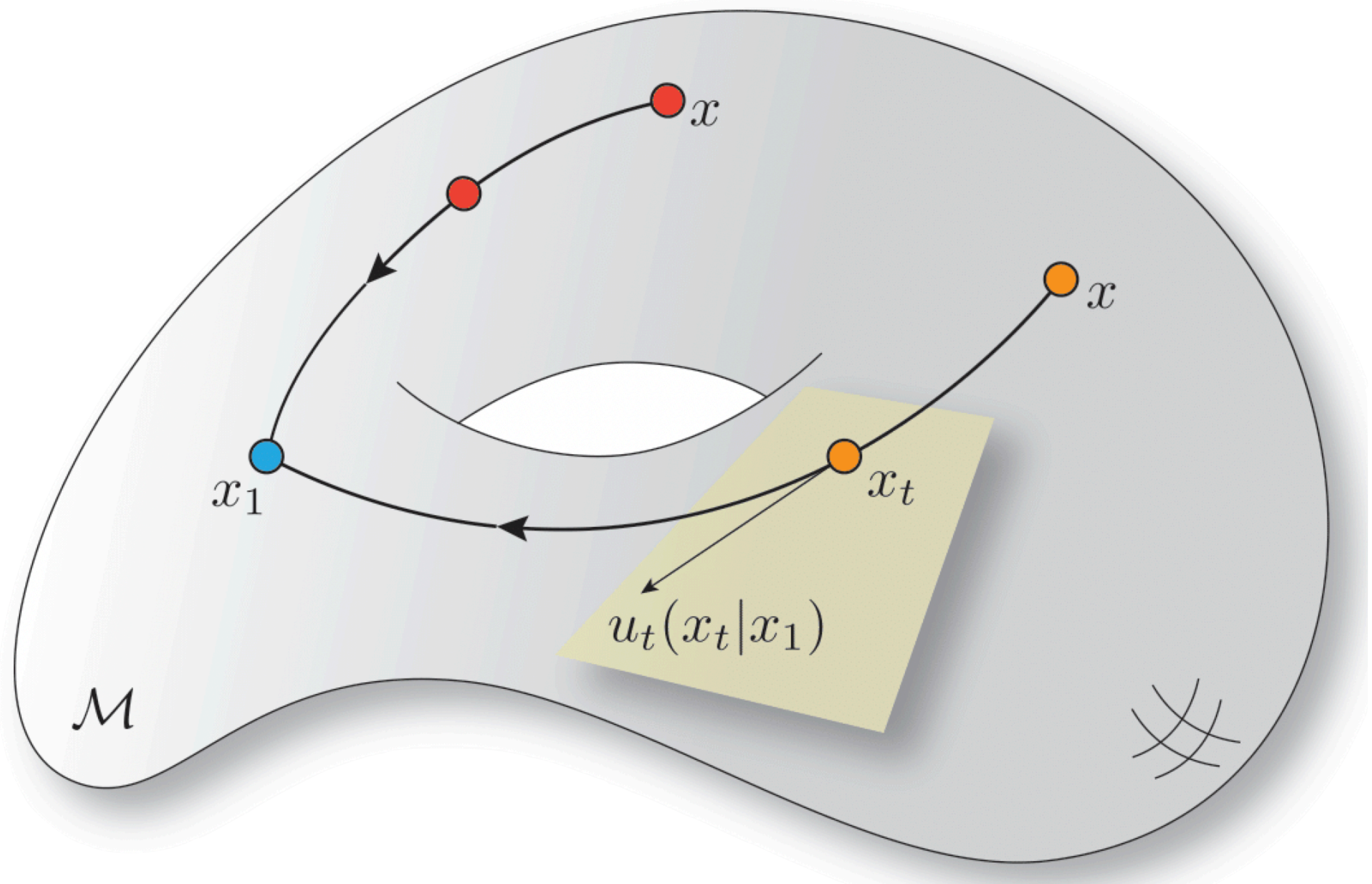
$$d(x_t, x_1) = \kappa(t) d(x_0, x_1)$$



$$x_t = \exp_{x_0}\left(\kappa(t) \log_{x_0} x_1\right)$$

[Chen and L 2023]

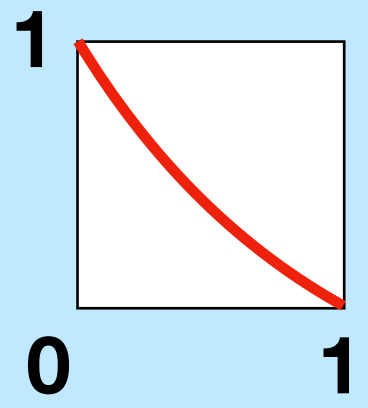
Flow Matching on (general) Manifolds



Pre-metric d:

1. *Non-negative:* $\mathrm{d}(x, y) \geq 0$ for all $x, y \in \mathcal{M}$.
2. *Positive:* $\mathrm{d}(x, y) = 0$ iff $x = y$.
3. *Non-degenerate:* $\nabla \mathrm{d}(x, y) \neq 0$ iff $x \neq y$.

$$d(x_t, x_1) = \kappa(t) d(x_0, x_1)$$

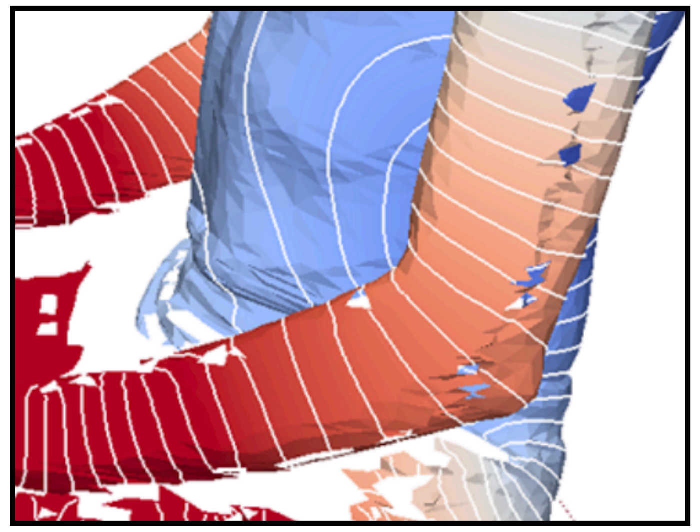


$$u_t(x \,|\, x_1) = \frac{d \log \kappa(t)}{dt} \frac{\nabla_x \frac{1}{2} d^2(x, x_1)}{\|d(x, x_1)\|_g^2}$$

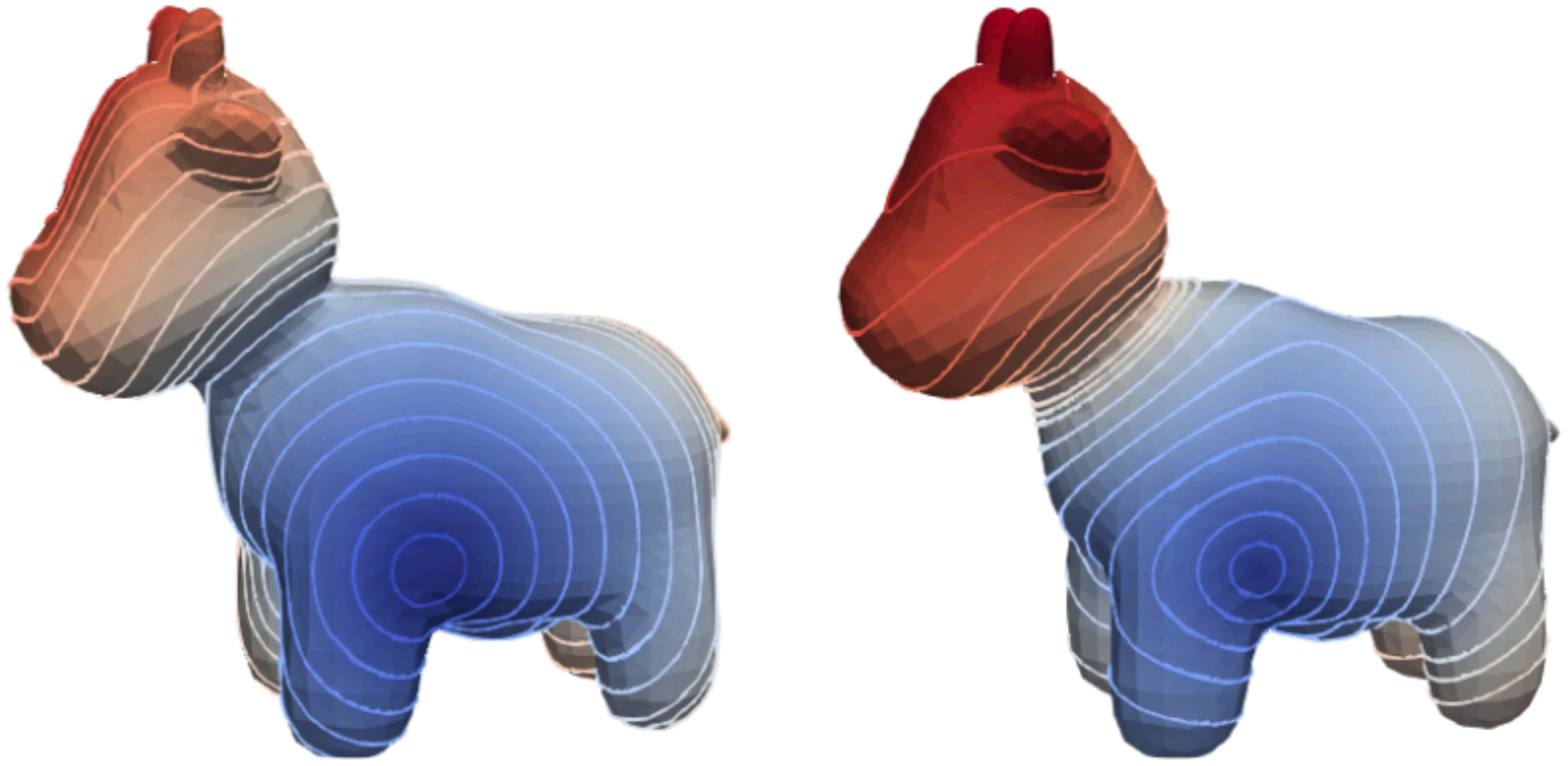
Spectral Distances

$$d(x,y) = \sum_{i=1}^{\infty} w(\lambda_i)\big(\phi_i(x) - \phi_i(y)\big)^2$$

$$\Delta_g \phi_i = \lambda_i \phi_i$$



Geodesic Biharmonic



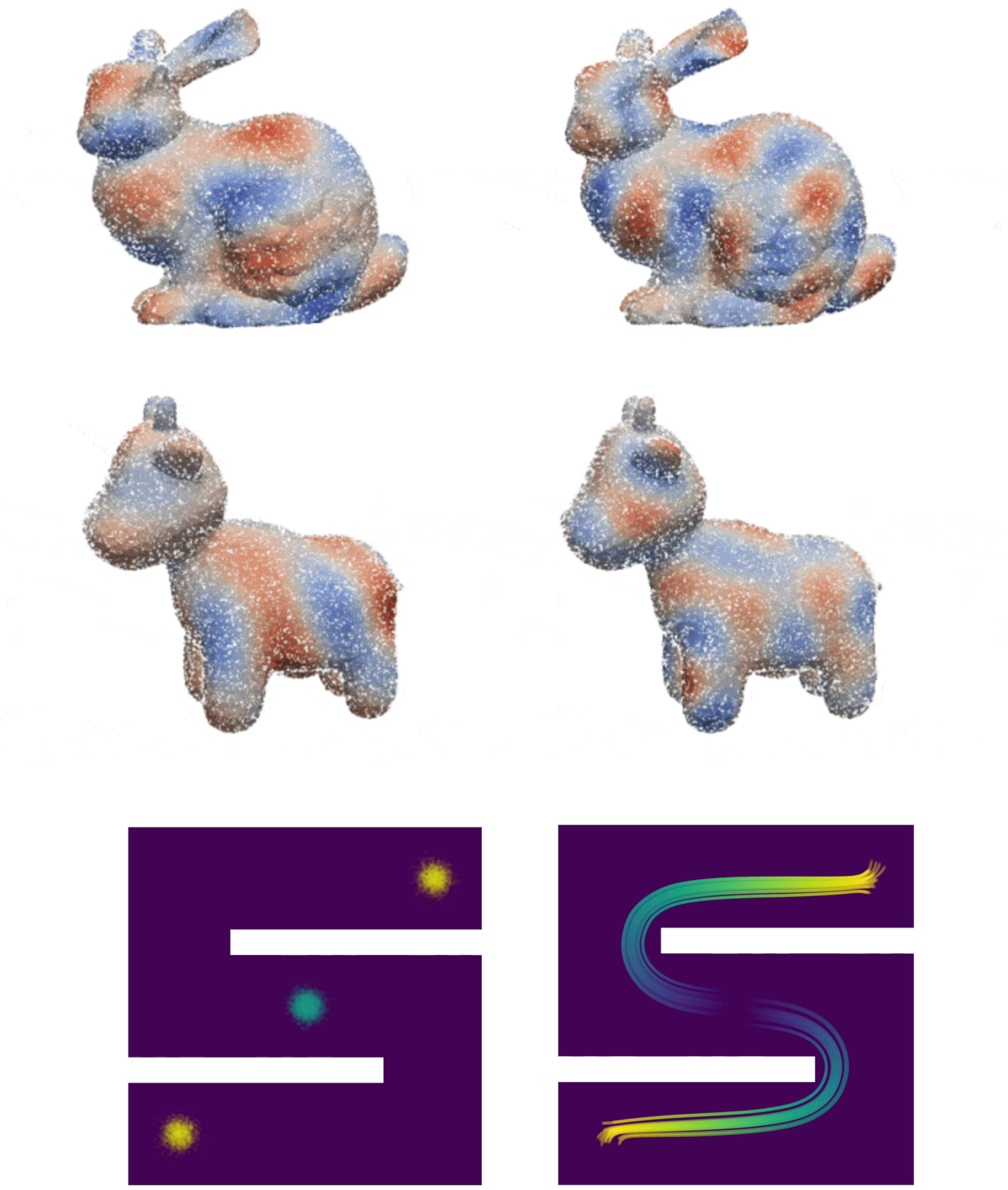
Geodesic Biharmonic

[Chen and L 2023]

Flow Matching on Manifolds

Algorithm 1 Riemannian CFM

Require: base distribution p, target q
  Initialize parameters θ of v_t
  while not converged **do**
    sample time $t \sim \mathcal{U}(0, 1)$
    sample training example $x_1 \sim q$
    sample noise $x_0 \sim p$
    if simple geometry **then**
      $x_t = \texttt{exp}_{x_0}(\kappa(t)\texttt{log}_{x_0}(x_1))$
    else if general geometry **then**
      $x_t = \texttt{solve\_ODE}([0, t], x_0, u_t(x|x_1))$
    end if
    $\ell(\theta) = \|v_t(x_t; \theta) - \dot{x}_t\|_g^2$
    $\theta = \texttt{optimizer\_step}(\ell(\theta))$
  end while



[Chen and L 2023]

Comparison with Diffusion on Manifolds

Algorithm 2 Riemannian Diffusion Models

Require: base distribution $p(x_T)$, target $q(x_0)$
Initialize parameters θ of s_t
while not converged **do**
sample time $t \sim \mathcal{U}(0, T)$
sample training example $x_0 \sim q(x_0)$

Simulation

% simulate Geometric Random Walk
$x_t = \texttt{solve\_SDE}([0,t], x_0)$

if denoising score matching **then**
% approximate conditional score

Adds bias

$$\nabla \log p_t(x|x_0) \approx \begin{cases} \text{eig-expansion} \\ \text{Varhadan} \end{cases}$$

$\ell(\theta) = \|s_t(x_t;\theta) - \nabla \log p_t(x|x_0)\|_g^2$
else if implicit score matching **then**
% estimate Riemannian divergence
sample $\varepsilon \sim \mathcal{N}(0, I)$

Adds variance

$\text{div}_g s_t \approx \varepsilon^\top \frac{\partial s_t}{\partial x_t} \varepsilon + \frac{1}{2} s_t^\top \frac{\partial \log \det g(x_t)}{\partial x_t}$
$\ell(\theta) = \frac{1}{2}\|s_t(x_t;\theta)\|_g^2 + \text{div}_g s_t$
end if

$\theta = \texttt{optimizer\_step}(\ell(\theta))$
end while

Algorithm 3 Riemannian Flow Matching

Require: base distribution $p(x_0)$, target $q(x_1)$
Initialize parameters θ of v_t
while not converged **do**
sample time $t \sim \mathcal{U}(0, 1)$
sample training example $x_1 \sim q(x_1)$
sample noise $x_0 \sim p(x_0)$

if simple geometry **then**
$x_t = \texttt{exp}_{x_0}(\kappa(t)\texttt{log}_{x_0}(x_1))$
else if general geometry **then**
$x_t = \texttt{solve\_ODE}([0,t], x_0, u_t(x|x_1))$
end if

% closed-form regression target $u_t(x_t|x_1)$
$\ell(\theta) = \|v_t(x_t;\theta) - u_t(x_t|x_1)\|_g^2$

$\theta = \texttt{optimizer\_step}(\ell(\theta))$
end while

Comparison with Diffusion on Manifolds

Algorithm 2 Riemannian Diffusion Models

Require: base distribution $p(x_T)$, target $q(x_0)$
  Initialize parameters θ of s_t
  while not converged **do**
    sample time $t \sim \mathcal{U}(0, T)$
    sample training example $x_0 \sim q(x_0)$

Simulation

    % simulate Geometric Random Walk
    $x_t = \texttt{solve\_SDE}([0, t], x_0)$

    if denoising score matching **then**
      % approximate conditional score

Adds bias

      $\nabla \log p_t(x|x_0) \approx \begin{cases} \text{eig-expansion} \\ \text{Varhadan} \end{cases}$
      $\ell(\theta) = \|s_t(x_t; \theta) - \nabla \log p_t(x|x_0)\|_g^2$
    else if implicit score matching **then**
      % estimate Riemmanian divergence
      sample $\varepsilon \sim \mathcal{N}(0, I)$

Adds variance

      $\text{div}_g s_t \approx \varepsilon^\mathsf{T} \frac{\partial s_t}{\partial x_t} \varepsilon + \frac{1}{2} s_t^\mathsf{T} \frac{\partial \log \det g(x_t)}{\partial x_t}$
      $\ell(\theta) = \frac{1}{2} \|s_t(x_t; \theta)\|_g^2 + \text{div}_g s_t$
    end if

    $\theta = \texttt{optimizer\_step}(\ell(\theta))$
  end while

Algorithm 3 Riemannian Flow Matching

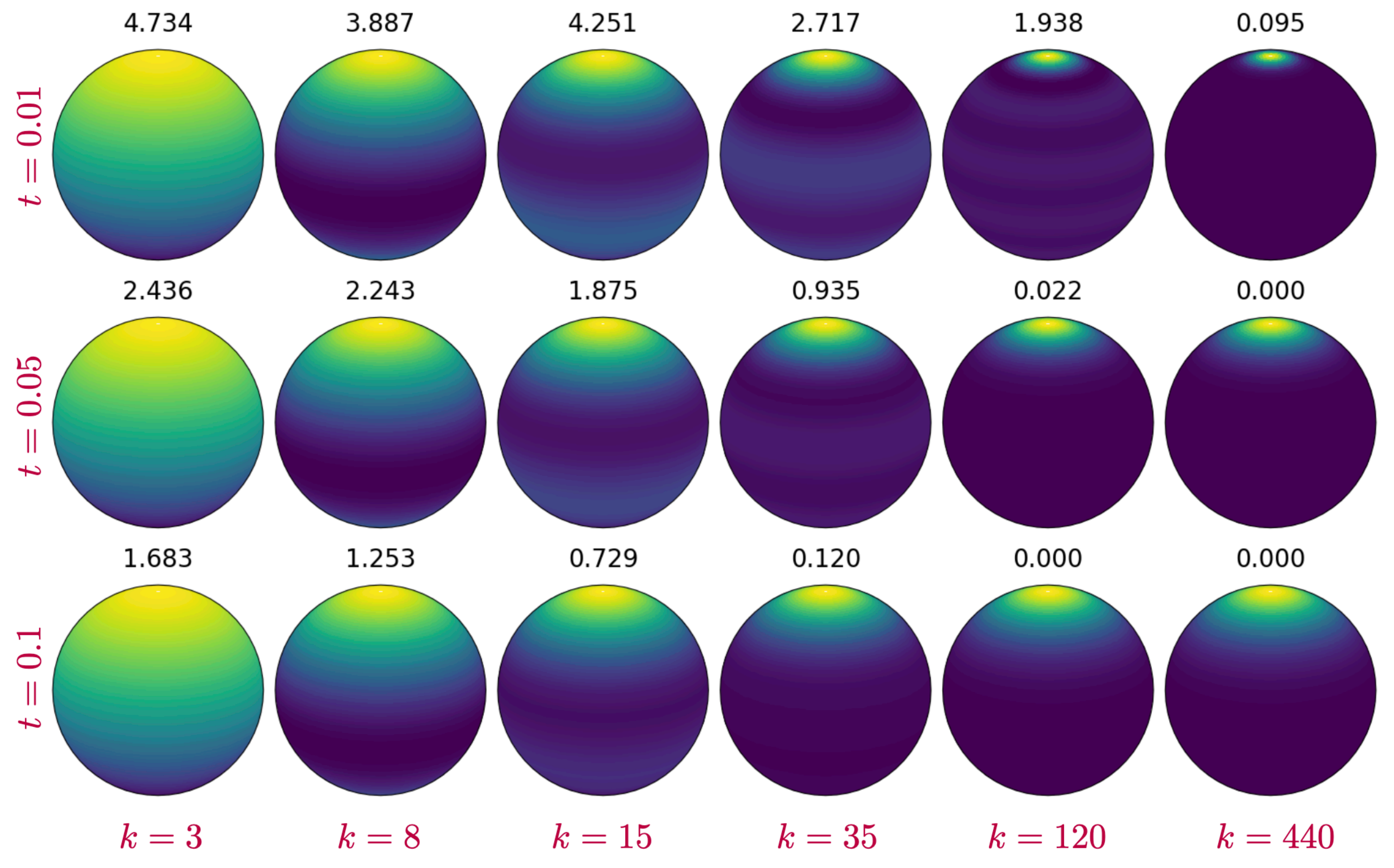
Require: base distribution $p(x_0)$, target $q(x_1)$
  Initialize parameters θ of v_t
  while not converged **do**
    sample time $t \sim \mathcal{U}(0, 1)$
    sample training example $x_1 \sim q(x_1)$
    sample noise $x_0 \sim p(x_0)$

    if simple geometry **then**
      $x_t = \texttt{exp}_{x_0}(\kappa(t)\texttt{log}_{x_0}(x_1))$
    else if general geometry **then**
      $x_t = \texttt{solve\_ODE}([0, t], x_0, u_t(x|x_1))$
    end if

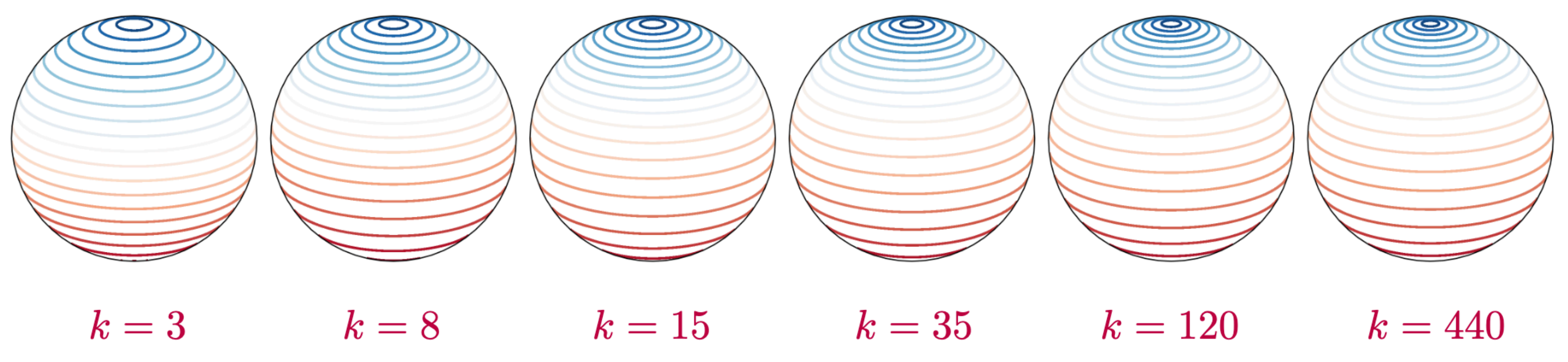
    % closed-form regression target $u_t(x_t|x_1)$
    $\ell(\theta) = \|v_t(x_t; \theta) - u_t(x_t|x_1)\|_g^2$

    $\theta = \texttt{optimizer\_step}(\ell(\theta))$
  end while

Heat Kernel $p_t(x \,|\, x_1)$



Biharmonic Distance $d(x, x_1)$



Comparison with Diffusion on Manifolds

Algorithm 2 Riemannian Diffusion Models

Require: base distribution $p(x_T)$, target $q(x_0)$
  Initialize parameters θ of s_t
  while not converged **do**
    sample time $t \sim \mathcal{U}(0, T)$
    sample training example $x_0 \sim q(x_0)$

Simulation

    % simulate Geometric Random Walk
    $x_t = \texttt{solve\_SDE}([0, t], x_0)$

Adds bias

    if denoising score matching **then**
      % approximate conditional score
      $\nabla \log p_t(x|x_0) \approx \begin{cases} \text{eig-expansion} \\ \text{Varhadan} \end{cases}$
      $\ell(\theta) = \|s_t(x_t; \theta) - \nabla \log p_t(x|x_0)\|_g^2$

Adds variance

    else if implicit score matching **then**
      % estimate Riemmanian divergence
      sample $\varepsilon \sim \mathcal{N}(0, I)$
      $\text{div}_g s_t \approx \varepsilon^\top \frac{\partial s_t}{\partial x_t} \varepsilon + \frac{1}{2} s_t^\top \frac{\partial \log \det g(x_t)}{\partial x_t}$
      $\ell(\theta) = \frac{1}{2} \|s_t(x_t; \theta)\|_g^2 + \text{div}_g s_t$
    end if

    $\theta = \texttt{optimizer\_step}(\ell(\theta))$
  end while

Algorithm 3 Riemannian Flow Matching

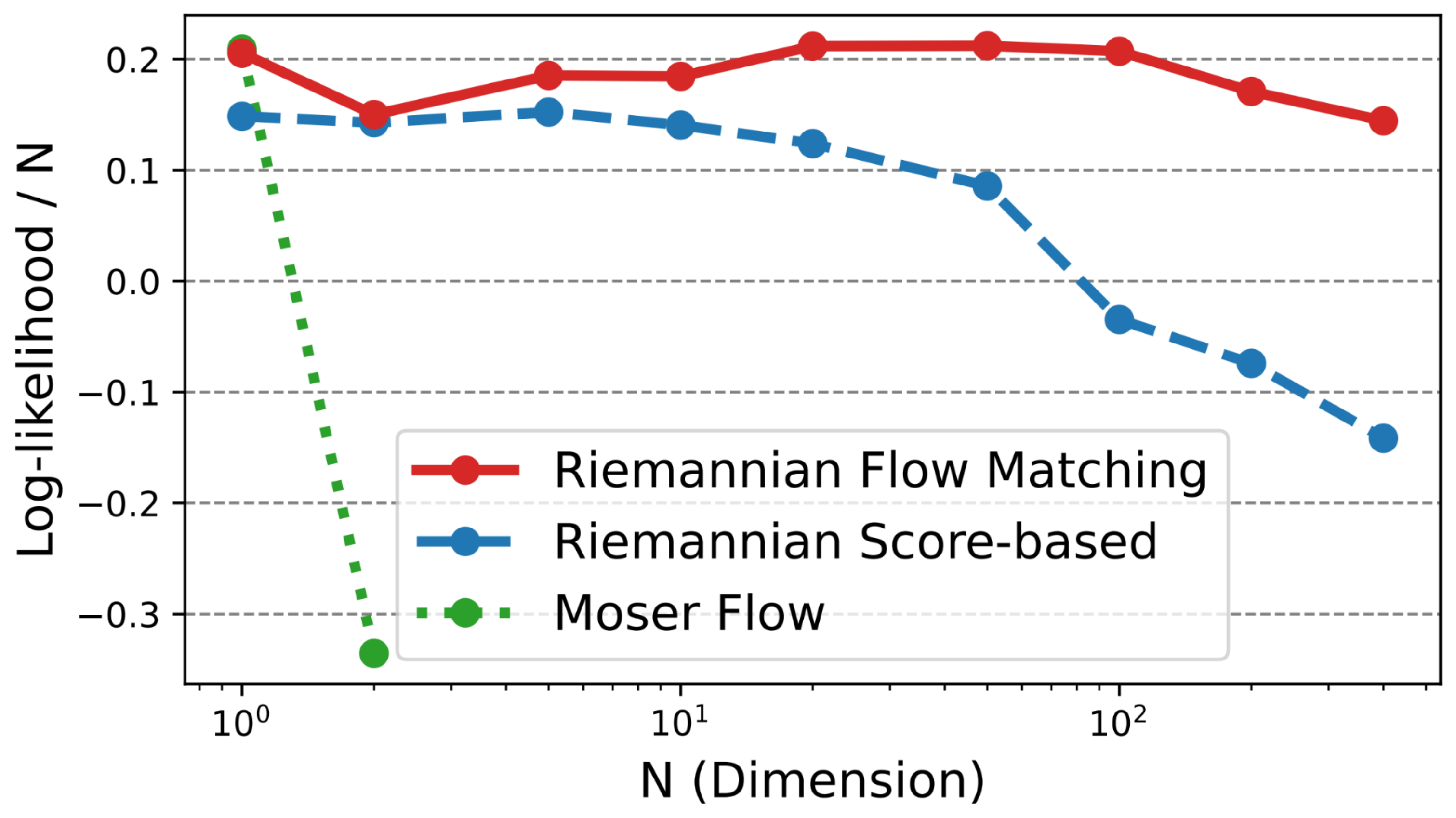
Require: base distribution $p(x_0)$, target $q(x_1)$
  Initialize parameters θ of v_t
  while not converged **do**
    sample time $t \sim \mathcal{U}(0, 1)$
    sample training example $x_1 \sim q(x_1)$
    sample noise $x_0 \sim p(x_0)$

    if simple geometry **then**
      $x_t = \texttt{exp}_{x_0}(\kappa(t)\texttt{log}_{x_0}(x_1))$
    else if general geometry **then**
      $x_t = \texttt{solve\_ODE}([0, t], x_0, u_t(x|x_1))$
    end if

    % closed-form regression target $u_t(x_t|x_1)$
    $\ell(\theta) = \|v_t(x_t; \theta) - u_t(x_t|x_1)\|_g^2$

    $\theta = \texttt{optimizer\_step}(\ell(\theta))$
  end while

RFM scales better

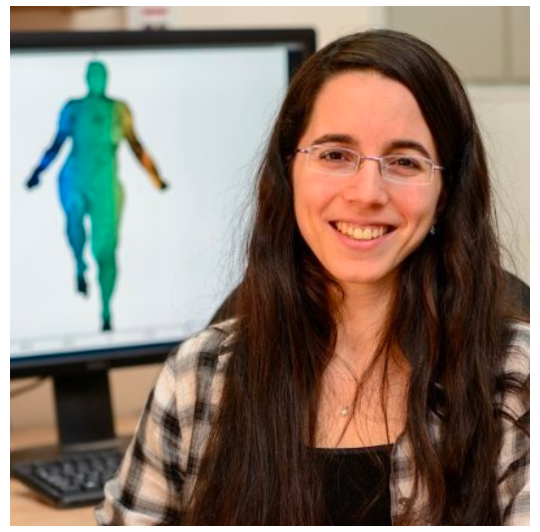


Conclusions

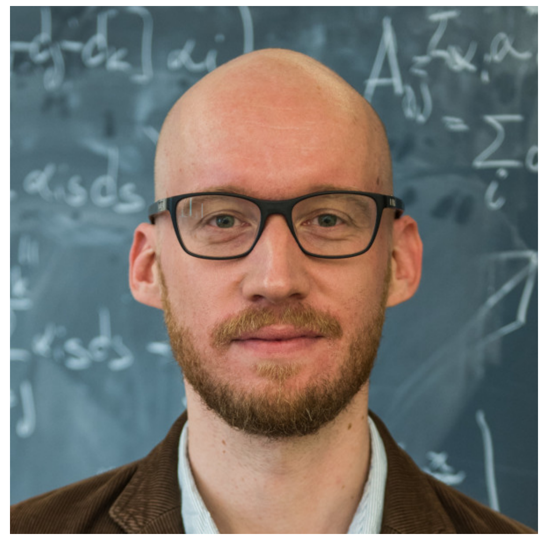
- Flow Matching is a **flexible** framework for training **generative flows**
- Simplifies and generalize Diffusion Models
- So far: improved sampling speed, stability, general source, noise at finite time, performance and scaling on non-euclidean data

Thanks!


Ricky T. Q. Chen


Heli Ben-Hamu


Neta Shaul


Maximilian Nickel


Matt Le


Aram-Alexandre Pooladian


Carles Domingo-Enrich


Brandon Amos

Meta

מכון ויצמן למדע
WEIZMANN INSTITUTE OF SCIENCE