Stochastic Control for Large Scale Reward-Driven Generative Modeling

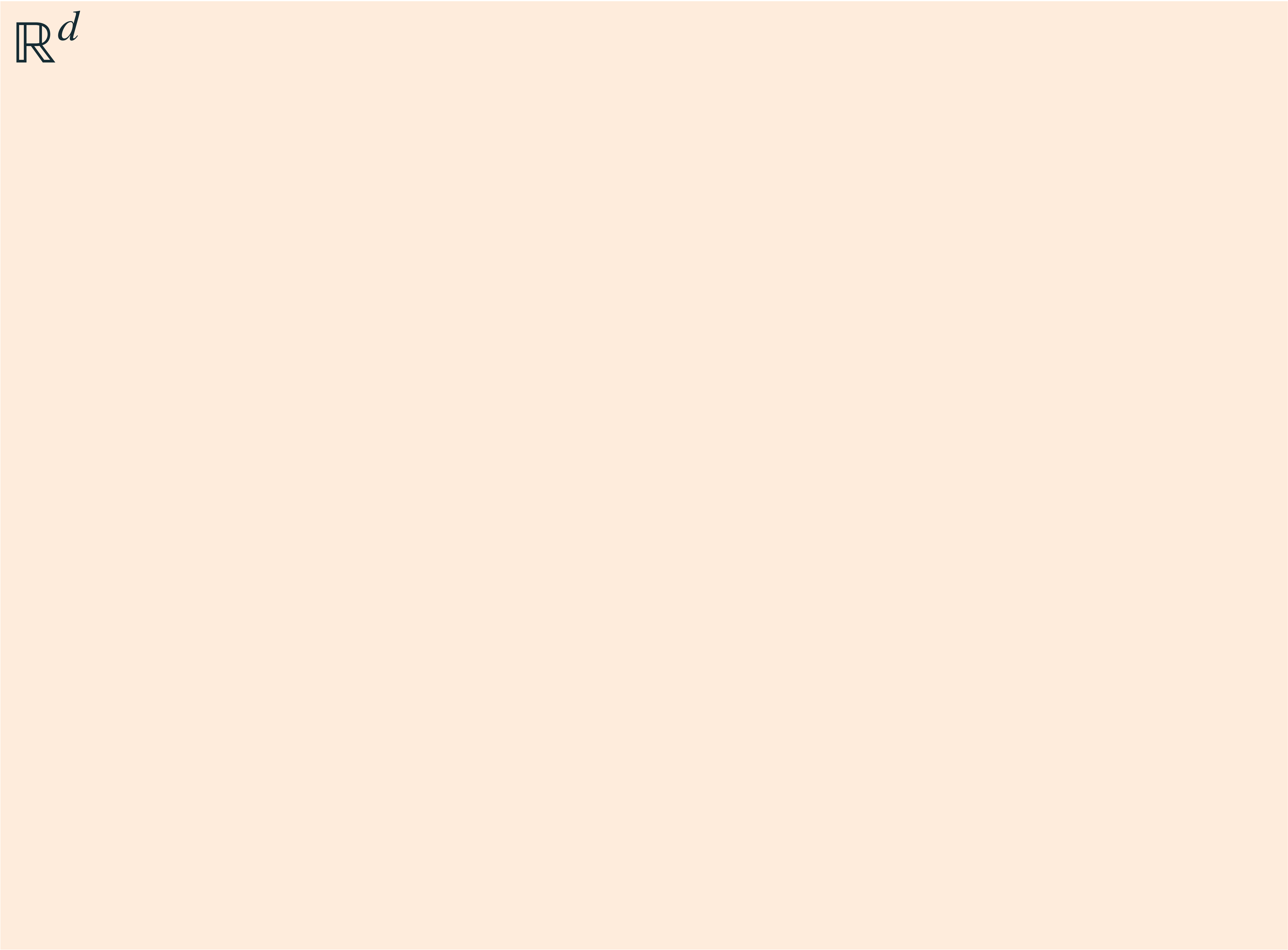
Link to slides:

Ricky T. Q. Chen

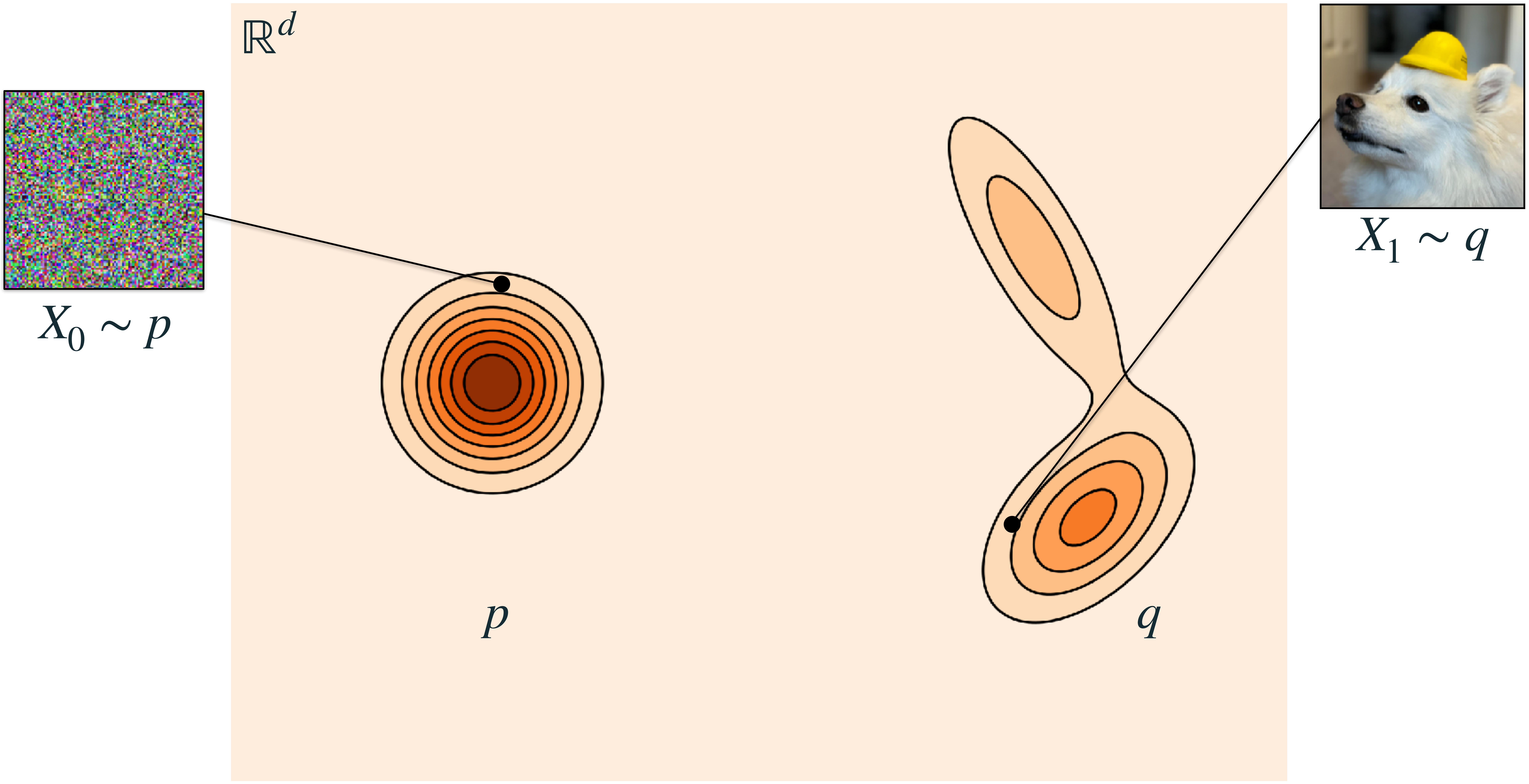
Meta

@ ICLR25 Workshop on Deep Generative Model in Machine Learning: Theory, Principle and Efficacy

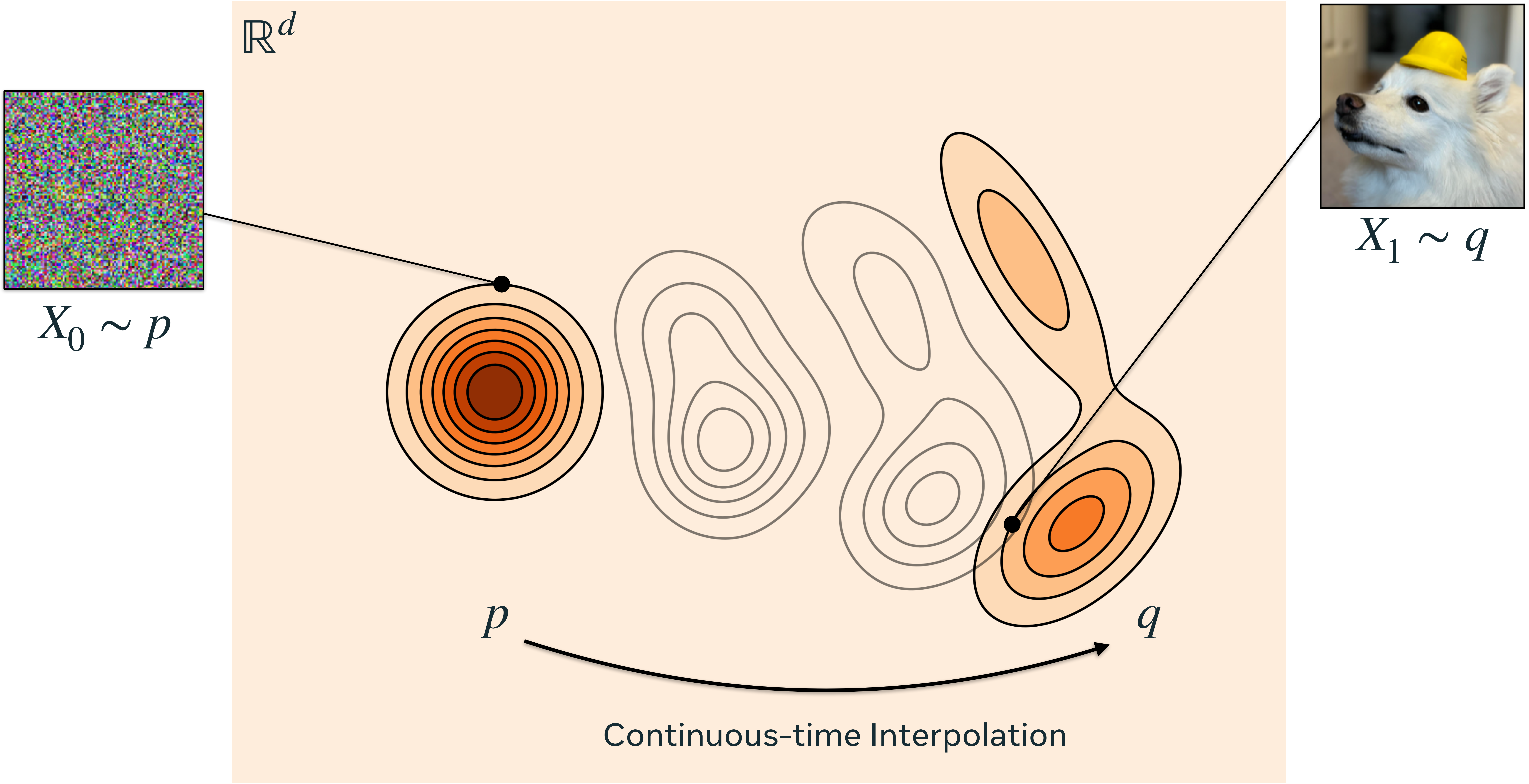
The Generative Modeling problem



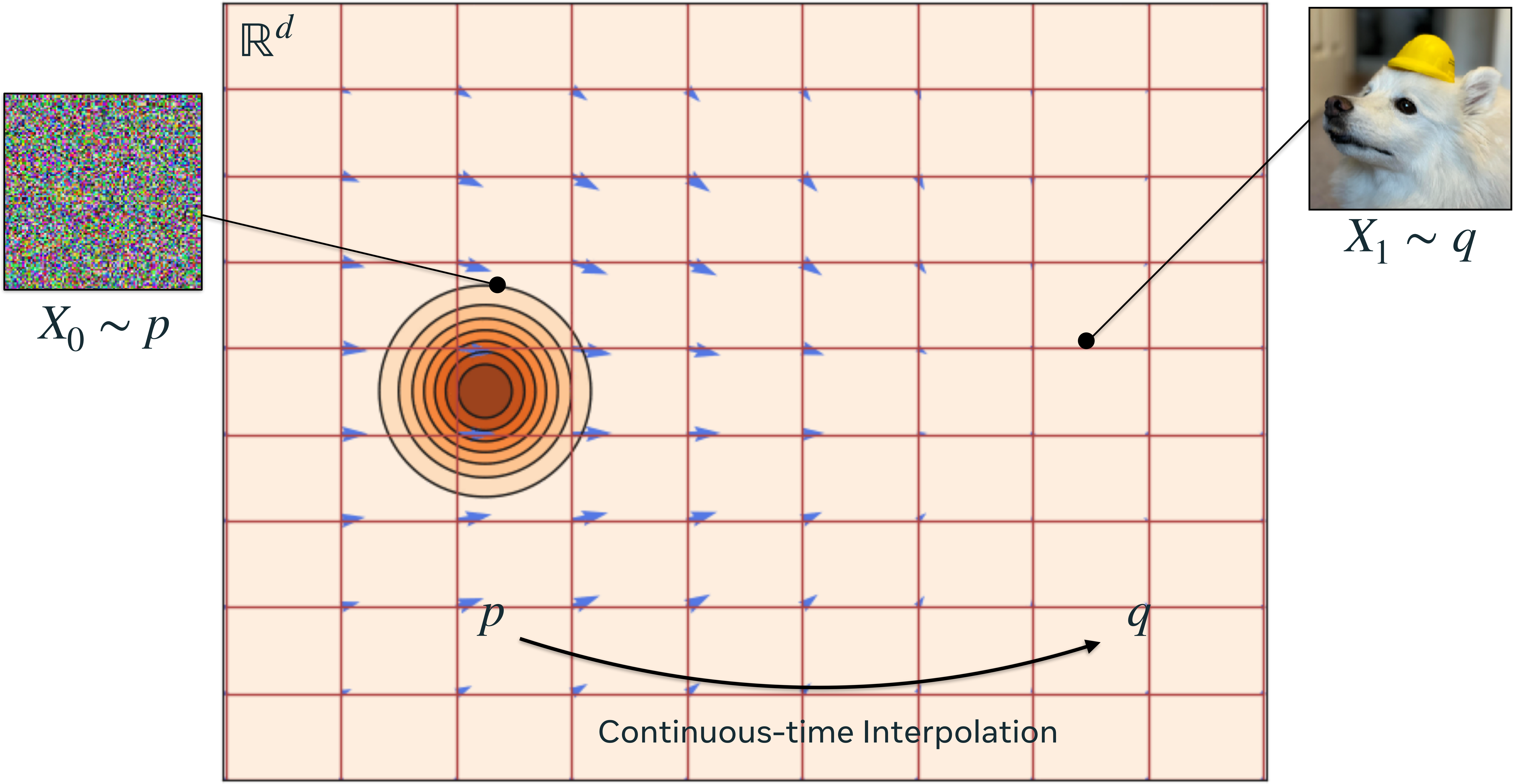
The Generative Modeling problem



The Generative Modeling problem



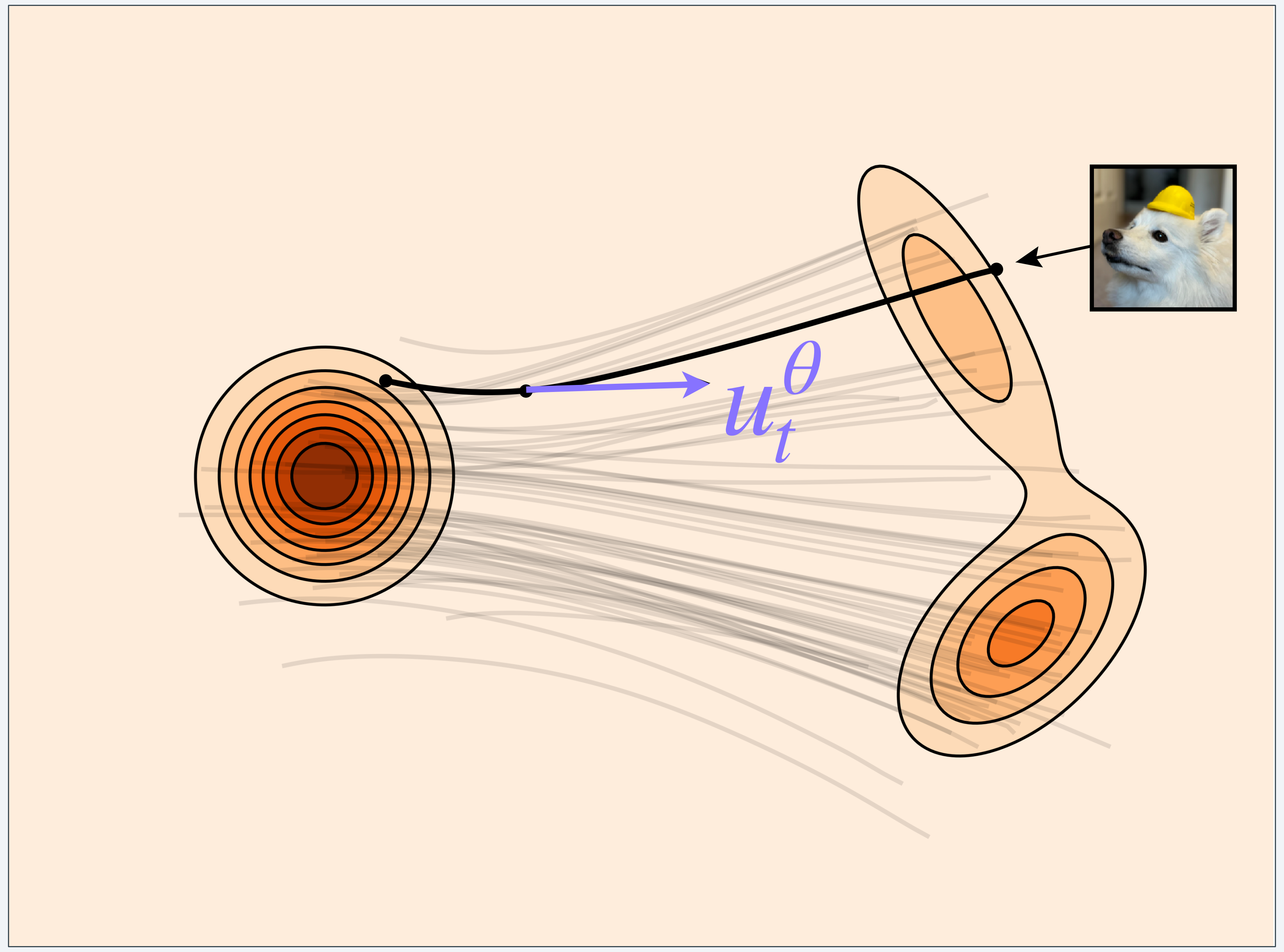
The Generative Modeling problem



A family of continuous-time generative models

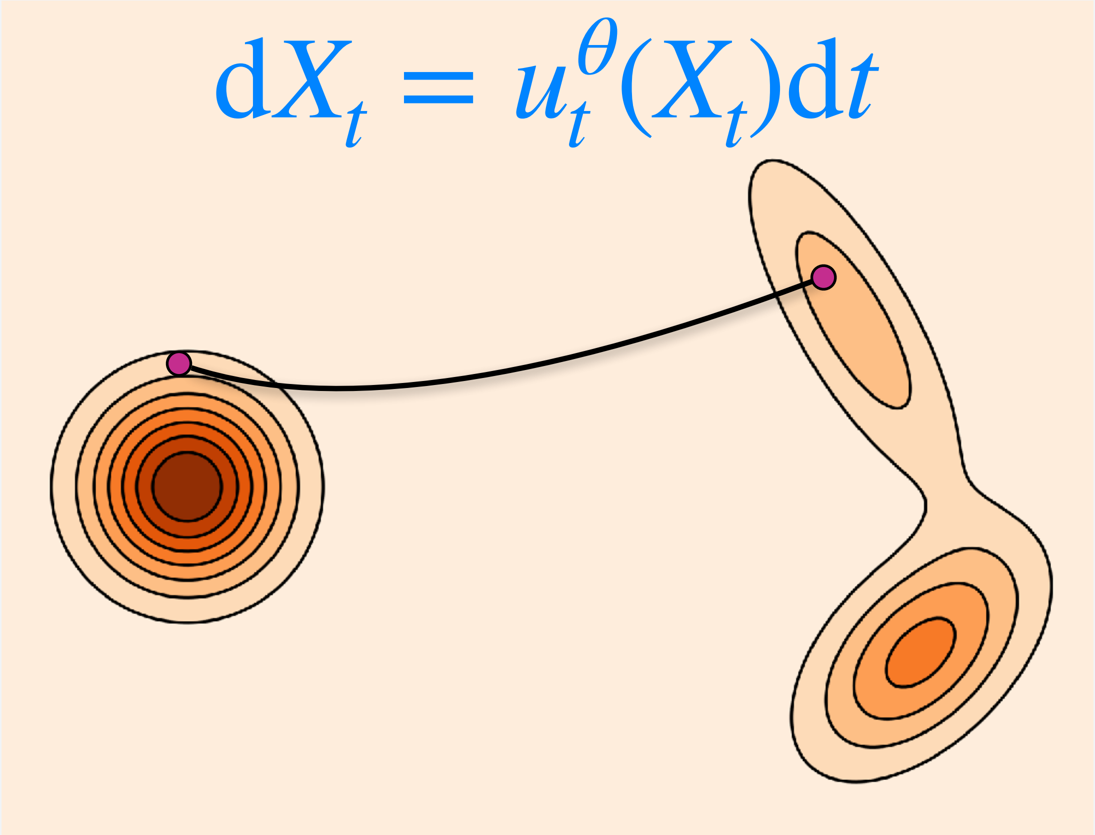
Parameterization

A vector field u_t^θ that points towards the data

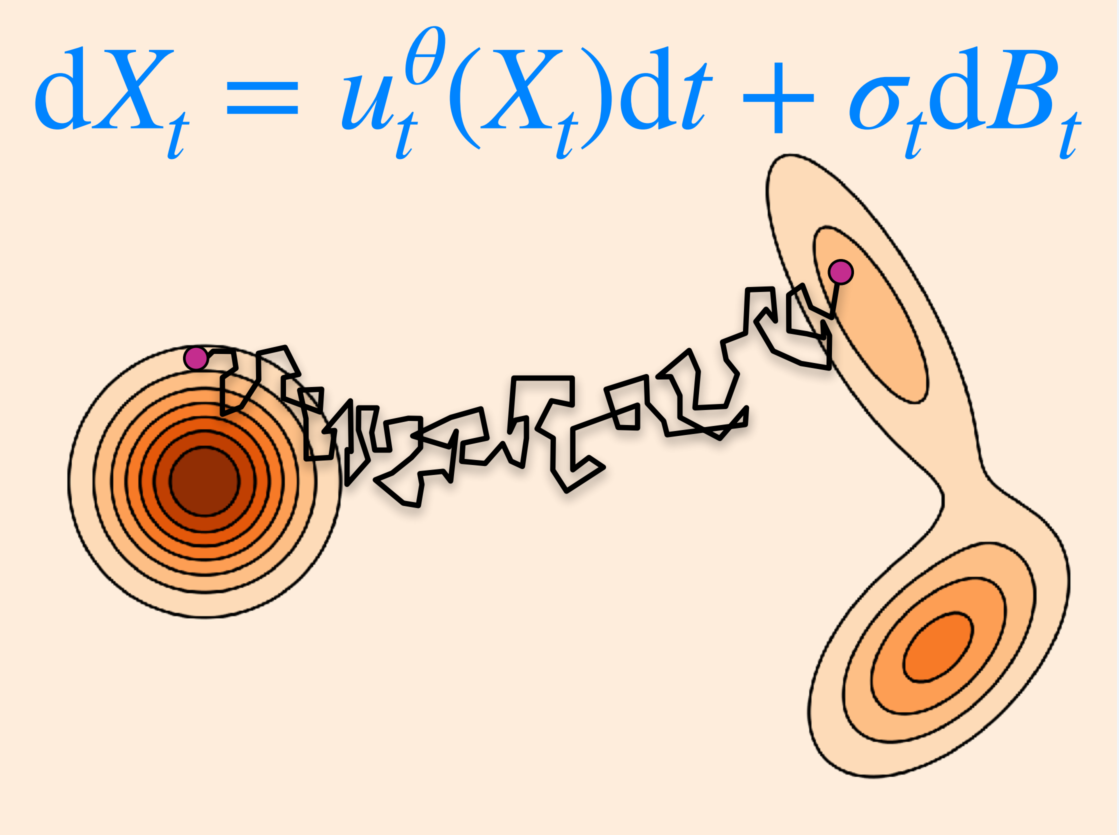


Sampling

A diffusion process X_t with (optional) diffusion coefficient σ_t



Ordinary Differential Equation



Stochastic Differential Equation

Data-driven vs. reward-driven learning problems

Data-driven

- fit to data
- simple & scalable

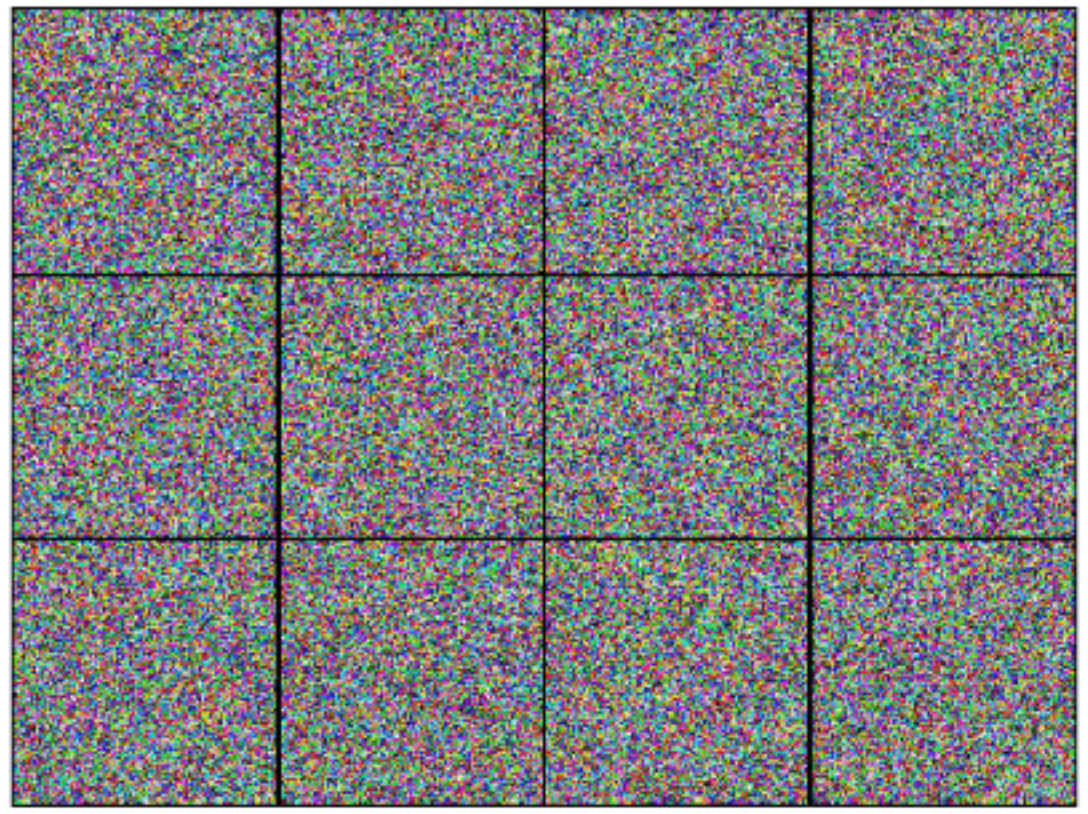


Image Generation

Meta MovieGen

Data-driven vs. reward-driven learning problems

Data-driven

- fit to data
- simple & scalable

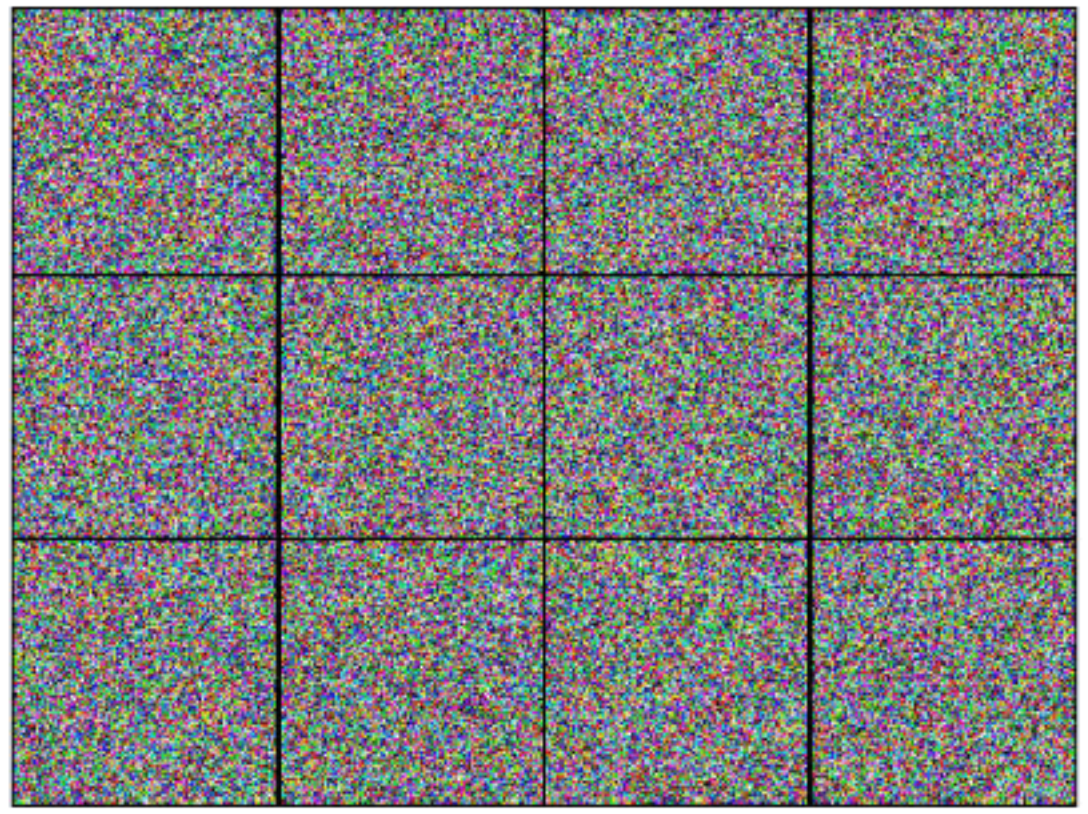


Image Generation

Meta MovieGen

Reward-driven

- maximize reward
- no data available

Reward Fine-tuning

Low Energy Generation

I. How to formulate the problem?

II. How to solve the problem?

III. How to scale the method?

I. How to formulate the problem?

Stochastic Optimal Control formulations for reward-driven generative modeling

Reward-driven generative modeling

Basic setup

Sampling from unnormalized distribution

$$p^*(x) \propto \exp\{-E(x)\} = \exp\{r(x)\}$$

(differentiable) energy model or (differentiable) reward model

More generally,

Sampling from tilted distribution

$$p^*(x) \propto p^{\text{base}}(x)\exp\{r(x)\}$$

(sample-able) base generative model (differentiable) reward model

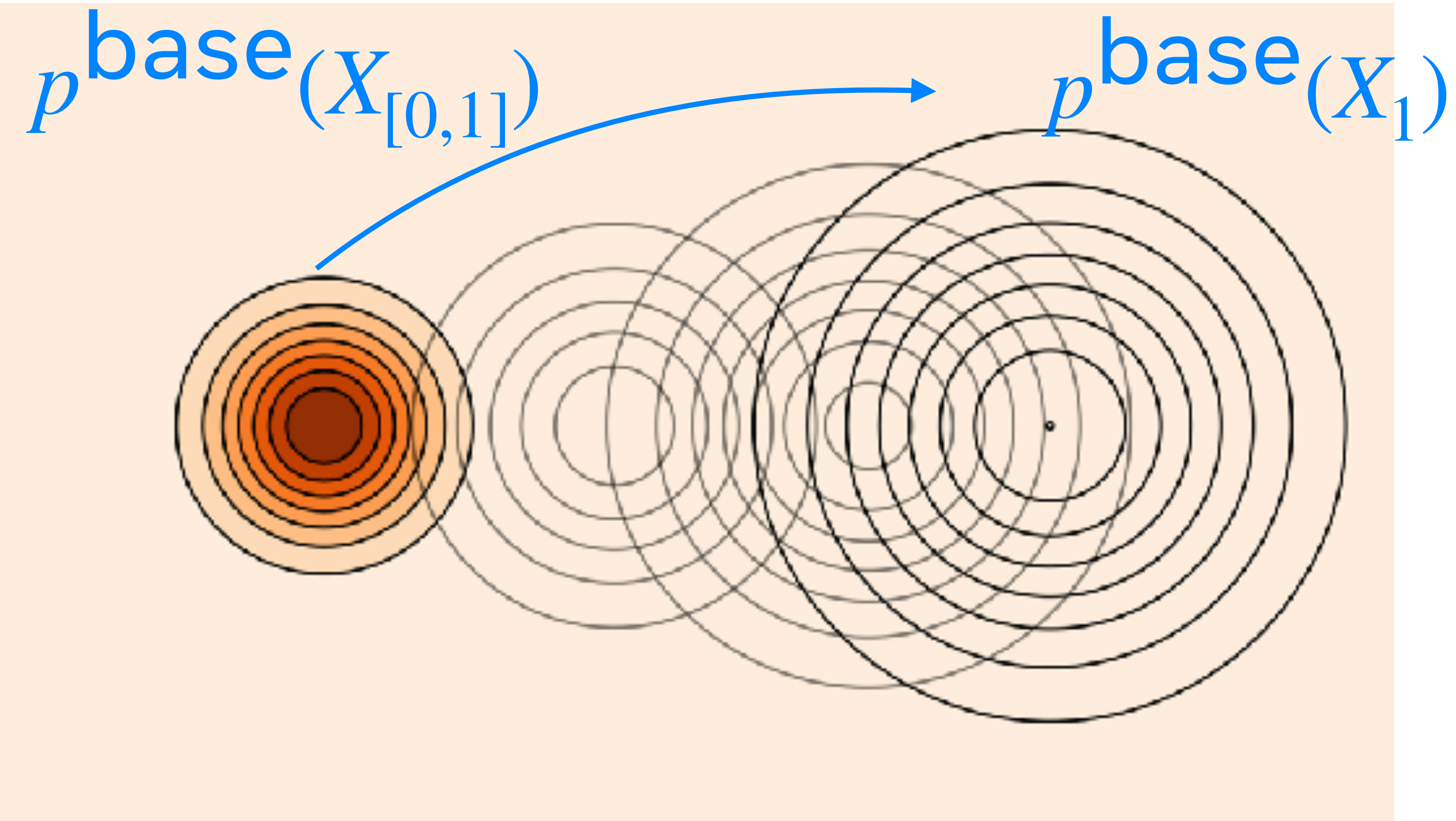
SDEs as generative models

Target distribution

$$p^*(x) \propto p^{\text{base}}(x)\exp\{r(x)\}$$

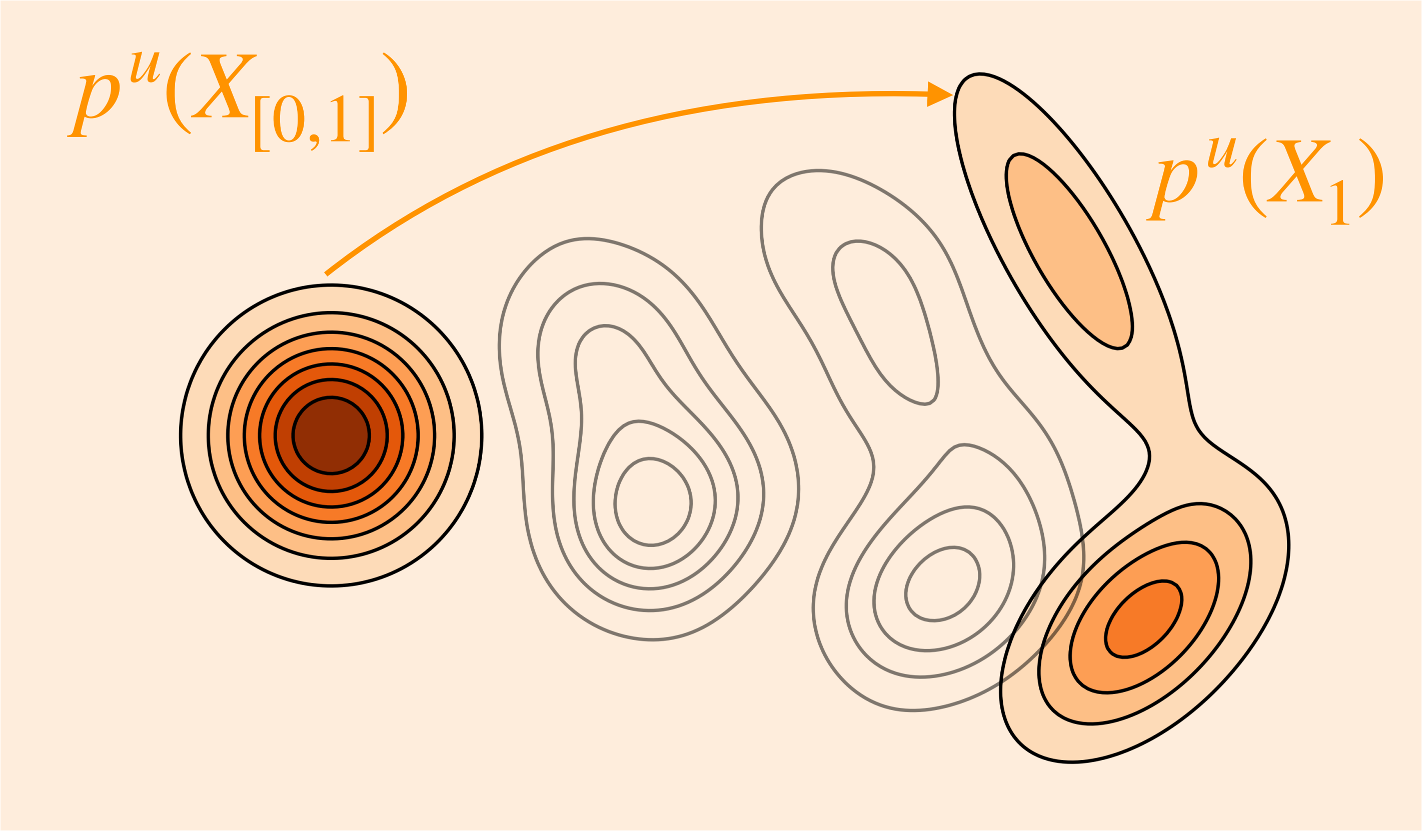
Base generative process

$$\mathrm{d}X_t = b_t(X_t)\mathrm{d}t + \sigma_t \mathrm{d}B_t \qquad X_0 \sim p$$



Controlled generative process

$$\mathrm{d}X_t = [b_t(X_t) + \sigma_t u_t^\theta(X_t)]\mathrm{d}t + \sigma_t \mathrm{d}B_t$$



KL divergence and path measures

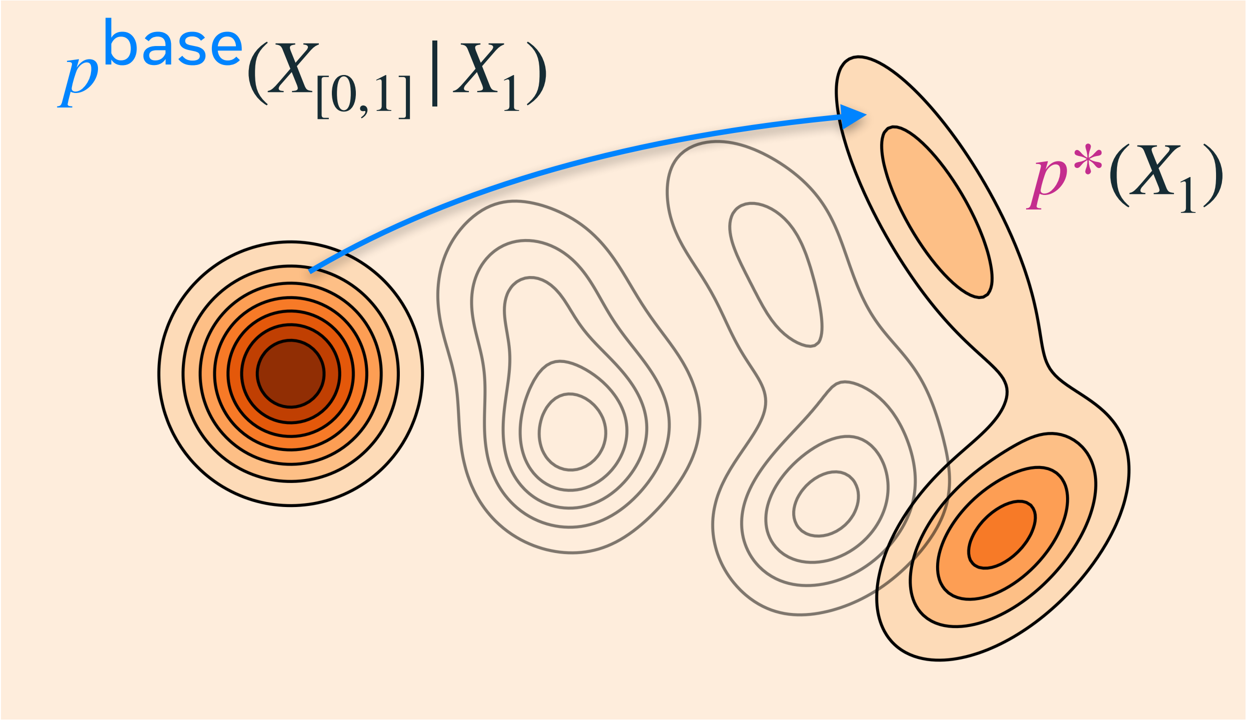
Target distribution

$$p^*(x) \propto p^{\text{base}}(x)\exp\{r(x)\}$$

$$\Uparrow$$

Extend to stochastic process

$$p^*(X_{[0,1]}) \triangleq p^{\text{base}}(X_{[0,1]} \mid X_1)\, p^*(X_1)$$



Base generative process

$$\mathrm{d}X_t = b_t(X_t)\mathrm{d}t + \sigma_t \mathrm{d}B_t \qquad X_0 \sim p$$

Controlled generative process

$$\mathrm{d}X_t = [b_t(X_t) + \sigma_t u_t^\theta(X_t)]\mathrm{d}t + \sigma_t \mathrm{d}B_t$$

KL divergence over stochastic process

$$D_{\text{KL}}(p^u(X_1) \| p^*(X_1))$$

$$\leq \boxed{D_{\text{KL}}(p^u(X_{[0,1]}) \| p^*(X_{[0,1]}))}$$

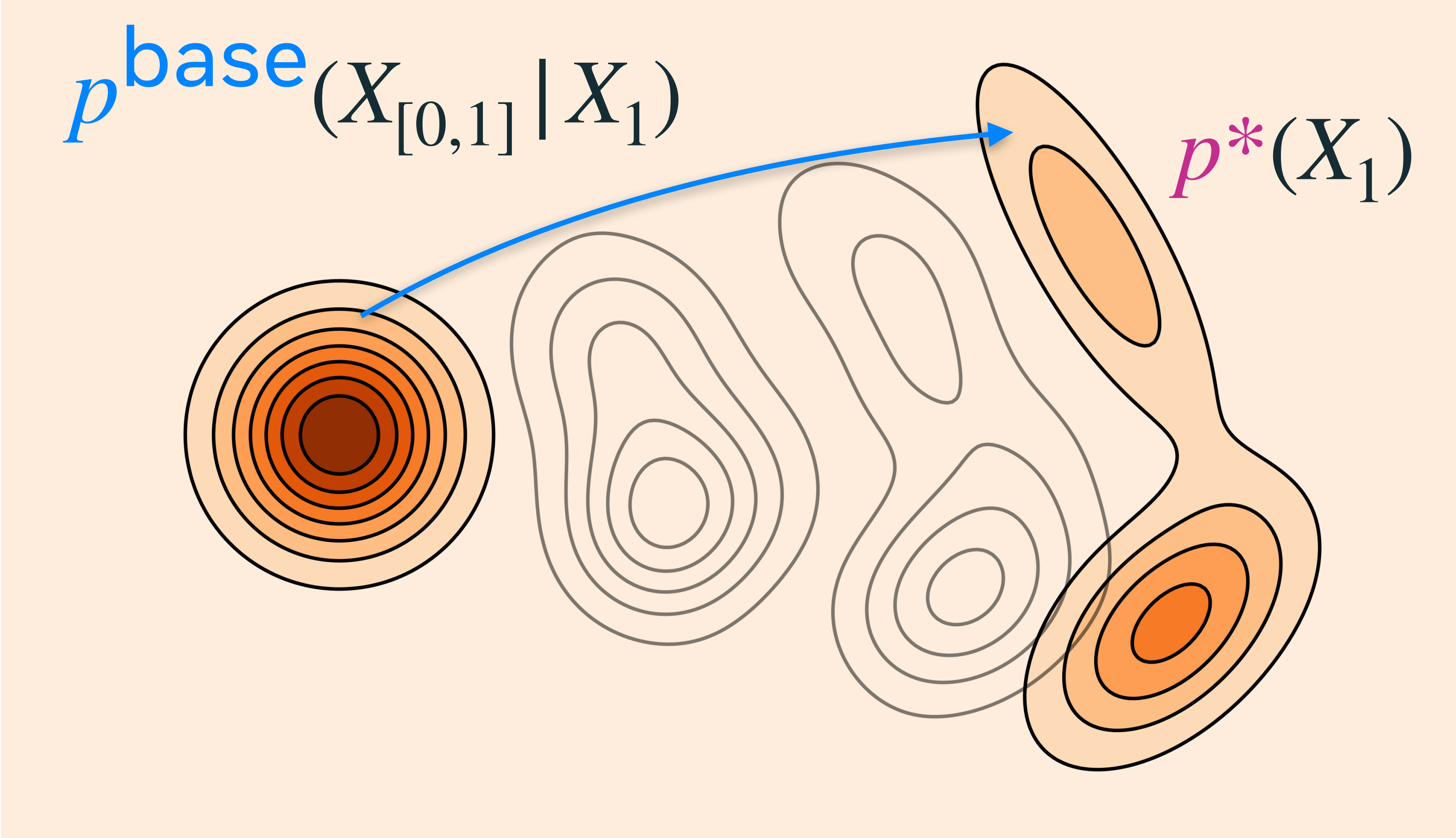
KL divergence and path measures

Target distribution

$$p^*(x) \propto p^{\text{base}}(x)\exp\{r(x)\}$$

KL objective

$$D_{\text{KL}}(p^u(X_{[0,1]}) \| p^*(X_{[0,1]}))$$



KL objective

$$D_{\text{KL}}(p^u(X_{[0,1]}) \| p^{\text{base}}(X_{[0,1]})) - \mathbb{E}_{p^u}[r(X_1)]$$

Theoretical guarantees for sampling and inference in generative models with latent diffusions. (Tzen & Raginsky 2019)
Path integral sampler: A stochastic control approach for sampling. (Zhang & Chen 2022)

Memoryless base processes

KL objective

$$D_{\mathrm{KL}}(p^u(X_{[0,1]}) \| p^{\mathrm{base}}(X_{[0,1]})) - \mathbb{E}_{p^u}[r(X_1)]$$

$$\Downarrow$$

Stochastic control objective

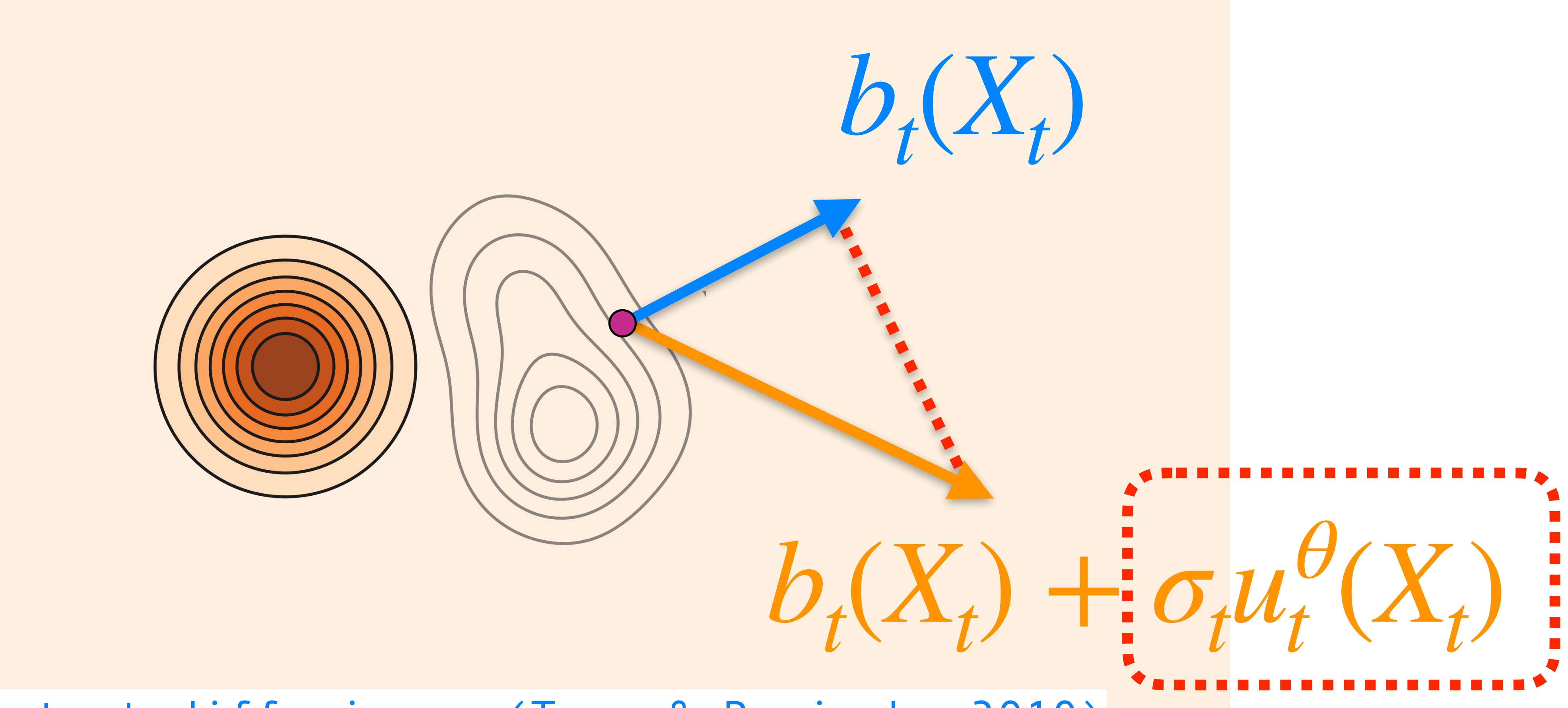
$$\cancel{D_{\mathrm{KL}}(p^u(X_0) \| p^{\mathrm{base}}(X_0))} + \mathbb{E}_{p^u}\left[\int_0^1 \frac{1}{2}\|u_t^\theta(X_t)\|^2 \mathrm{d}t - r(X_1)\right]$$

KL at t=0

Conditional KL

Don’t need to optimize $p^u(X_0)$ if the base process is “**memoryless**”:

$$p^{\mathrm{base}}(X_0, X_1) = p^{\mathrm{base}}(X_0)p^{\mathrm{base}}(X_1)$$



Theoretical guarantees for sampling and inference in generative models with latent diffusions. (Tzen & Raginsky 2019)
Path integral sampler: A stochastic control approach for sampling. (Zhang & Chen 2022)

Summary of the SOC formulation

Target distribution

$$p^*(x) \propto p^{\text{base}}(x)\exp\{r(x)\}$$

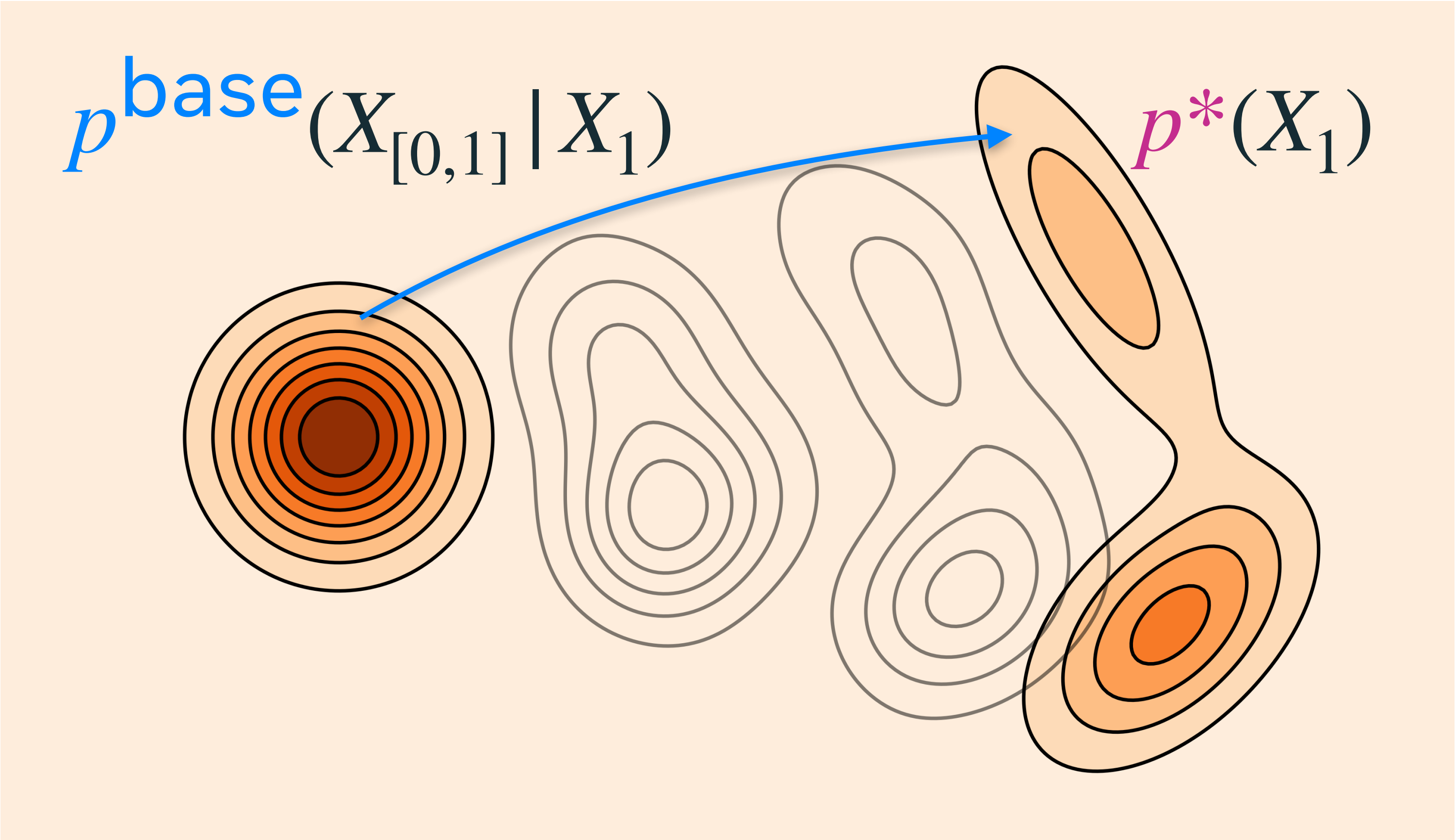
Base generative process

$$\mathrm{d}X_t = b_t(X_t)\mathrm{d}t + \sigma_t \mathrm{d}B_t \qquad X_0 \sim p$$

Controlled generative process

$$\mathrm{d}X_t = [b_t(X_t) + \sigma_t u_t^\theta(X_t)]\mathrm{d}t + \sigma_t \mathrm{d}B_t$$

Minimize KL to



Theoretical guarantees for sampling and inference in generative models with latent diffusions. (Tzen & Raginsky 2019)
Path integral sampler: A stochastic control approach for sampling. (Zhang & Chen 2022)

Summary of the SOC formulation

Target distribution

$$p^*(x) \propto p^{\text{base}}(x)\exp\{r(x)\}$$

Base generative process

$$\mathrm{d}X_t = b_t(X_t)\mathrm{d}t + \sigma_t \mathrm{d}B_t \qquad X_0 \sim p$$

Controlled generative process

$$\mathrm{d}X_t = [b_t(X_t) + \sigma_t u_t^\theta(X_t)]\mathrm{d}t + \sigma_t \mathrm{d}B_t$$

If the base process is "**memoryless**":

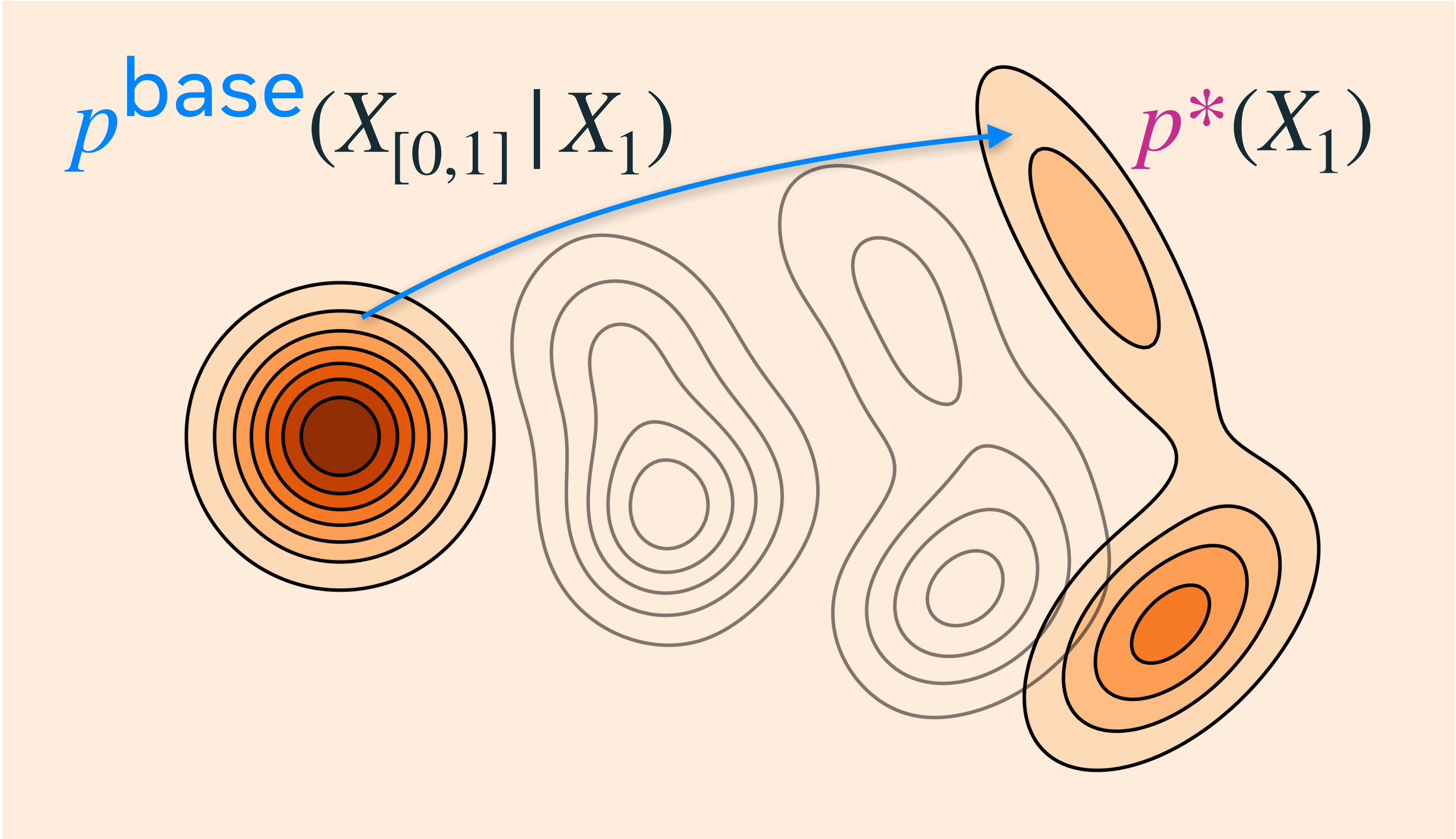
$$p^{\text{base}}(X_0, X_1) = p^{\text{base}}(X_0)p^{\text{base}}(X_1)$$

Minimizing KL is equivalent to optimizing:

Stochastic control objective

$$\boxed{\mathbb{E}_{p^u}\left[\int_0^1 \frac{1}{2}\|u_t^\theta(X_t)\|^2\mathrm{d}t - r(X_1)\right]}$$

Minimize KL to



Theoretical guarantees for sampling and inference in generative models with latent diffusions. (Tzen & Raginsky 2019)
Path integral sampler: A stochastic control approach for sampling. (Zhang & Chen 2022)

A memoryless noise schedule for Flow Matching

<u>If</u> the base process is "**memoryless**":

$$p^{\text{base}}(X_0, X_1) = p^{\text{base}}(X_0)p^{\text{base}}(X_1)$$

Flow Matching Base Model ?

Reward Fine-tuned Model

Adjoint Matching: Fine-tuning Flow and Diffusion Generative Models with Memoryless Stochastic Optimal Control (Domingo et al. 2024)

A memoryless noise schedule for Flow Matching

If the base process is "**memoryless**":

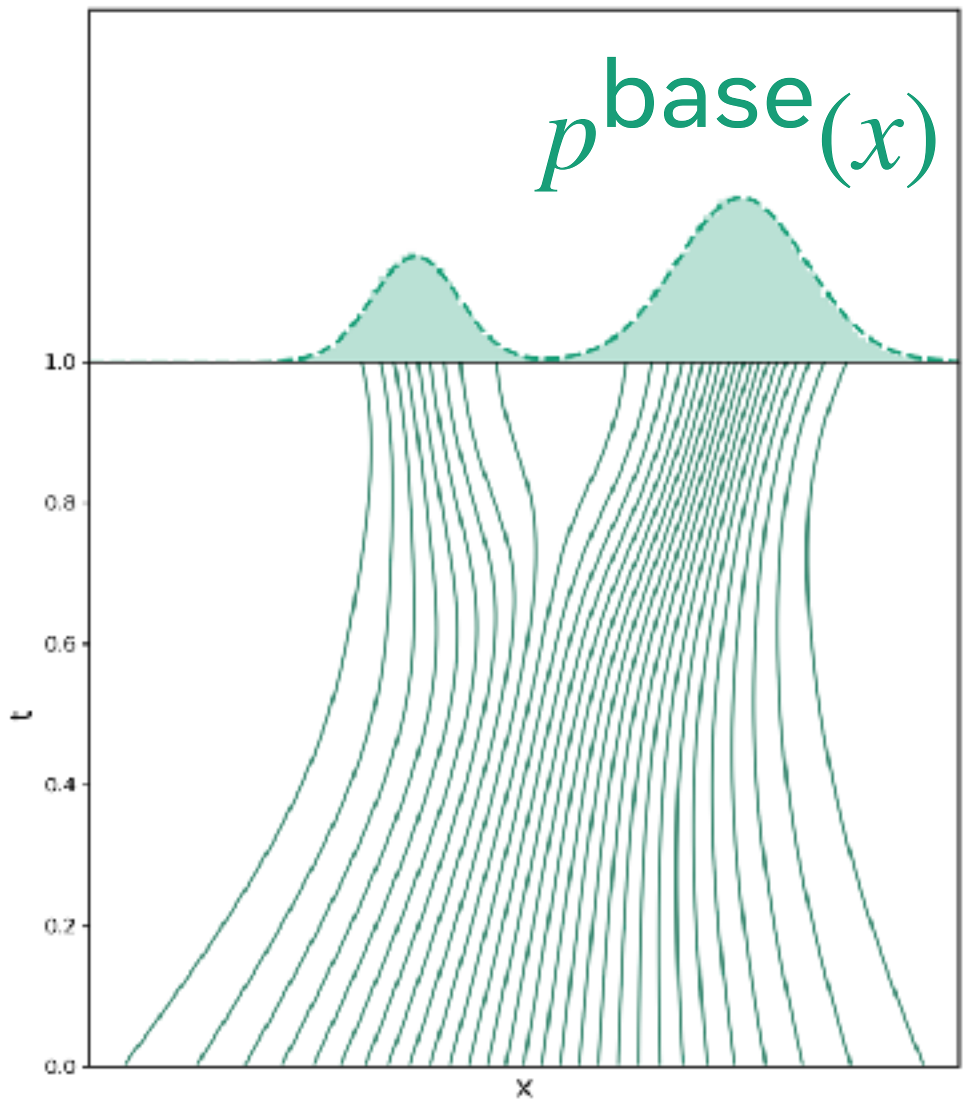
$$p^{\text{base}}(X_0, X_1) = p^{\text{base}}(X_0)p^{\text{base}}(X_1)$$

Pretrained Flow Matching model with linear scheduler $X_t = (1-t)X_0 + tX_1$:

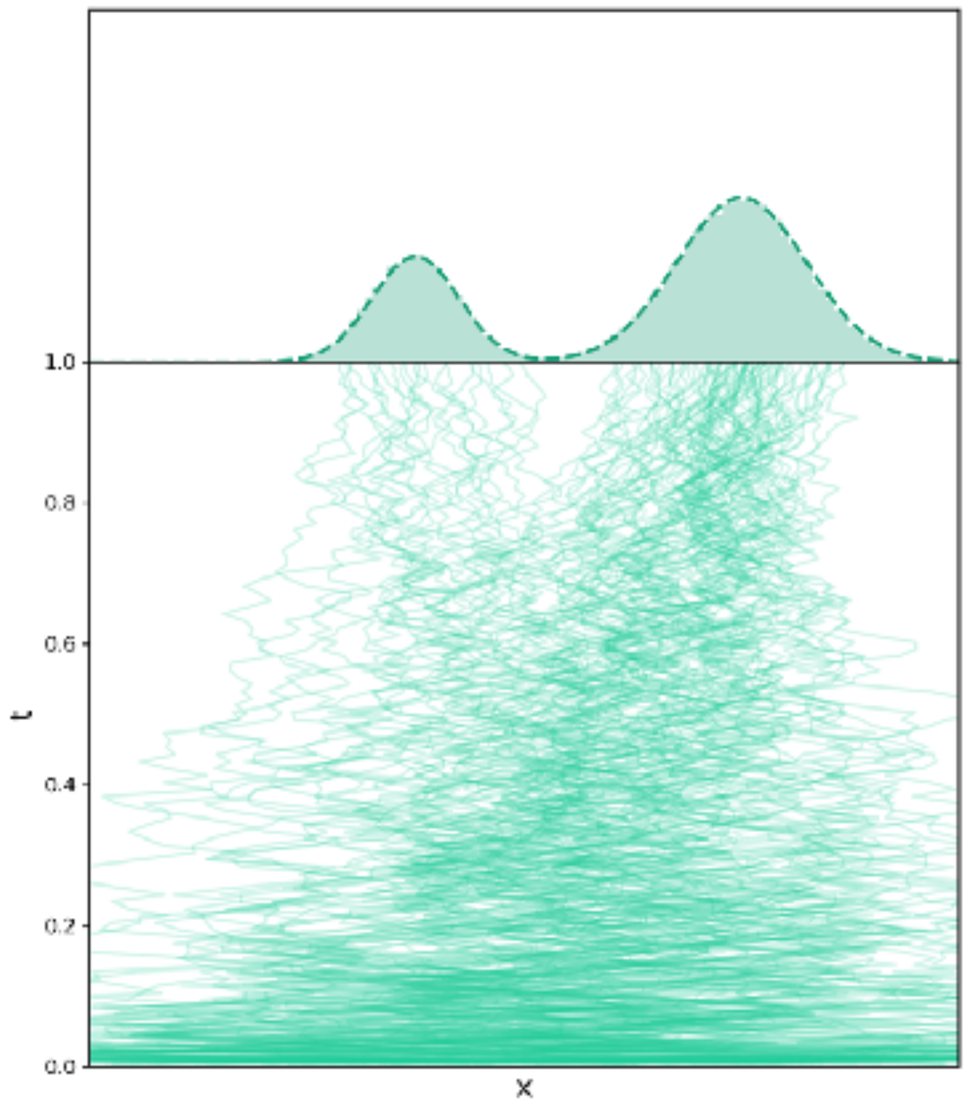
$$\mathrm{d}X_t = v_t(X_t)\mathrm{d}t$$

The **Memoryless** Flow Matching conversion is:

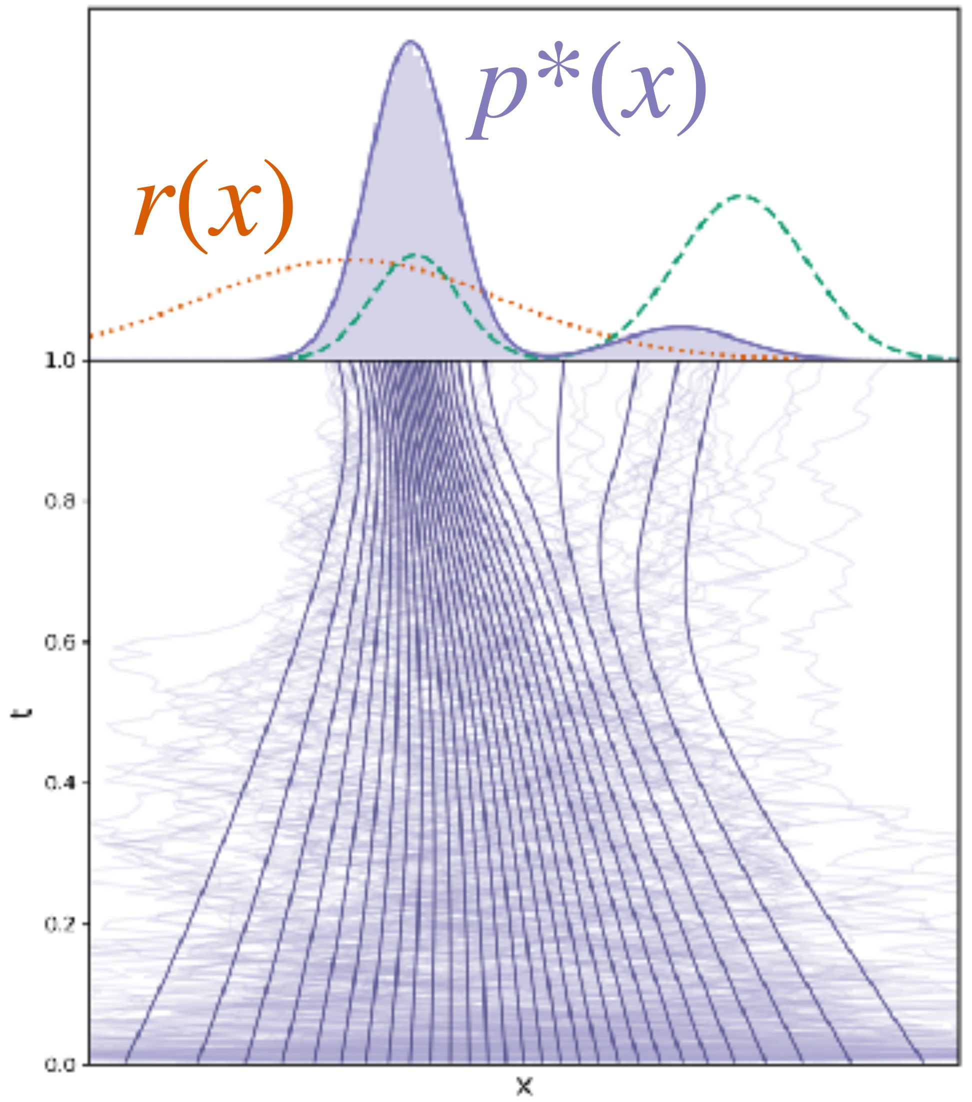
$$\mathrm{d}X_t = b_t(X_t)\mathrm{d}t + \sigma_t \mathrm{d}B_t \quad \begin{cases} b_t(X_t) = 2v_t(X_t) - \frac{1}{t}X_t \\ \sigma_t = \frac{1-t}{t} \end{cases}$$



Convert to Memoryless



Fine-tune with SOC



Adjoint Matching: Fine-tuning Flow and Diffusion Generative Models with Memoryless Stochastic Optimal Control (Domingo et al. 2024)

A memoryless noise schedule for Flow Matching

If the base process is "**memoryless**":

$$p^{\text{base}}(X_0, X_1) = p^{\text{base}}(X_0)p^{\text{base}}(X_1)$$

Pretrained Flow Matching model with linear scheduler $\boxed{X_t = (1-t)X_0 + tX_1:}$

$$\mathrm{d}X_t = v_t(X_t)\mathrm{d}t$$

The **Memoryless** Flow Matching conversion is:

$$\mathrm{d}X_t = b_t(X_t)\mathrm{d}t + \sigma_t \mathrm{d}B_t \quad \begin{cases} b_t(X_t) = 2v_t(X_t) - \frac{1}{t}X_t \\ \sigma_t = \frac{1-t}{t} \end{cases}$$

[!] Related to minimizing KL to the (memoryless) process used as training signal.

refs: DDPM (Ho et al. 2020), Scalable Interpolant Transformers (Ma et al. 2024)

II. How to solve the problem?

Adjoint Matching: a new approach for solving stochastic optimal control

The most classic approach: adjoint method

Stochastic control

$$\min_u \mathscr{L}(u) = \mathbb{E}_{p^u}\left[\int_0^1 \frac{1}{2}\|u_t^\theta(X_t)\|^2 \mathrm{d}t - r(X_1)\right]$$

s.t. $\mathrm{d}X_t = [b_t(X_t) + \sigma_t u_t^\theta(X_t)]\mathrm{d}t + \sigma_t \mathrm{d}B_t\,, \qquad X_0 \sim p$

The most classic approach: adjoint method

Stochastic control

$$\min_u \mathscr{L}(u) = \mathbb{E}_{p^u}\left[\int_0^1 \frac{1}{2}\|u_t^\theta(X_t)\|^2 \mathrm{d}t - r(X_1)\right]$$

s.t. $\mathrm{d}X_t = [b_t(X_t) + \sigma_t u_t^\theta(X_t)]\mathrm{d}t + \sigma_t \mathrm{d}B_t\,, \quad X_0 \sim p$

Adjoint state

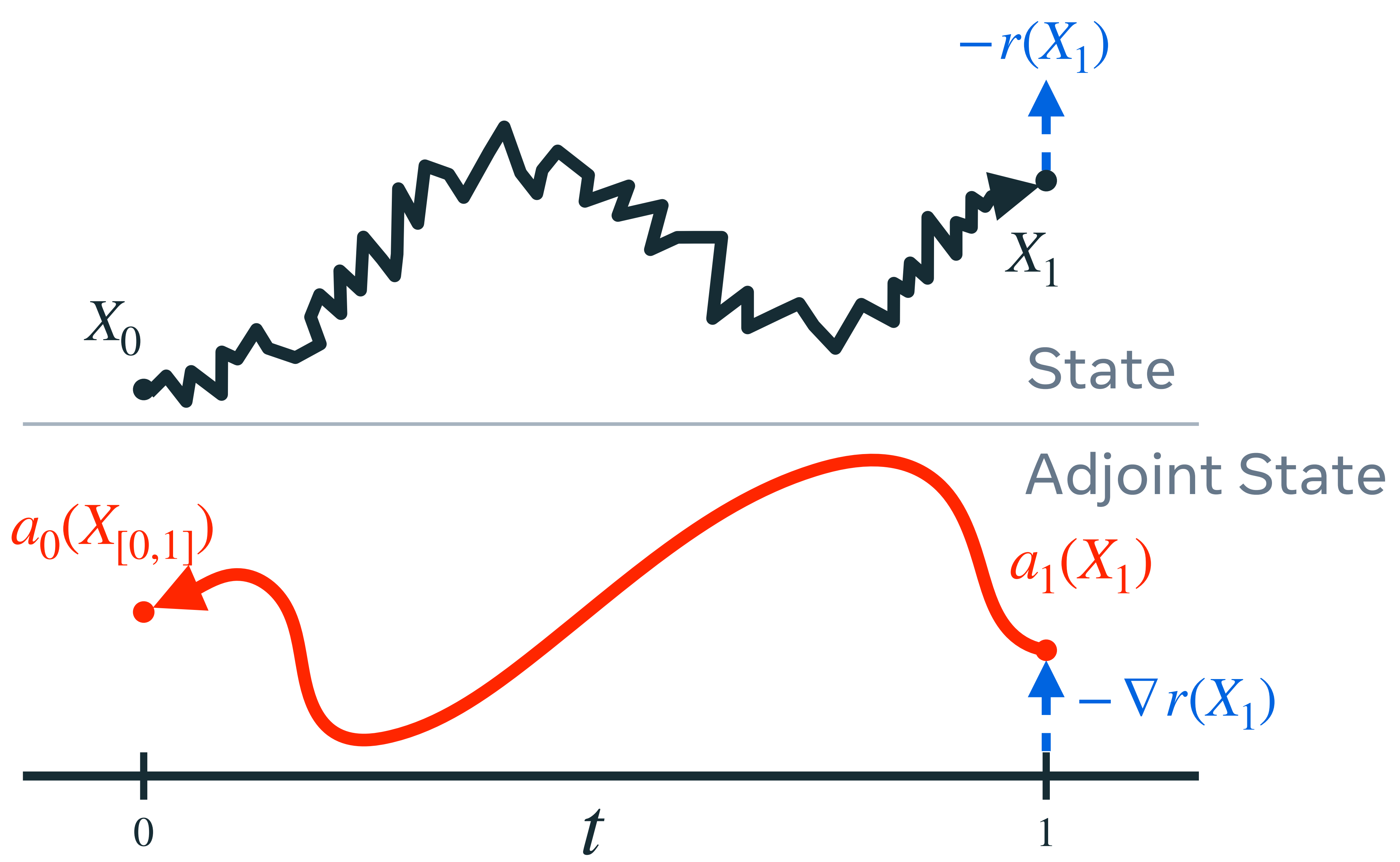
(a.k.a. the gradient w.r.t. state)

$$a_t(X_{[0,1]}) := \nabla_{X_t}\left[\int_t^1 \frac{1}{2}\|u_t^\theta(X_t)\|^2 \mathrm{d}t - r(X_1)\right]$$

Discretization

$$X_{t_{i+1}} = X_{t_i} + h\left[b_{t_i}(X_{t_i}) + \sigma_{t_i} u_{t_i}^\theta(X_{t_i})\right] + \sqrt{h}\sigma_t \varepsilon$$

$$a_{t_i} = a_{t_{i+1}}^\top \frac{\partial X_{t_{i+1}}}{\partial X_{t_i}} = a_{t_{i+1}} + h a_{t_{i+1}}^\top \nabla_{X_{t_i}}\left[b_{t_i}(X_{t_i}) + \sigma_t u_{t_i}^\theta(X_{t_i})\right]$$



The most classic approach: adjoint method

Stochastic control

$$\min_u \mathscr{L}(u) = \mathbb{E}_{p^u}\left[\int_0^1 \frac{1}{2}\|u_t^\theta(X_t)\|^2 \mathrm{d}t - r(X_1)\right]$$

s.t. $\mathrm{d}X_t = [b_t(X_t)+\sigma_t u_t^\theta(X_t)]\mathrm{d}t + \sigma_t \mathrm{d}B_t\,, \qquad X_0 \sim p$

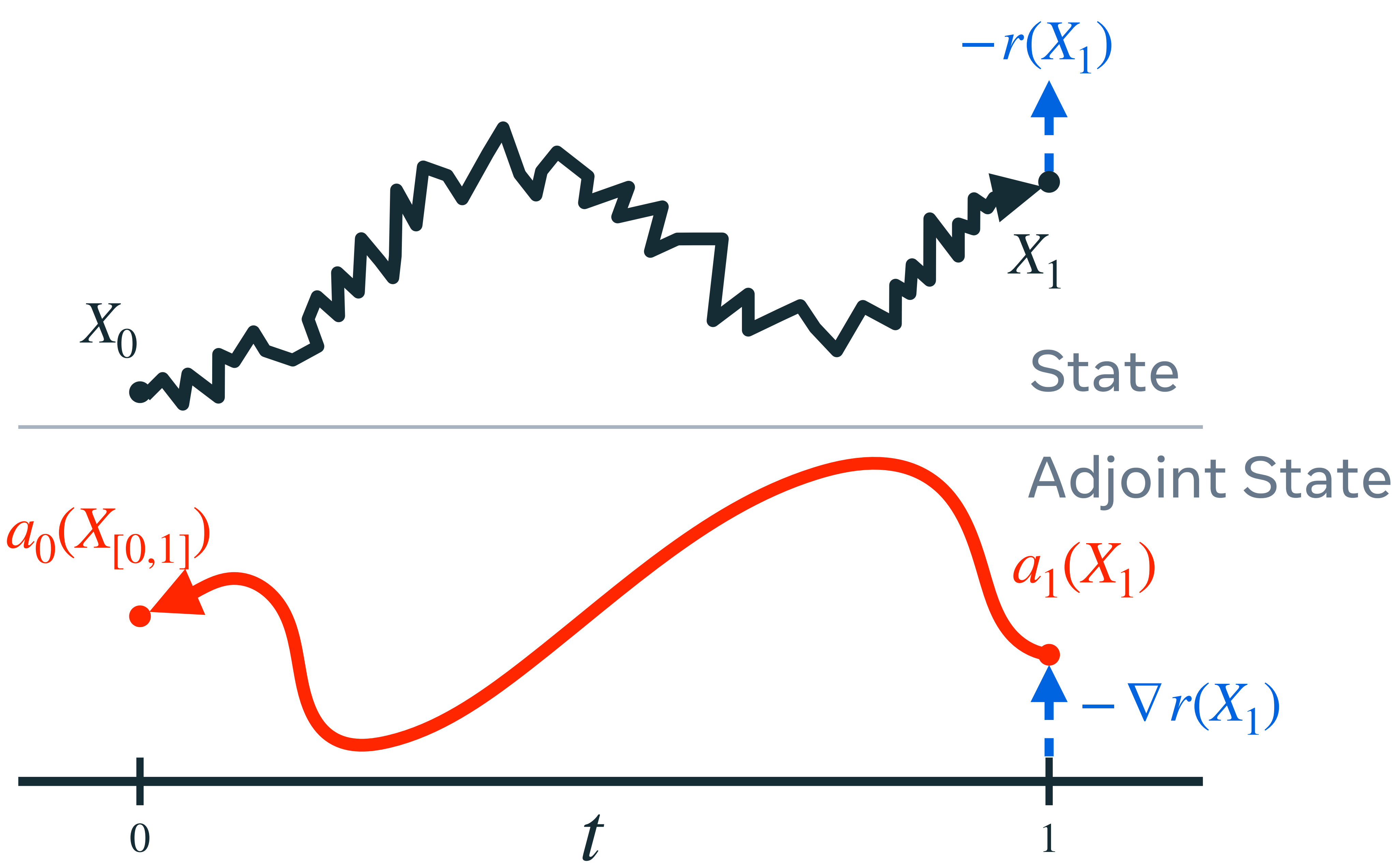
Adjoint state

(a.k.a. the gradient w.r.t. state)

$$a_t(X_{[0,1]}) := \nabla_{X_t}\left[\int_t^1 \frac{1}{2}\|u_t^\theta(X_t)\|^2 \mathrm{d}t - r(X_1)\right]$$

$$\frac{\mathrm{d}}{\mathrm{d}t}a_t(X_{[0,1]}) = -a_t(X_{[0,1]})^\top \nabla_{X_t}\left(b_t(X_t) + \sigma_t u_t^\theta(X_t)\right)$$

$$a_1(X_{[0,1]}) = -\nabla r(X_1)$$



The most classic approach: adjoint method

Stochastic control

$$\min_u \mathscr{L}(u) = \mathbb{E}_{p^u}\left[\int_0^1 \frac{1}{2}\|u_t^\theta(X_t)\|^2 \mathrm{d}t - r(X_1)\right]$$

s.t. $\mathrm{d}X_t = [b_t(X_t) + \sigma_t u_t^\theta(X_t)]\mathrm{d}t + \sigma_t \mathrm{d}B_t\,, \qquad X_0 \sim p$

Adjoint state

(a.k.a. the gradient w.r.t. state)

$$a_t(X_{[0,1]}) := \nabla_{X_t}\left[\int_t^1 \frac{1}{2}\|u_t^\theta(X_t)\|^2 \mathrm{d}t - r(X_1)\right]$$

$$\frac{\mathrm{d}}{\mathrm{d}t} a_t(X_{[0,1]}) = -a_t(X_{[0,1]})^\top \nabla_{X_t}\left(b_t(X_t) + \sigma_t u_t^\theta(X_t)\right) - \nabla_{X_t}\left(\frac{1}{2}\|u_t^\theta(X_t)\|^2\right)$$

$$a_1(X_{[0,1]}) = -\nabla r(X_1)$$

Aggregate

$$\frac{\mathrm{d}\mathscr{L}}{\mathrm{d}\theta} = \int_0^1 \frac{\partial u_t^\theta(X_t)}{\partial \theta}^\top \sigma_t\, a_t(X_{[0,1]}) + \frac{\partial}{\partial \theta}\frac{1}{2}\|u_t^\theta(X_t)\|^2 \mathrm{d}t$$

An alternative perspective: fixed point method

Cost functional

(a.k.a. expected future cost)

$$J(u; t, x) := \mathbb{E}_{p^u}\left[\int_t^1 \frac{1}{2} \|u_t^\theta(X_t)\|^2 \mathrm{d}t - r(X_1) \,|\, X_t = x \right]$$

Optimality criterion

(a.k.a. steepest descent)

$$u_t^*(x) = -\sigma_t \nabla_x J(u^*; t, x)$$

Relation to adjoint state

$$\nabla_x J(u^*; t, x) = \mathbb{E}_{p^u}\left[a_t(X_{[0,1]}) \,|\, X_t = x \right]$$

"Basic" Adjoint Matching

$$\min_u \mathcal{L}(u) = \int_0^1 \|u_t^\theta(X_t) + \sigma_t \nabla_x J(u^*; t, X_t)\|^2 \mathrm{d}t$$

$$X_{[0,1]} \sim p^{\bar{u}}$$
$$\bar{u} = \mathrm{stopgrad}(u)$$

An alternative perspective: fixed point method

Cost functional

(a.k.a. expected future cost)

$$J(u;t,x) := \mathbb{E}_{p^u}\left[\int_t^1 \frac{1}{2}\|u_t^\theta(X_t)\|^2 \mathrm{d}t - r(X_1) \,|\, X_t = x\right]$$

Optimality criterion

(a.k.a. steepest descent)

$$u_t^*(x) = -\,\sigma_t \nabla_x J(u^*;t,x)$$

Relation to adjoint state

$$\nabla_x J(u^*;t,x) = \mathbb{E}_{p^u}\left[a_t(X_{[0,1]}) \,|\, X_t = x\right]$$

"Basic" Adjoint Matching

$$\min_u \mathscr{L}(u) = \int_0^1 \|u_t^\theta(X_t) + \sigma_t \quad a_t(X_{[0,1]}) \quad \|^2 \mathrm{d}t$$

$$X_{[0,1]} \sim p^{\bar{u}}$$
$$\bar{u} = \text{stopgrad}(u)$$

Exactly equivalent to adjoint method: $\frac{\mathrm{d}\mathscr{L}}{\mathrm{d}\theta} = \int_0^1 \frac{\partial u_t^\theta(X_t)}{\partial \theta}^\top \sigma_t\, a_t(X_{[0,1]}) + \frac{\partial}{\partial \theta}\frac{1}{2}\|u_t^\theta(X_t)\|^2 \mathrm{d}t$

Adjoint Matching and the *lean* adjoint state

Adjoint state

$$\frac{\mathrm{d}}{\mathrm{d}t} a_t(X_{[0,1]}) = -\cancel{\boxed{a_t(X_{[0,1]})^\top \nabla_{X_t} (\sigma_t u_t^\theta(X_t)}} + b_t(X_t)) - \cancel{\boxed{\nabla_{X_t}(\tfrac{1}{2}\|u_t^\theta(X_t)\|^2)}}$$

At optimum

$$u_t^*(x) = -\sigma_t \nabla_x J(u^*; t, x)$$

$$u_t^*(x) = \mathbb{E}_{p^u}\left[-\sigma_t\, a_t(X_{[0,1]}) \,|\, X_t = x\right]$$

$$\mathbb{E}_{p^u}\left[u_t^*(X_t) + \sigma_t\, a_t(X_{[0,1]})\right] = 0$$

$$\mathbb{E}_{p^u}\left[\boxed{u_t^*(X_t)^\top \nabla_{X_t} u_t^*(X_t)} + \boxed{\sigma_t\, a_t(X_{[0,1]})^\top \nabla_{X_t} u_t^*(X_t)}\right] = 0$$

Adjoint Matching and the *lean* adjoint state

Adjoint state

$$\frac{\mathrm{d}}{\mathrm{d}t} a_t(X_{[0,1]}) = -\cancel{\boxed{a_t(X_{[0,1]})^\top \nabla_{X_t}(\sigma_t u_t^\theta(X_t)}} + b_t(X_t)) - \cancel{\boxed{\nabla_{X_t}(\tfrac{1}{2}\|u_t^\theta(X_t)\|^2)}}$$

Adjoint Matching

"Lean" adjoint state

$$\frac{\mathrm{d}}{\mathrm{d}t} \tilde{a}_t(X_{[0,1]}) = -\tilde{a}_t(X_{[0,1]})^\top \nabla_{X_t} b_t(X_t)$$

$$\tilde{a}_1(X_{[0,1]}) = -\nabla r(X_1)$$

$$\min_u \mathscr{L}(u) = \int_0^1 \|u_t^\theta(X_t) + \sigma_t \quad \tilde{a}_t(X_{[0,1]}) \quad \|^2 \mathrm{d}t \qquad X_{[0,1]} \sim p^{\bar{u}}$$

$\bar{u}$ = stopgrad(u)

Fixed point $\Longleftrightarrow$ optimal control $\quad \frac{\delta \mathscr{L}(u)}{\delta u} = 0 \quad \Longleftrightarrow \quad u = u^*$

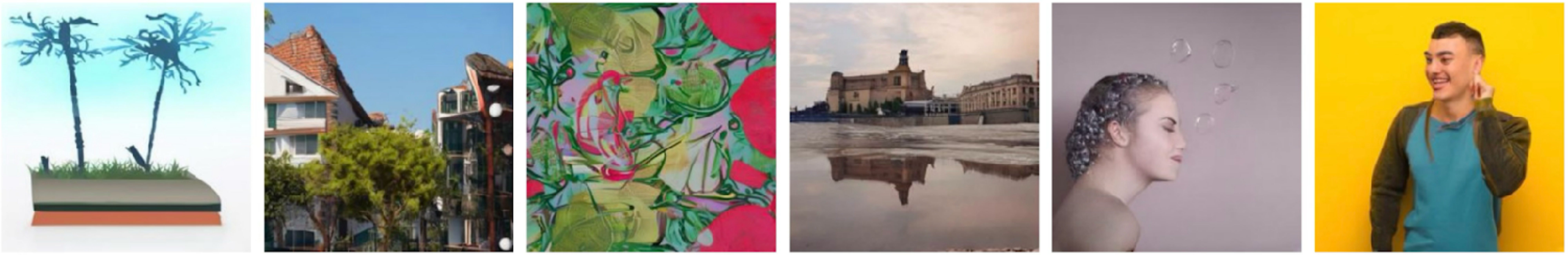
Adjoint Matching for reward fine-tuning

Target distribution

$$p^*(x) \propto p^{\text{base}}(x)\exp\{r(x)\}$$

Flow Matching

$$\mathrm{d}X_t = v_t(X_t)\mathrm{d}t \qquad X_0 \sim p$$



Flow

(Pre-trained)

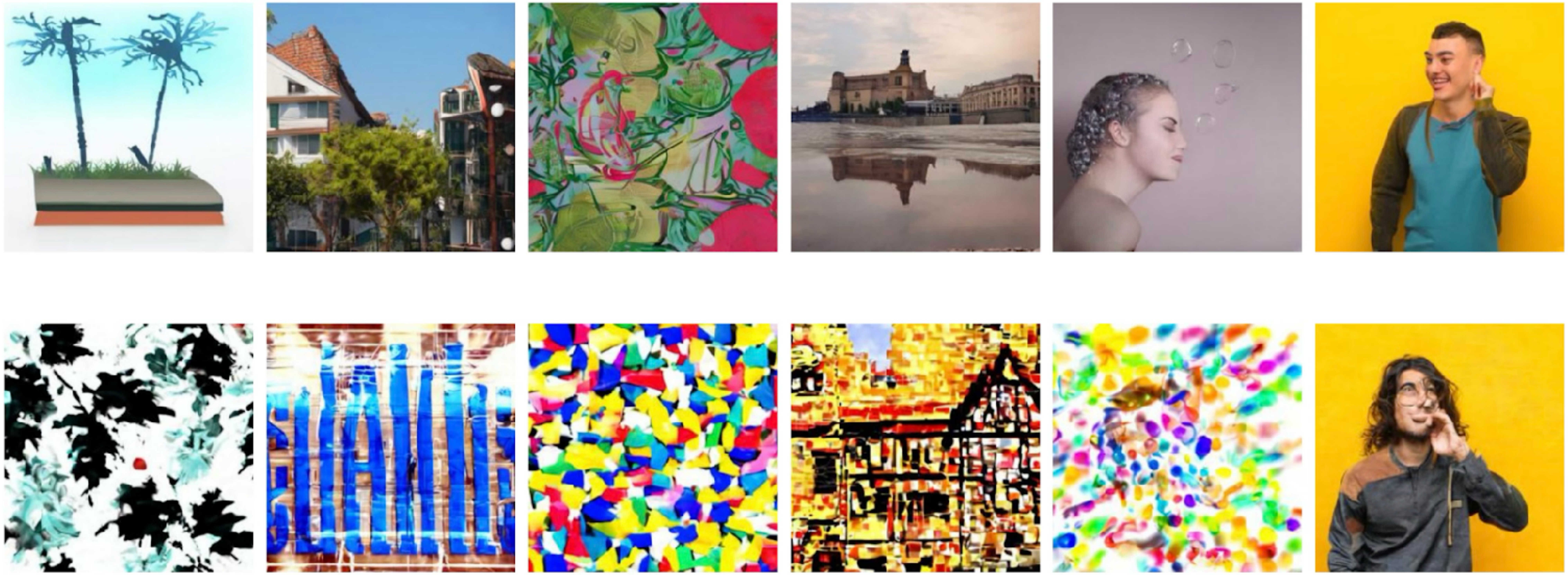
Adjoint Matching for reward fine-tuning

Target distribution

$$p^*(x) \propto p^{\text{base}}(x)\exp\{r(x)\}$$

Memoryless Flow Matching

$$\mathrm{d}X_t = b_t(X_t)\mathrm{d}t + \sigma_t \mathrm{d}B_t \qquad X_0 \sim p$$



Flow

(Pre-trained)

Memoryless

(Pre-trained)

Adjoint Matching for reward fine-tuning

Stochastic control

$$\min_{u} \mathscr{L}(u) = \mathbb{E}_{p^u}\Big[\int_0^1 \frac{1}{2} \|u_t^\theta(X_t)\|^2 \mathrm{d}t - r(X_1) \Big]$$

s.t. $\mathrm{d}X_t = [b_t(X_t) + \sigma_t u_t^\theta(X_t)]\mathrm{d}t + \sigma_t \mathrm{d}B_t \,, \qquad X_0 \sim p$

Memoryless

(Pre-trained)

Memoryless

(Fine-tuned)

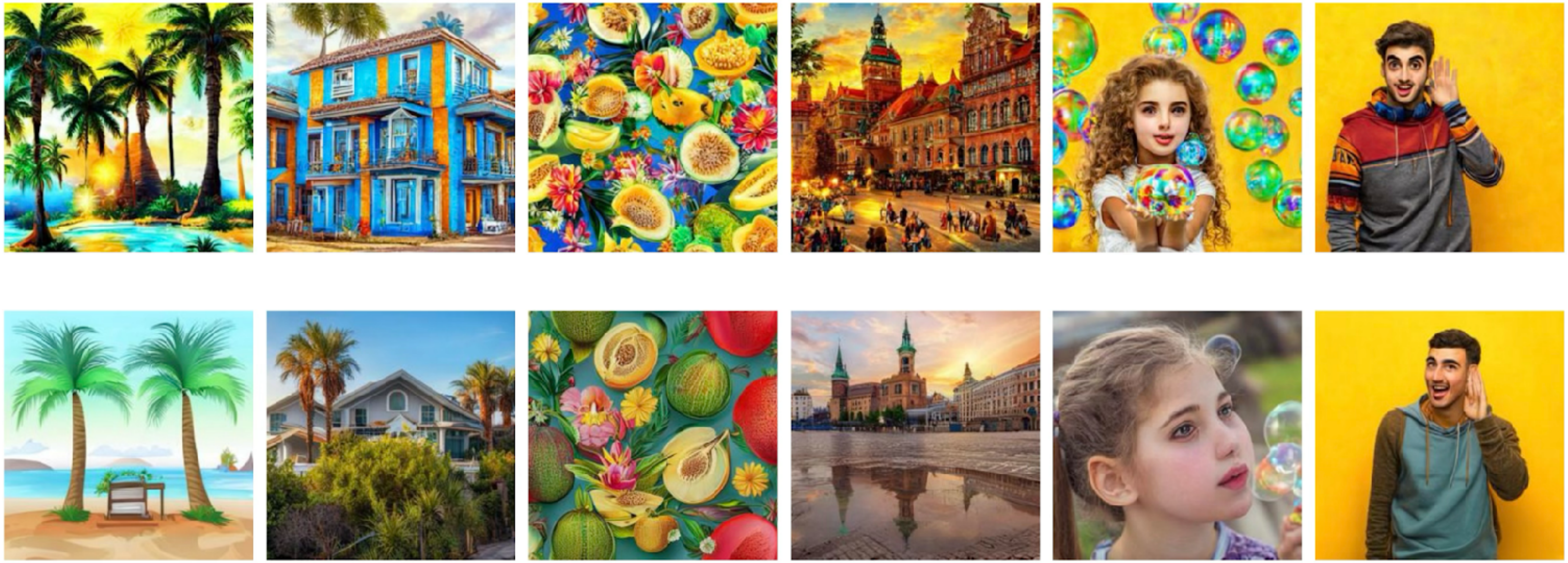
Adjoint Matching for reward fine-tuning

Memoryless Flow Matching

$$\mathrm{d}X_t = b_t(X_t) + \sigma_t u_t^\theta(X_t)\mathrm{d}t + \sigma_t \mathrm{d}B_t$$

Fine-tuned Flow Matching

$$\mathrm{d}X_t = v_t^\theta(X_t)\mathrm{d}t$$



Memoryless
(Fine-tuned)

Flow
(Fine-tuned)

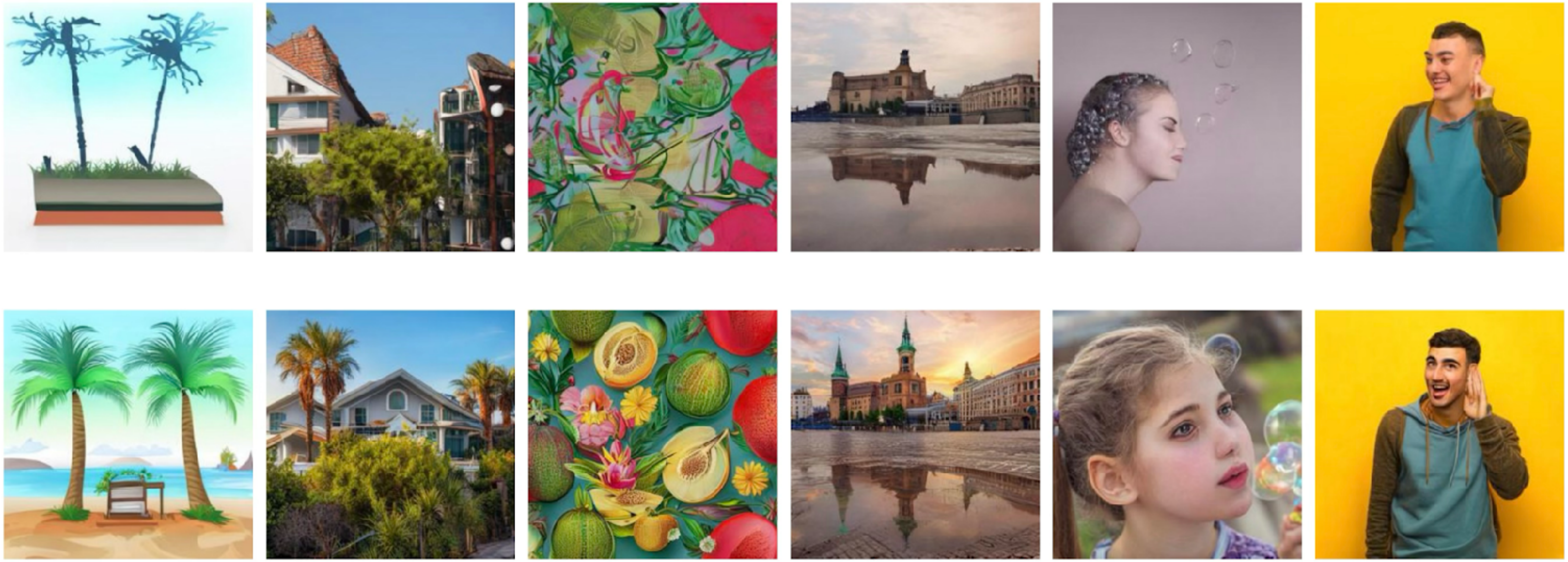
Adjoint Matching for reward fine-tuning

Memoryless Flow Matching

$$\mathrm{d}X_t = b_t(X_t) + \sigma_t u_t^{\theta}(X_t)\mathrm{d}t + \sigma_t \mathrm{d}B_t \longrightarrow$$

Fine-tuned Flow Matching

$$\mathrm{d}X_t = v_t^{\theta}(X_t)\mathrm{d}t$$



Flow
(Pre-trained)

Flow
(Fine-tuned)

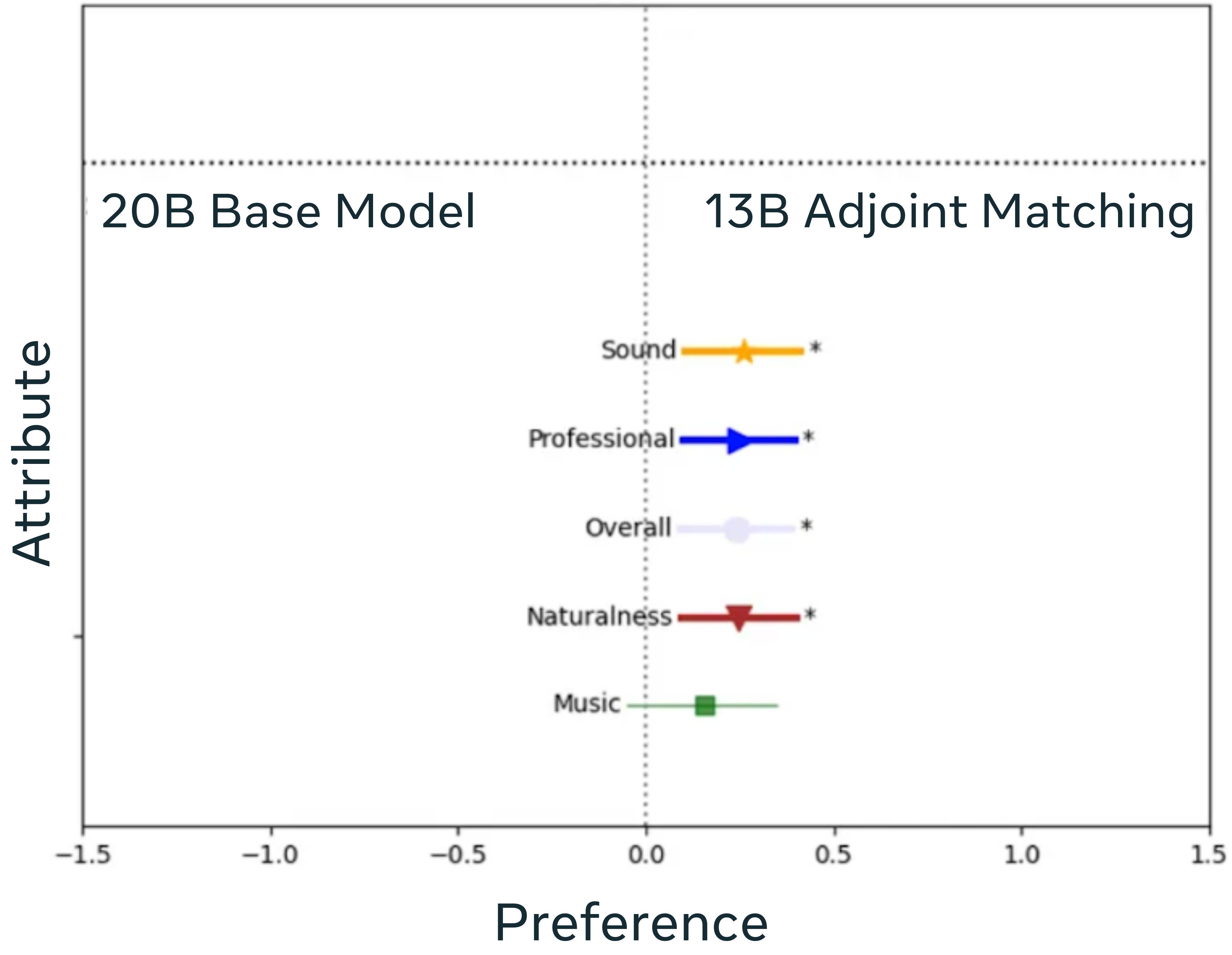
Reward fine-tuning for MovieGen Audio

Meta MovieGen

Text & Video → Audio

Text input: “Whistling sounds, followed by a sharp explosion and loud crackling.”

Human Evaluation



(Statistically significant improvement)

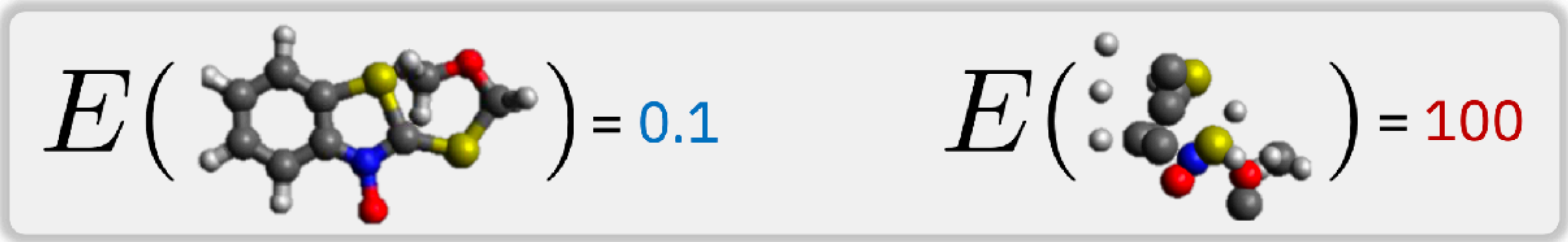
III. How to scale the method?

Adjoint Sampling: highly scalable method for amortized sampling at scale

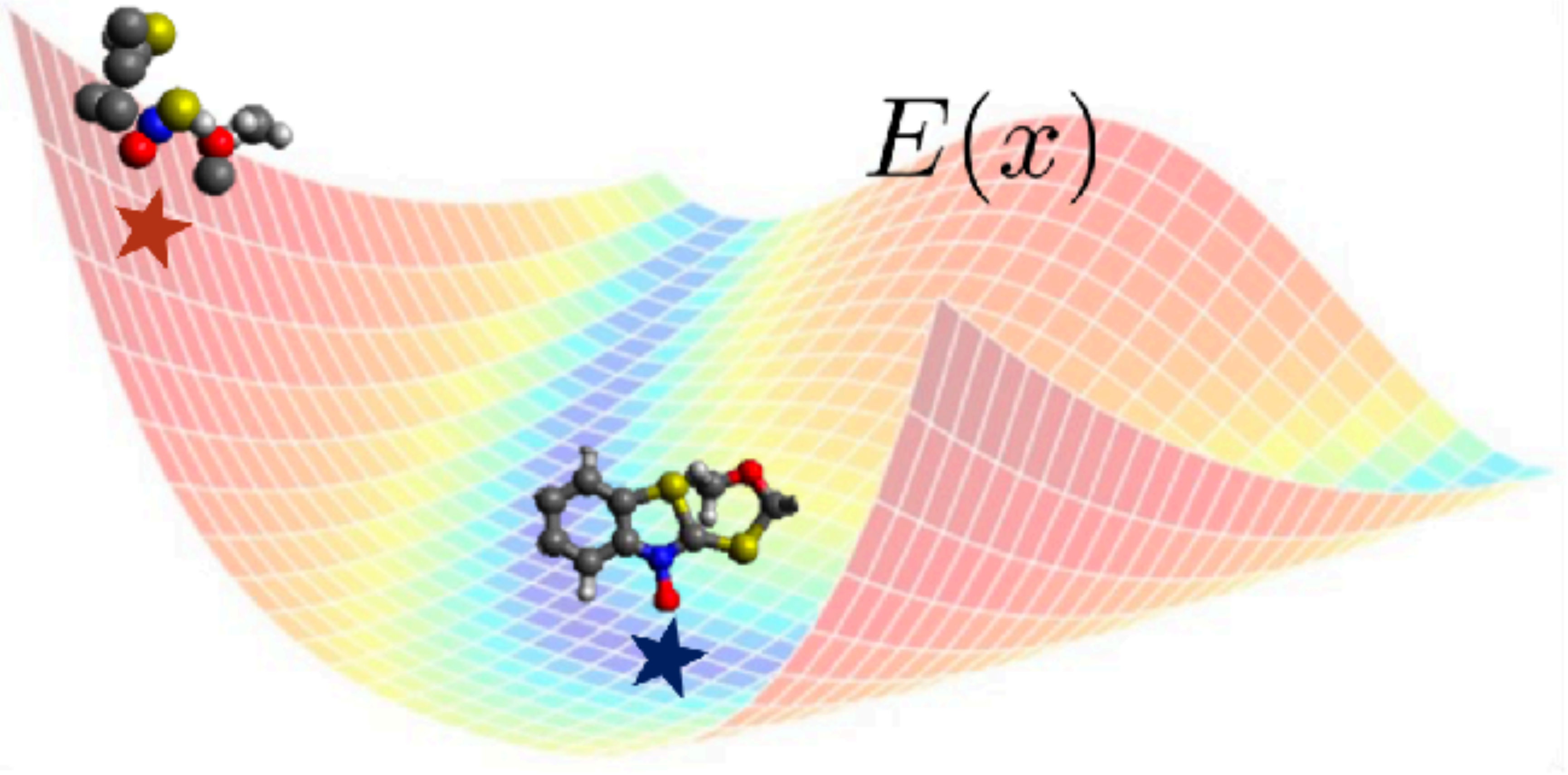
Stochastic control formulation for sampling

Sampling from unnormalized distribution

$$p^*(x) \propto \exp\{-E(x)\}$$



low energy → stable structure → likely to appear → high probability
high energy → unstable structure → unlikely to appear → low probability



[!] Estimating this energy is also **very expensive**.

Stochastic control formulation for sampling

Sampling from unnormalized distribution

$$p^*(x) \propto \exp\{-E(x)\}$$

Know how to target this:

$$p_1^*(x) \propto p_1^{\text{base}}(X_1)\exp\{r(X_1)\}$$

Can define reward:

$$r(X_1) = -E(X_1) - \log p_1^{\text{base}}(X_1)$$

Stochastic control objective

$$\min_u \mathscr{L}(u) = \mathbb{E}_{p^u}\left[\int_0^1 \frac{1}{2}\|u_t^\theta(X_t)\|^2 \mathrm{d}t + E(X_1) + \log p_1^{\text{base}}(X_1)\right]$$

s.t. $\mathrm{d}X_t = [b_t(X_t) + \sigma_t u_t^\theta(X_t)]\mathrm{d}t + \sigma_t \mathrm{d}B_t\,, \qquad X_0 \sim p$

Adjoint Matching simplified

Base generative process

$$\mathrm{d}X_t = b_t(X_t)\mathrm{d}t + \sigma_t \mathrm{d}B_t \qquad X_0 = 0$$

Adjoint Matching

"Lean" adjoint state

$$\frac{\mathrm{d}}{\mathrm{d}t}\tilde{a}_t(X_{[0,1]}) = -\tilde{a}_t(X_{[0,1]})^\top \nabla_{X_t} b_t(X_t)$$

$$\tilde{a}_1(X_{[0,1]}) = \nabla E(X_1) + \nabla \log p_1^{\text{base}}(X_1)$$

$$\int_0^1 \| u_t^\theta(X_t) + \sigma_t \quad \tilde{a}_t(X_{[0,1]}) \quad \|^2 \mathrm{d}t \qquad X_{[0,1]} \sim p^{\bar{u}}$$

$$\bar{u} = \text{stopgrad}(u)$$

Adjoint Matching simplified

Base process

$$\mathrm{d}X_t = \sigma_t \mathrm{d}B_t \qquad X_0 = 0$$

Adjoint Matching

"Lean" adjoint state

$$\frac{\mathrm{d}}{\mathrm{d}t}\tilde{a}_t(X_{[0,1]}) = 0$$

$$\tilde{a}_1(X_{[0,1]}) = \nabla E(X_1) + \nabla \log p_1^{\text{base}}(X_1)$$

$$\int_0^1 \| u_t^\theta(X_t) + \sigma_t \quad \tilde{a}_t(X_{[0,1]}) \quad \|^2 \mathrm{d}t$$

$$X_{[0,1]} \sim p^{\bar{u}}$$

$$\bar{u} = \text{stopgrad}(u)$$

Adjoint Matching simplified

Base process

$$\mathrm{d}X_t = \sigma_t \mathrm{d}B_t \qquad X_0 = 0$$

Adjoint Matching

$$\int_0^1 \mathbb{E}_{(X_t, X_1) \sim p^u} \| u_t^\theta(X_t) + \sigma_t (\nabla E(X_1) + \nabla \log p_1^{\text{base}}(X_1)) \|^2 \mathrm{d}t^{\ast}$$

*Also appeared in Particle Denoising Diffusion Sampler (Phillips et al. 2024)

(But focuses on using SMC for ground truth samples.)

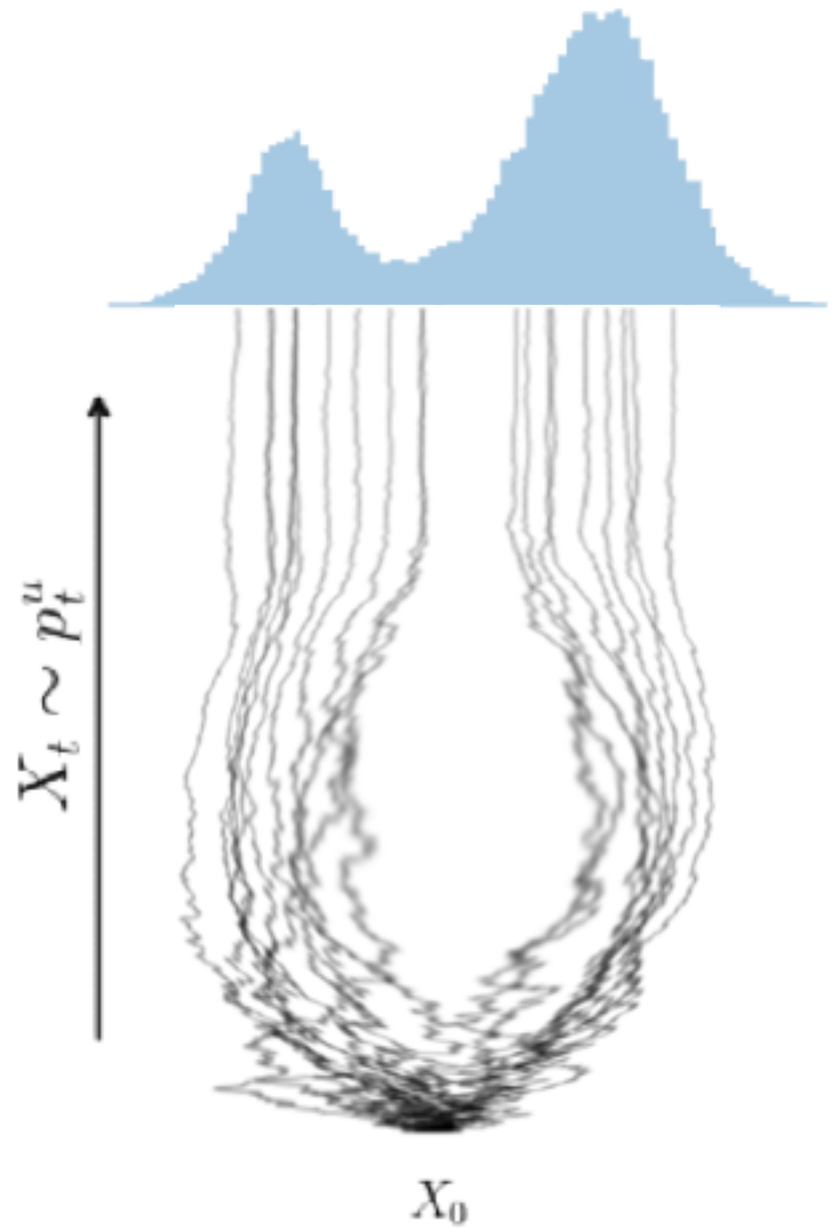
Improving optimality with a projection

Adjoint Matching

$$\int_0^1 \mathbb{E}_{(X_t,X_1)\sim p^u} \| u_t^\theta(X_t) + \sigma_t(\nabla E(X_1) + \nabla \log p_1^{\text{base}}(X_1)) \|^2 \mathrm{d}t$$

Recall optimal stochastic process

$$u^* = \arg\min_\nu D_{\mathrm{KL}}(p^\nu(X_{[0,1]}) \| p^{\text{base}}(X_{[0,1]} \mid X_1) p^*(X_1))$$



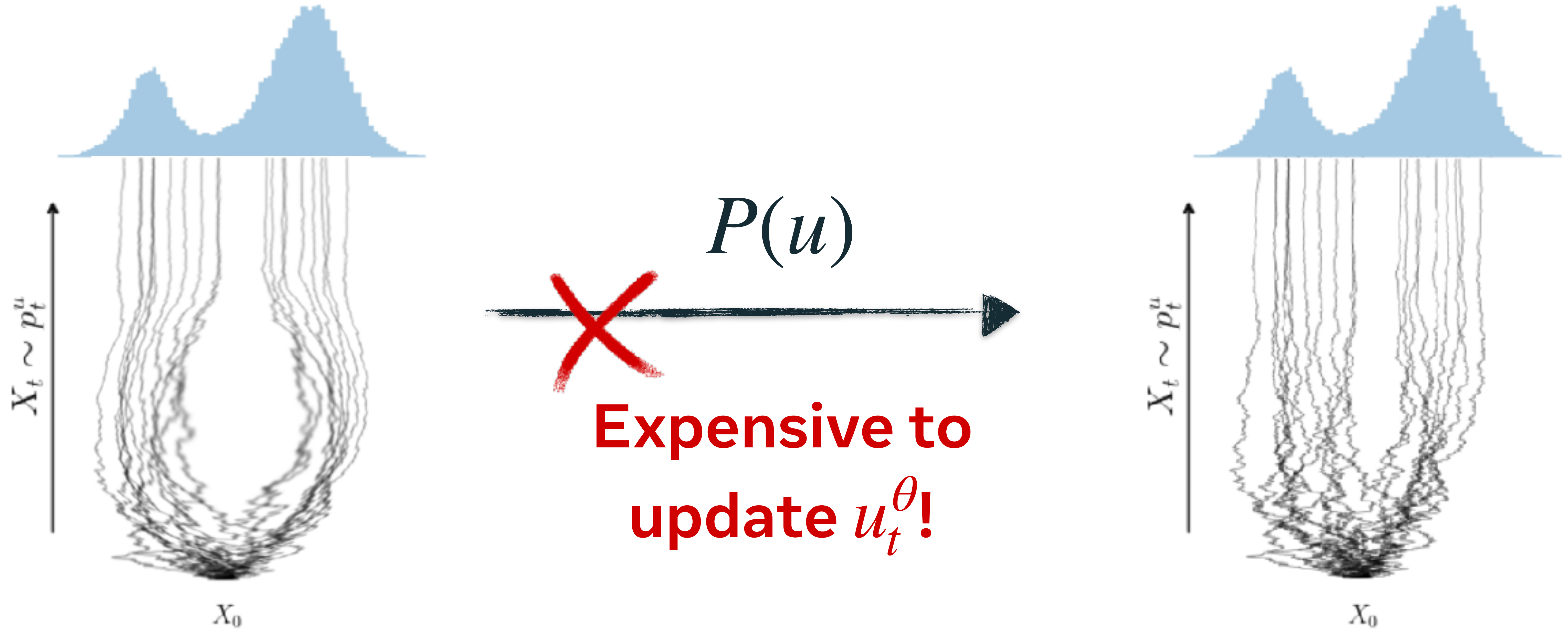
Improving optimality with a projection

Adjoint Matching

$$\int_0^1 \mathbb{E}_{(X_t,X_1)\sim p^u} \| u_t^\theta(X_t) + \sigma_t(\nabla E(X_1) + \nabla \log p_1^{\text{base}}(X_1)) \|^2 \mathrm{d}t$$

Projection of the current control

$$P(u) = \arg\min_v D_{\mathrm{KL}}(p^v(X_{[0,1]}) \| p^{\text{base}}(X_{[0,1]} \mid X_1) p^u(X_1))$$



Improving optimality with a projection

Reciprocal Adjoint Matching

$$\int_0^1 \mathbb{E}_{X_1 \sim p_1^u, \boxed{X_t \sim p^{\text{base}}(X_t | X_1)}} \| u_t^\theta(X_t) + \sigma_t (\nabla E(X_1) + \nabla \log p_1^{\text{base}}(X_1)) \|^2 \mathrm{d}t$$

Sample $X_t \,|\, X_1$ optimally

Improving optimality with a projection

Reciprocal Adjoint Matching

$$\int_0^1 \mathbb{E}_{X_1 \boxed{\sim p_1^u}, \boxed{X_t \sim p^{\text{base}}(X_t|X_1)}} \| v_t(X_t) + \sigma_t(\nabla E(X_1) + \nabla \log p_1^{\text{base}}(X_1)) \|^2 \mathrm{d}t$$

Leave Fixed

Sample $X_t \mid X_1$ optimally

$=: \mathcal{L}_{\text{RAM}}(v; u)$

Theory of RAM: project for free

$$u_{i+1} = \arg\min_v \mathcal{L}_{\text{RAM}}(v; u_i)$$

$$= P(u_i) - \frac{\delta \mathcal{L}_{\text{AM}}}{\delta u}(P(u_i))$$

i.e., Equivalent to projecting the control then performing Adjoint Matching

Fixed point $\iff$ optimal control

Adjoint Sampling

Reciprocal Adjoint Matching

$$\int_0^1 \mathbb{E}_{X_1 \boxed{\sim \mathcal{B}}, X_t \sim p^{\text{base}}(X_t|X_1)} \|u_t^\theta(X_t) + \sigma_t(\nabla E(X_1) + \nabla \log p_1^{\text{base}}(X_1))\|^2 \mathrm{d}t$$

Replay Buffer*

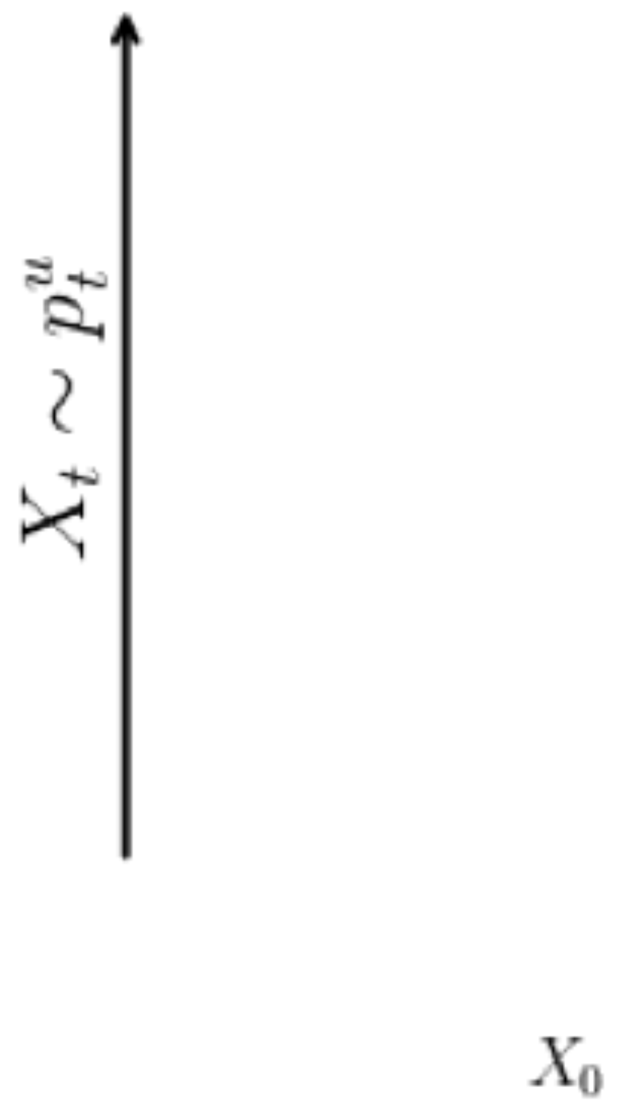
Adjoint Sampling Algorithm

Alternate between:

1. $\mathcal{B} \leftarrow \{X_1^i, E^i\}, \ X_1^i \sim p_1^u, E^i = E(X_1)$
2. Optimize $\mathcal{L}_{\text{RAM}}$ multiple iterations

Infrequent sample generation & energy evaluation.

Very fast gradient updates.

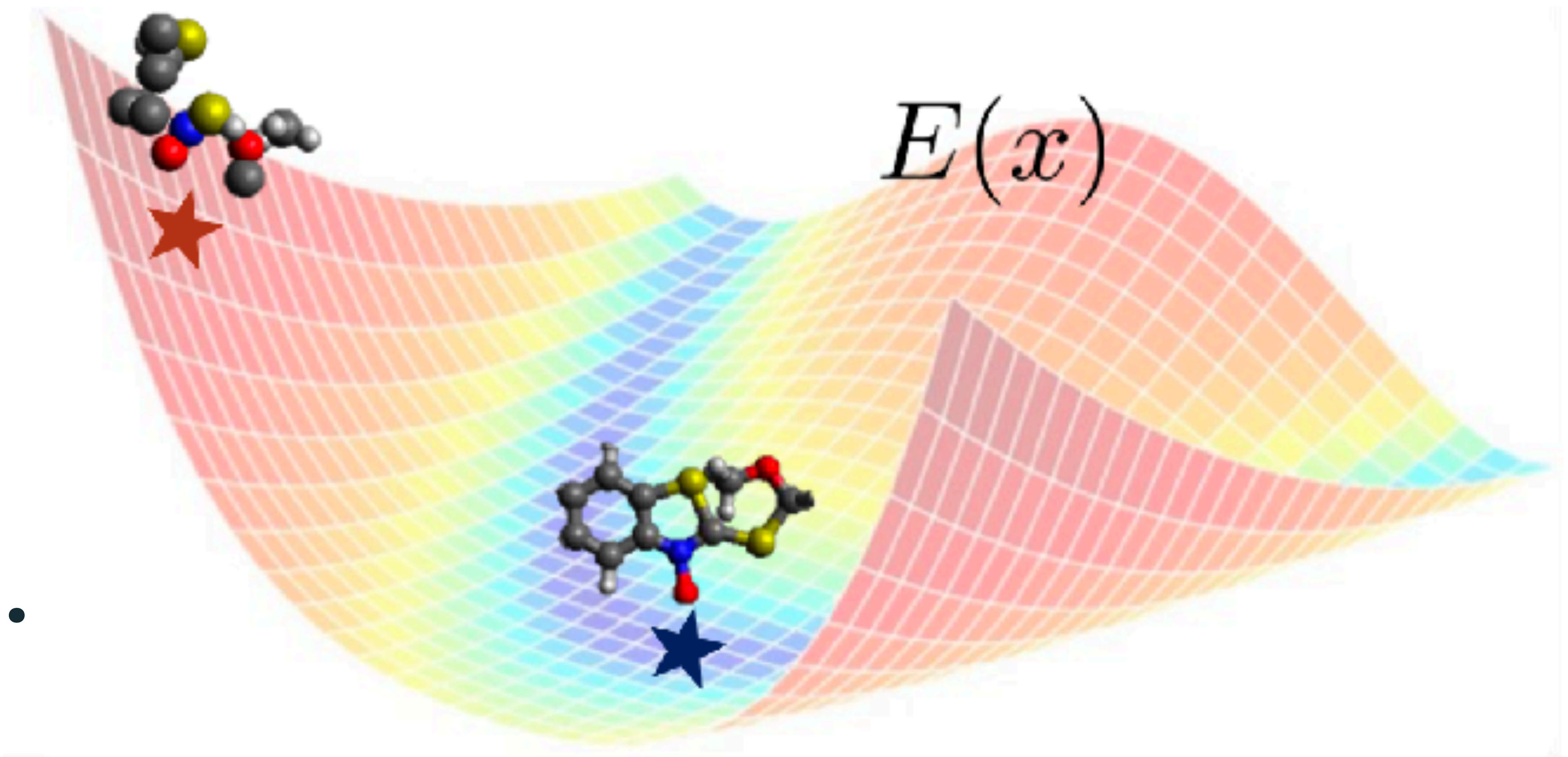


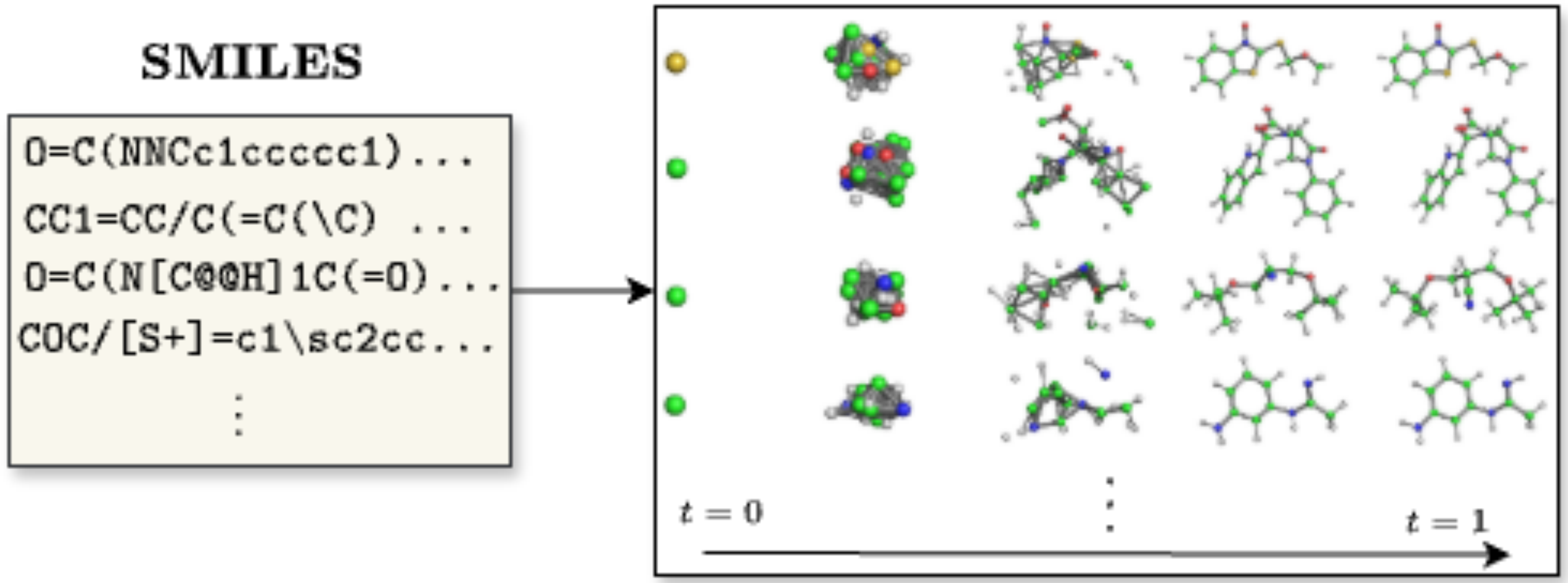
* Iterated Denoising Energy Matching for Sampling from Boltzmann Densities (Akhound-Sadegh et al. 2024)

A new benchmark for highly scalable sampling

Evaluates sampling methods on both **efficiency** and **generalization**.

Energy evaluations modeled by **deep graph neural networks**.



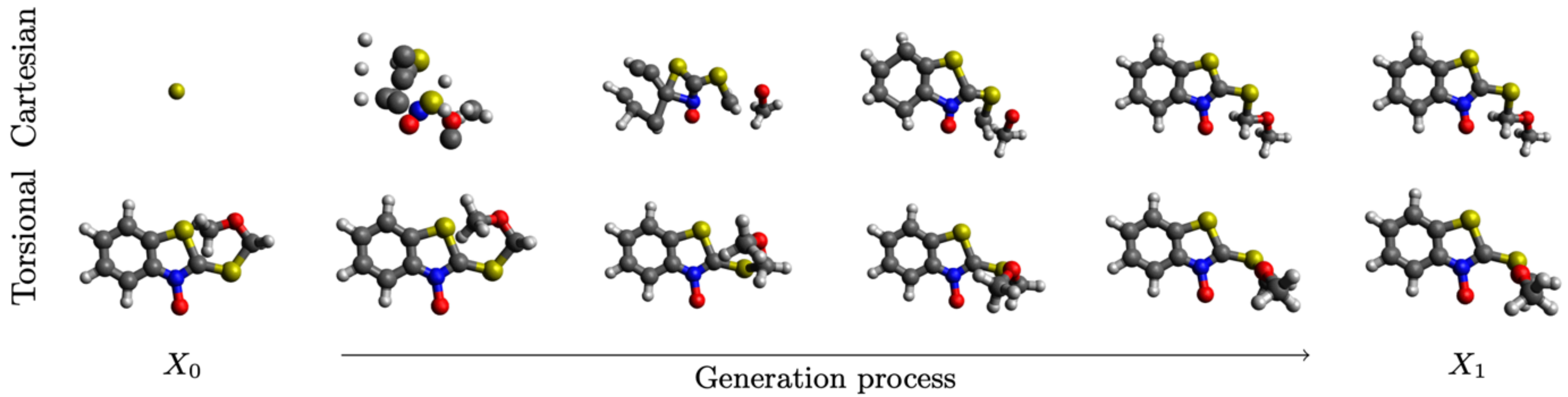


Train on **24,000** molecules.

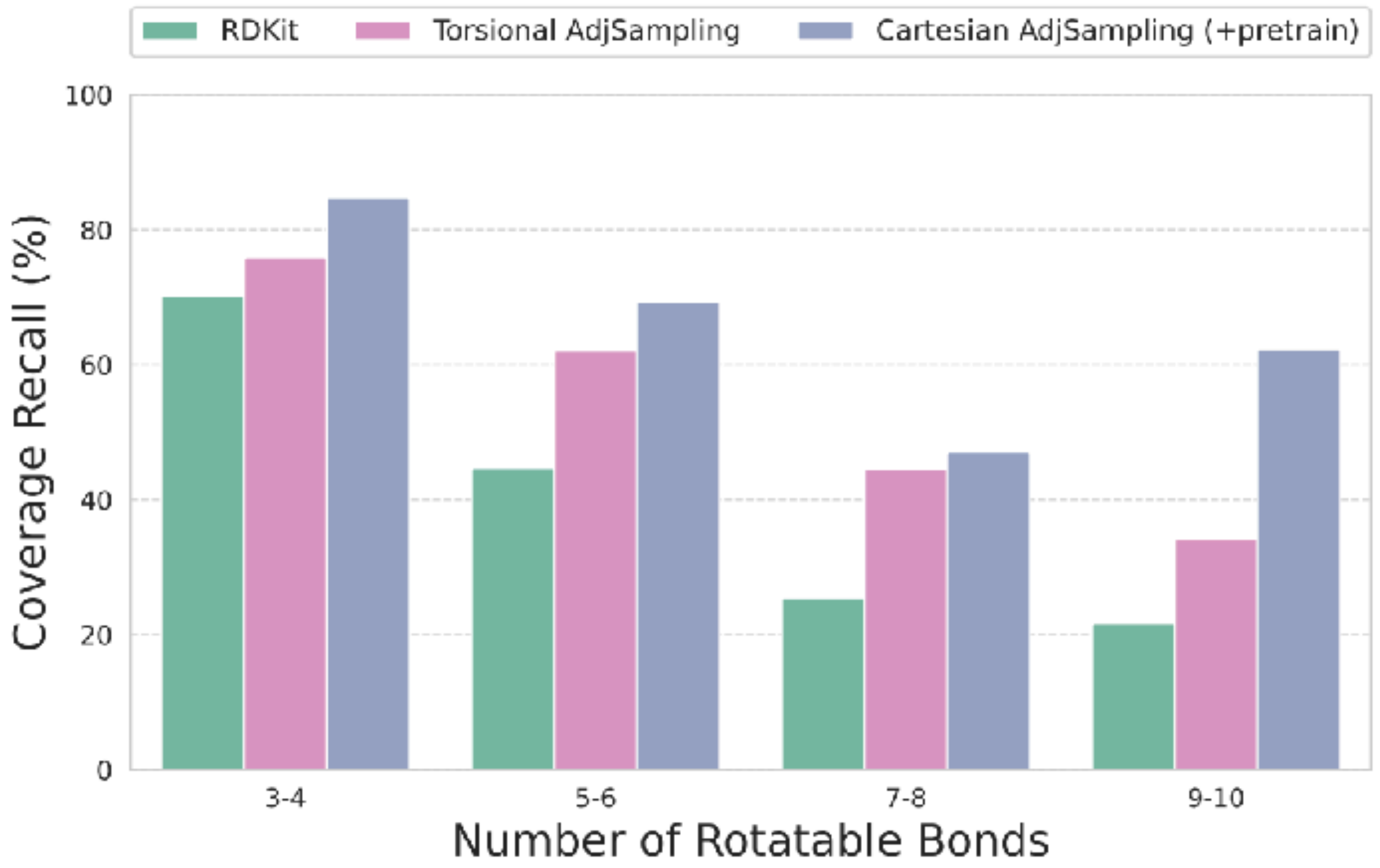
Tests generalization to **unseen molecules**.

Each with $\mathcal{O}(100)$ of local minima. **Need to find all.**

A new benchmark for highly scalable sampling

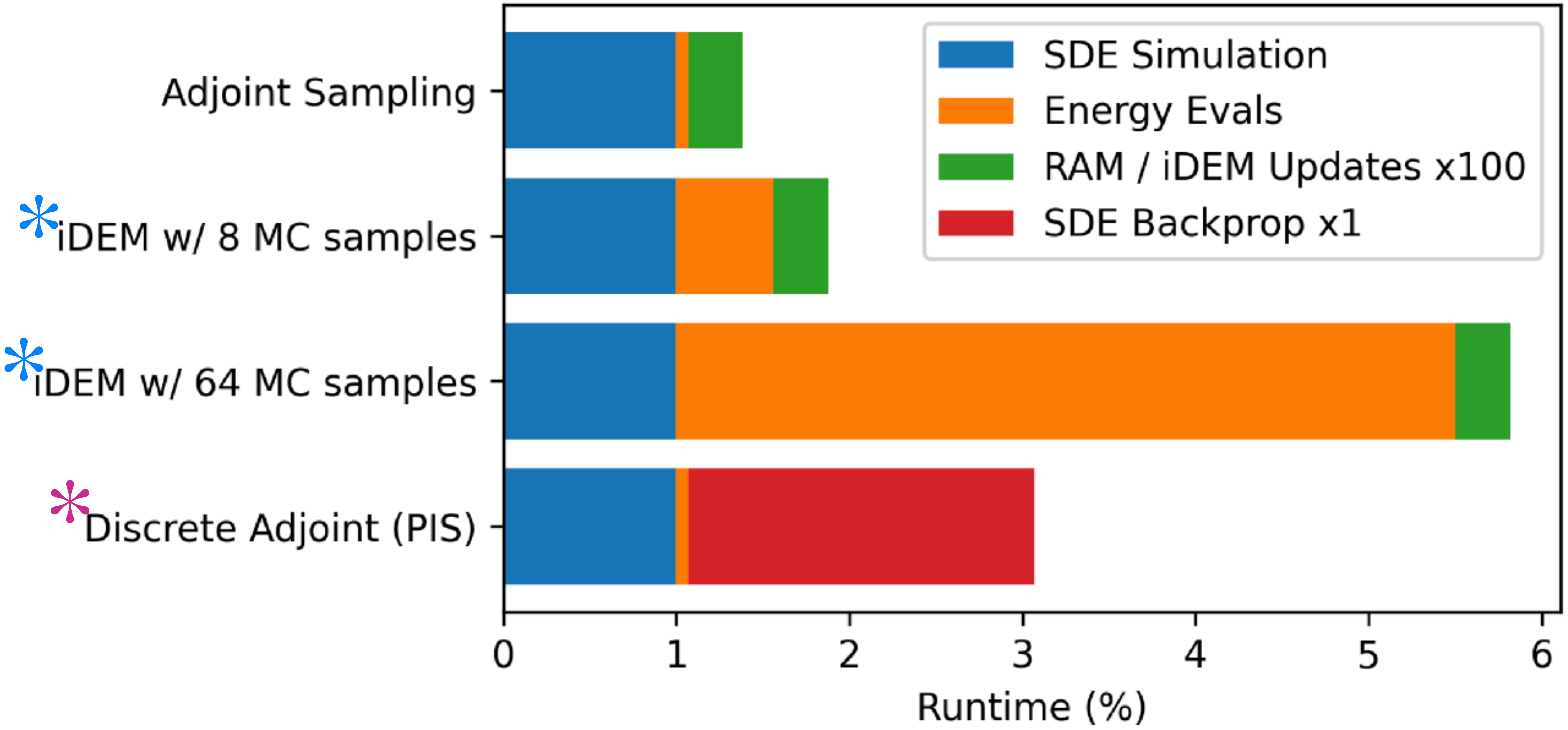


Larger improvement over RD-KIT at higher difficulties



More difficult / more local minima

A new benchmark for highly scalable sampling



* Iterated Denoising Energy Matching for Sampling from Boltzmann Densities (Akhound-Sadegh et al. 2024)

* Path Integral Sampler: a stochastic control approach for sampling (Zhang & Chen 2021)

Papers & Collaborators

Adjoint Matching: Fine-tuning Flow and Diffusion Generative Models with Memoryless Stochastic Optimal Control

Carles Domingo-Enrich[1], Michal Drozdzal[1], Brian Karrer[1], Ricky T. Q. Chen[1]

[1]FAIR, Meta

Adjoint Sampling: Highly Scalable Diffusion Samplers via Adjoint Matching

Aaron Havens[2,†,*], Benjamin Kurt Miller[1,*], Bing Yan[1,3,*], Carles Domingo-Enrich[4], Anuroop Sriram[1], Brandon Wood[1], Daniel Levine[1], Bin Hu[2], Brandon Amos[1], Brian Karrer[1], Xiang Fu[1,*], Guan-Horng Liu[1,*], Ricky T. Q. Chen[1,*]

[1]FAIR at Meta, [2]University of Illinois, [3]New York University, [4]Microsoft Research New England
[*]Core contributors, [†]Work done during internship at FAIR

Adjoint Matching open source re-implementation:

https://github.com/microsoft/soc-fine-tuning-sd