Annealed Diffusion Models

for

Efficient Transport in LFT

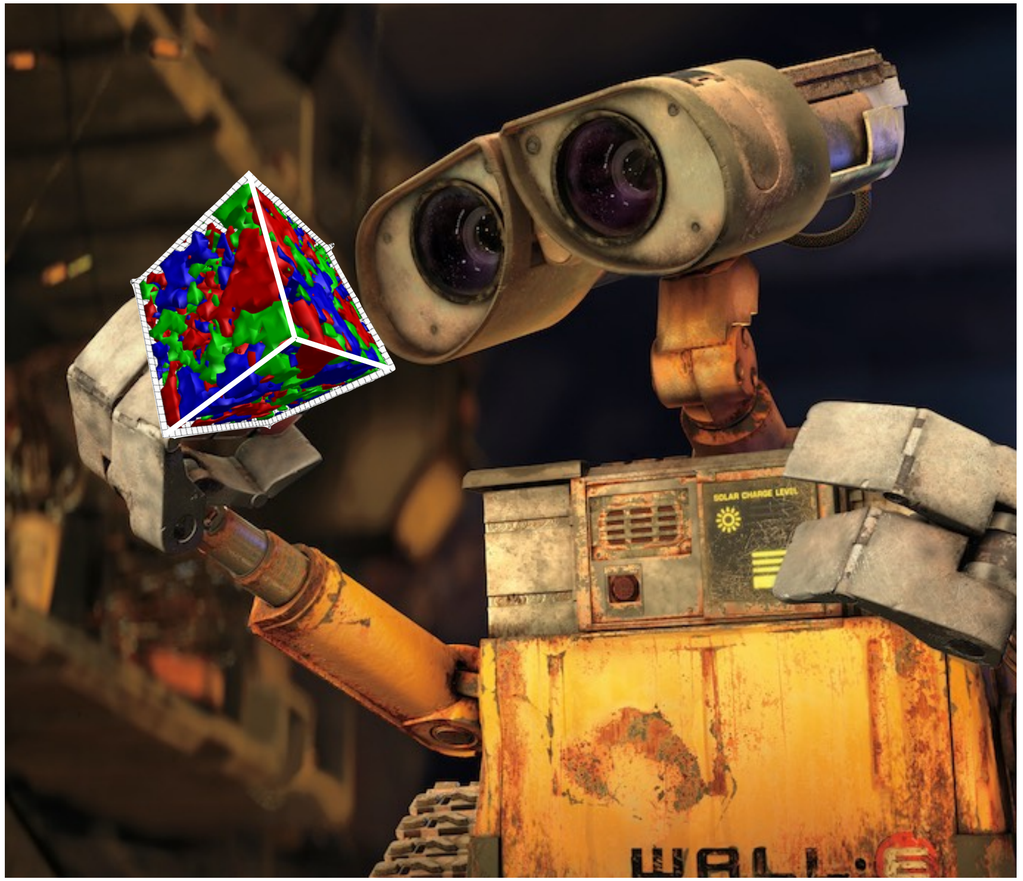


Image credits: Leinweber, et al; Pixar.

Gurtej Kanwar
Chancellor's Fellow in AI & Datascience
University of Edinburgh

Sept 7, 2026
Bridging ML and Field Theory, ECT*, Trento



III MACHINE LEARNING APPROACHES IN LATTICE QCD - AN INTERDISCIPLINARY EXCHANGE

May 2027

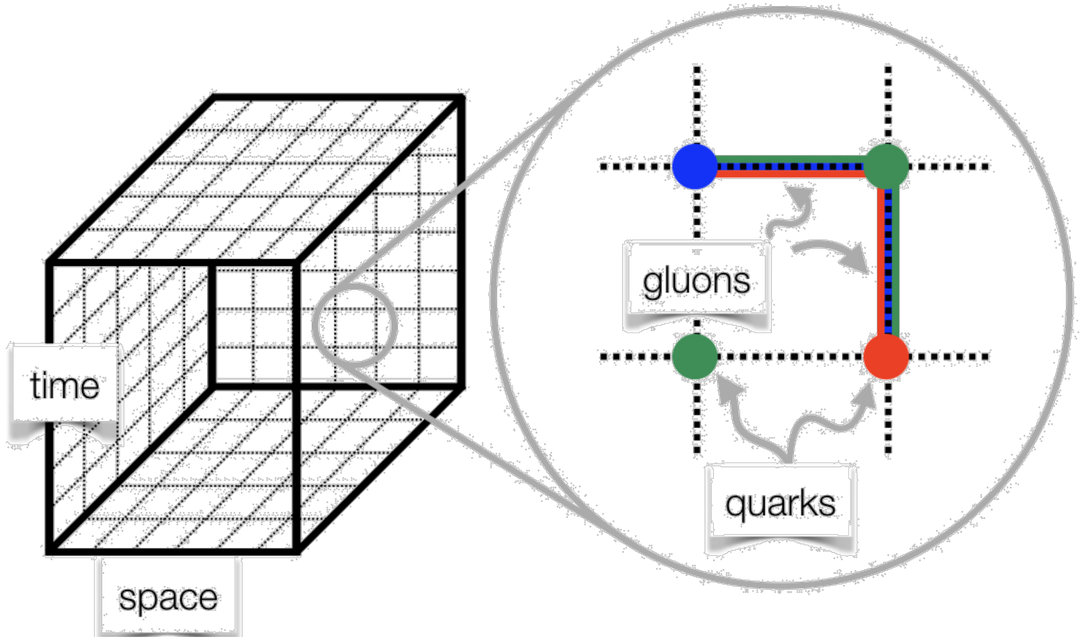
Higgs Centre for Theoretical Physics
University of Edinburgh

THE UNIVERSITY of EDINBURGH

Lattice simulations

High-dimensional path integral over degrees of freedom assigned to points and edges of a lattice

- Boltzmann weight $e^{-S(U)}$ encodes distribution over field configurations



Partition function

$$Z \equiv \left[\prod_{x,\mu} \int dU_\mu(x)\right] e^{-S(U)}$$

Thermal expt. value of operator $\mathcal{O}$

$$\langle \mathcal{O} \rangle = \left[\prod_{x,\mu} \int dU_\mu(x)\right] \mathcal{O}(U)\, e^{-S(U)}/Z$$

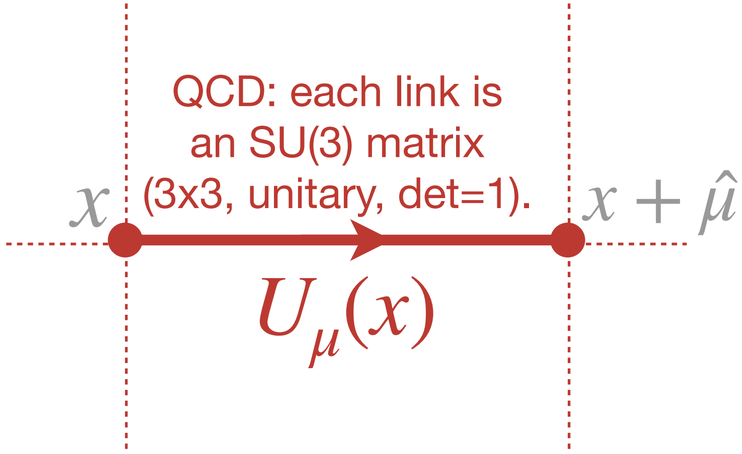


Image credit: Evan Berkowitz

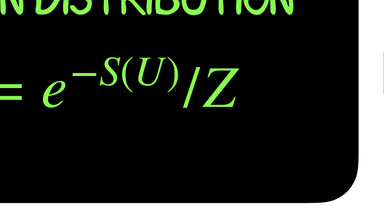
Monte Carlo simulation

$$\langle \mathcal{O} \rangle = \left[\prod_{x,\mu} \int dU_\mu(x) \right] \mathcal{O}(U)\, e^{-S(U)}/Z$$

Approximate the path integral using ***Markov chain Monte Carlo***

BOLTZMANN DISTRIBUTION

$$p(U) = e^{-S(U)}/Z$$



MONTE CARLO AVERAGE

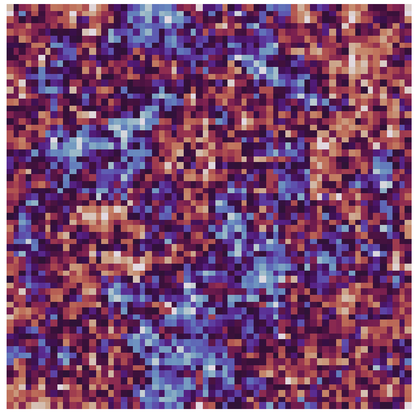
$$\langle \mathcal{O} \rangle \approx \frac{1}{n} \sum_{i=1}^{n} \mathcal{O}(U_i)$$

Image credit: Leinweber, et al.

Monte Carlo simulation

$$\langle \mathcal{O} \rangle = \left[\prod_{x,\mu} \int dU_\mu(x) \right] \mathcal{O}(U)\, e^{-S(U)}/Z$$

Approximate the path integral using ***Markov chain Monte Carlo***



BOLTZMANN DISTRIBUTION

$$p(U) = e^{-S(U)}/Z$$

MONTE CARLO AVERAGE

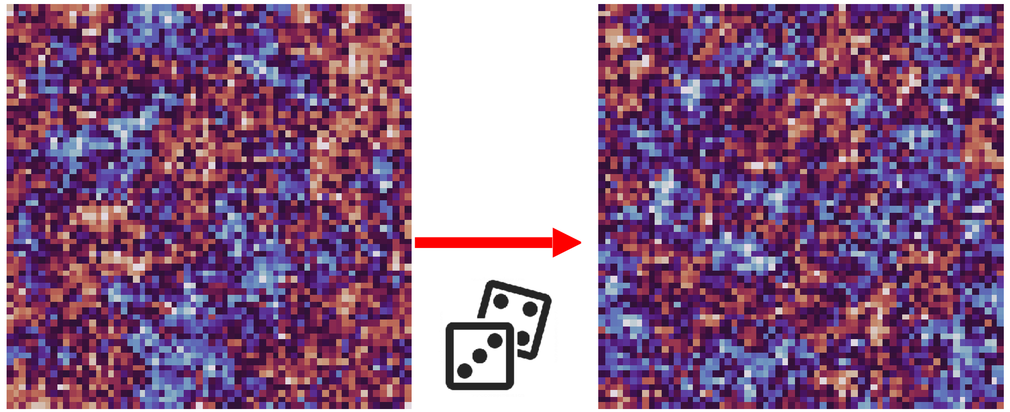
$$\langle \mathcal{O} \rangle \approx \frac{1}{n} \sum_{i=1}^{n} \mathcal{O}(U_i)$$

Image credit: Leinweber, et al.

Monte Carlo simulation

$$\langle \mathcal{O} \rangle = \left[\prod_{x,\mu} \int dU_\mu(x) \right] \mathcal{O}(U)\, e^{-S(U)}/Z$$

Approximate the path integral using ***Markov chain Monte Carlo***



BOLTZMANN DISTRIBUTION

$$p(U) = e^{-S(U)}/Z$$

MONTE CARLO AVERAGE

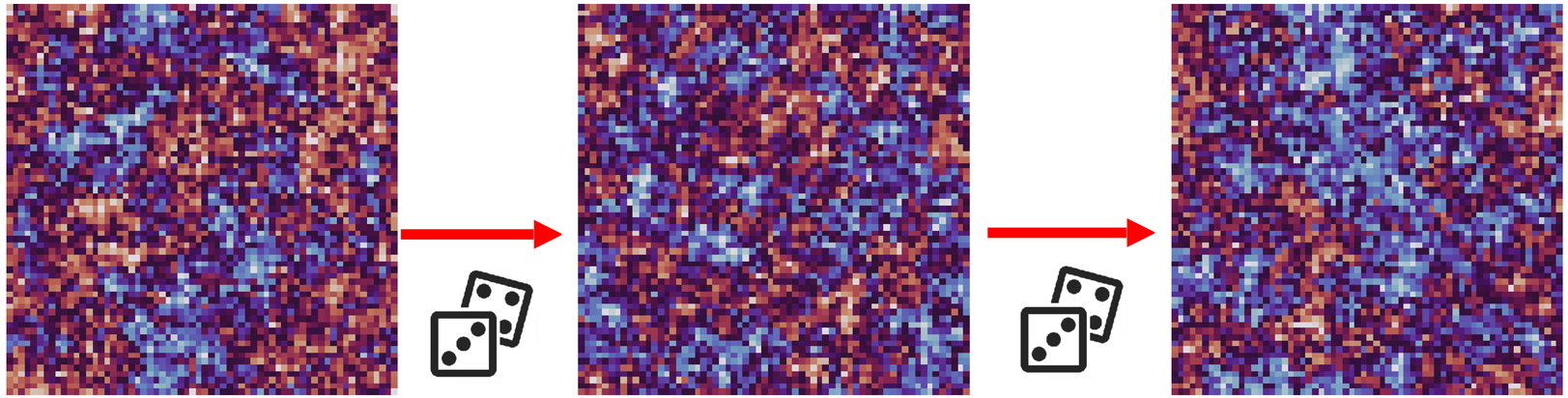
$$\langle \mathcal{O} \rangle \approx \frac{1}{n} \sum_{i=1}^{n} \mathcal{O}(U_i)$$

Image credit: Leinweber, et al.

Monte Carlo simulation

$$\langle \mathcal{O} \rangle = \left[\prod_{x,\mu} \int dU_\mu(x) \right] \mathcal{O}(U)\, e^{-S(U)}/Z$$

Approximate the path integral using ***Markov chain Monte Carlo***



BOLTZMANN DISTRIBUTION

$$p(U) = e^{-S(U)}/Z$$

MONTE CARLO AVERAGE

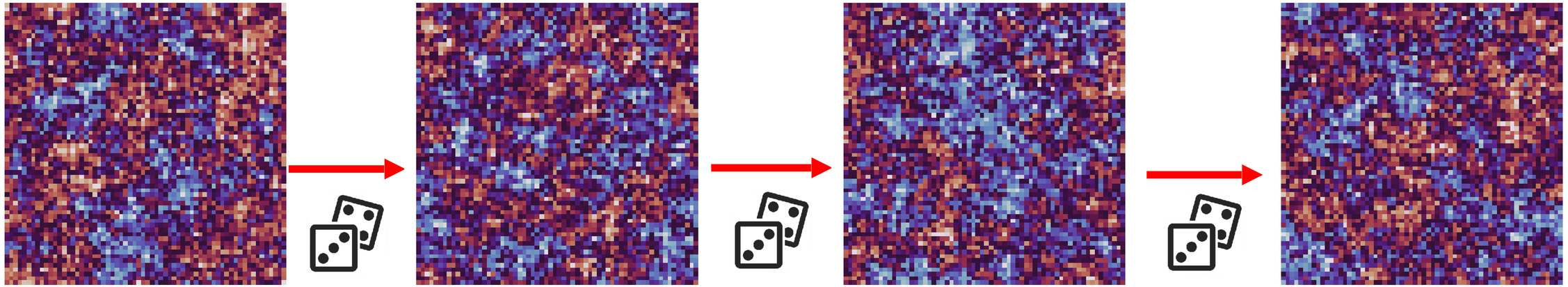
$$\langle \mathcal{O} \rangle \approx \frac{1}{n} \sum_{i=1}^{n} \mathcal{O}(U_i)$$

Image credit: Leinweber, et al.

Monte Carlo simulation

$$\langle \mathcal{O} \rangle = \left[\prod_{x,\mu} \int dU_\mu(x) \right] \mathcal{O}(U)\, e^{-S(U)}/Z$$

Approximate the path integral using ***Markov chain Monte Carlo***



BOLTZMANN DISTRIBUTION

$$p(U) = e^{-S(U)}/Z$$

MONTE CARLO AVERAGE

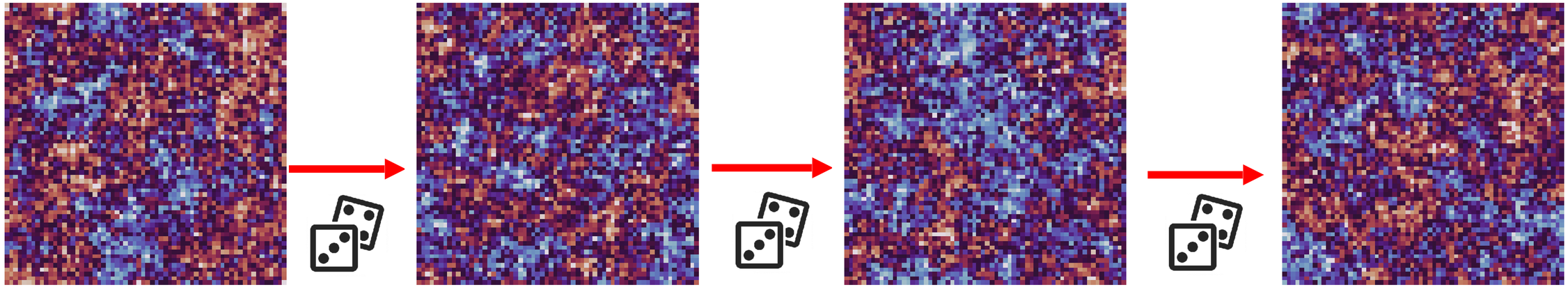
$$\langle \mathcal{O} \rangle \approx \frac{1}{n} \sum_{i=1}^{n} \mathcal{O}(U_i)$$

Image credit: Leinweber, et al.

Monte Carlo simulation

$$\langle \mathcal{O} \rangle = \left[\prod_{x,\mu} \int dU_\mu(x) \right] \mathcal{O}(U)\, e^{-S(U)}/Z$$

Approximate the path integral using ***Markov chain Monte Carlo***



- ⚠ 4D grids of 3×3 matrices
- ⚠ $O(10{,}000{,}000{,}000)$ components/sample

BOLTZMANN DISTRIBUTION

$$p(U) = e^{-S(U)}/Z$$

MONTE CARLO AVERAGE

$$\langle \mathcal{O} \rangle \approx \frac{1}{n} \sum_{i=1}^{n} \mathcal{O}(U_i)$$

Image credit: Leinweber, et al.

Critical slowing down (CSD)

Markov chains are typically **inefficient** as $a \to 0$

- Information transfer is local while correlations grow
- **CSD:** diverging autocorrelation time
- **Topological freezing:** Markov chain gets “stuck”

This **limits the physics results:**

- Heavy quarks (short-range interactions)
- Systematic uncertainties

Critical slowing down (CSD)

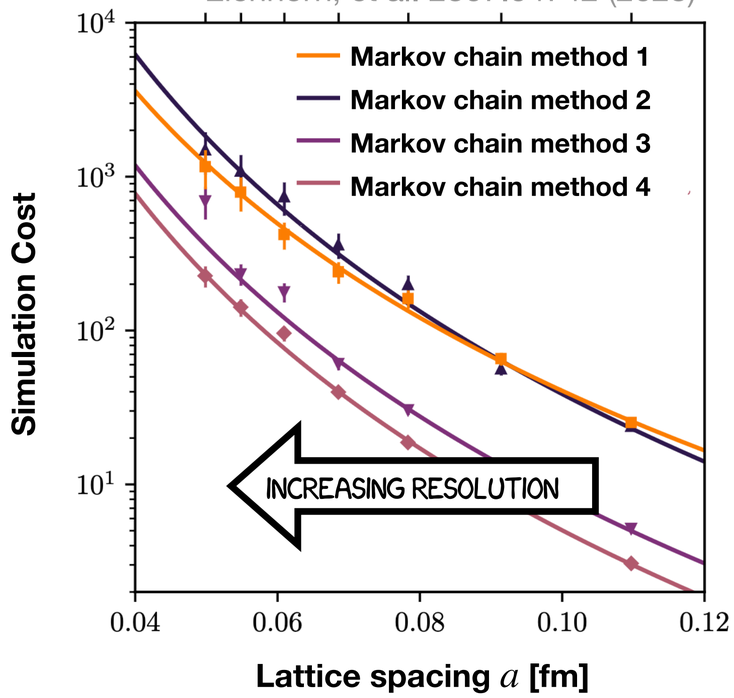
Markov chains are typically **inefficient** as $a \to 0$

- Information transfer is local while correlations grow
- **CSD:** diverging autocorrelation time
- **Topological freezing:** Markov chain gets “stuck”

This **limits the physics results:**

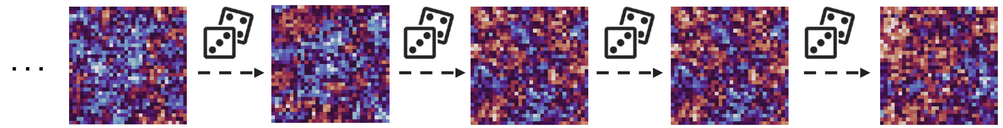
- Heavy quarks (short-range interactions)
- Systematic uncertainties

Eichhorn, et al. 2307.04742 (2023)



Can we use **Generative models** to accelerate sampling?

Replace or augment **Markov Chain Monte Carlo**…



… with generated samples?



Measure transport

1. Start from samples V from **initial probability measure**
   - Typically a simple "prior density" $r(V)$
2. Apply a **transformation** $f : V \mapsto U$
   - Deterministic transformation squishes/stretches
   - Stochastic transformation mixes/diffuses
3. **Result:** a new probability measure $q(U)$ over samples U

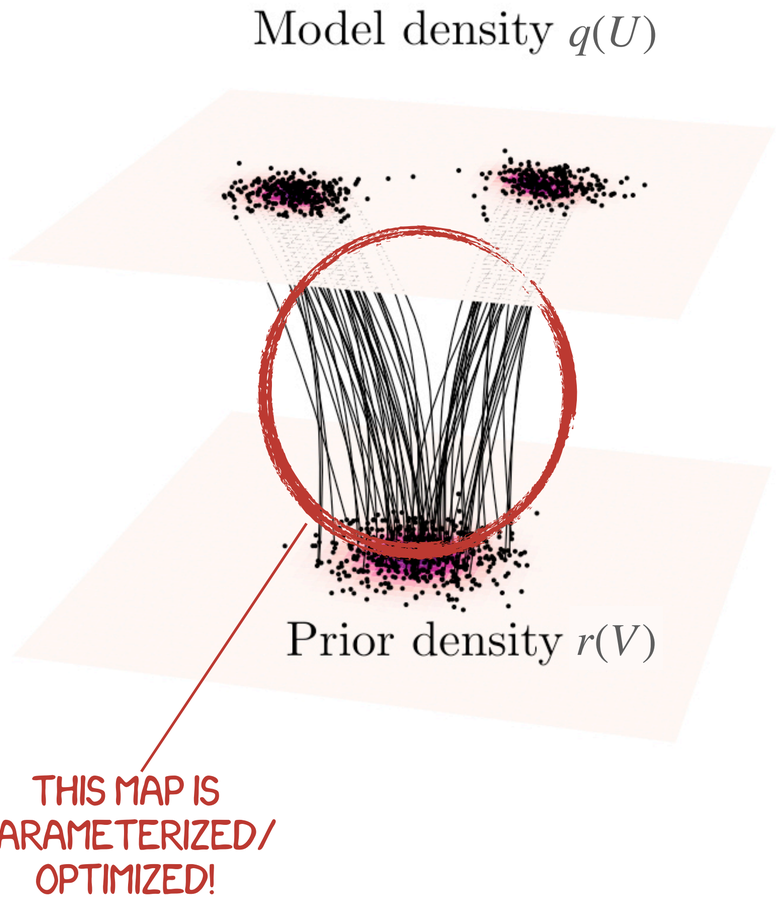


Image credit: Eric Jang, Blog post (2019)

Measure transport strategies

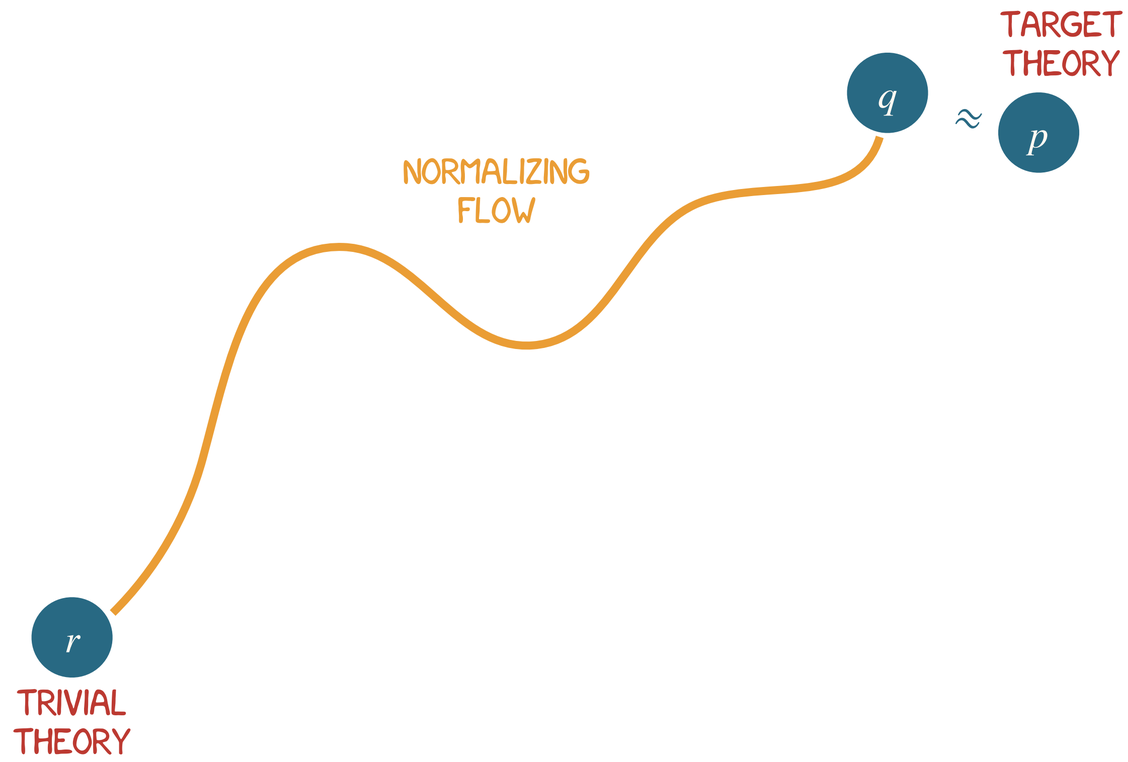
Various possible paths through space of distributions:



Measure transport strategies

Various possible paths through space of distributions:

- **Freely learnable**
  Rezende & Mohamed. PMLR 37 (2015)
  - Normalizing flow models

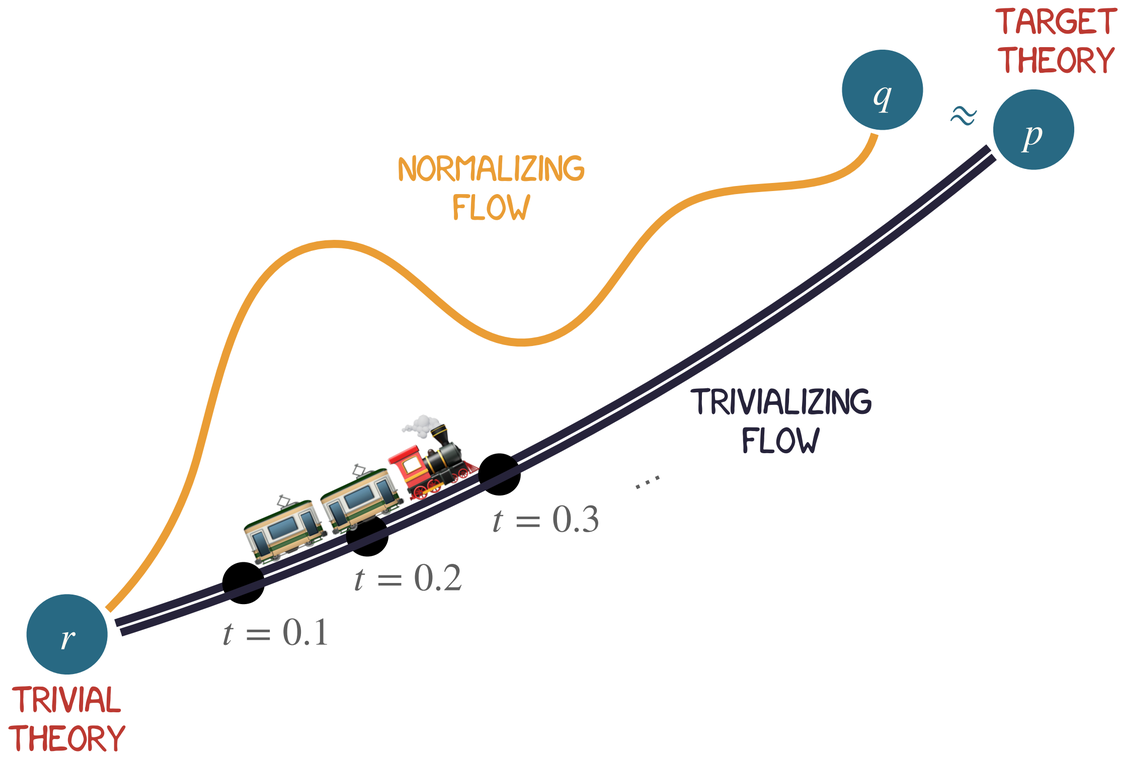


Measure transport strategies

Various possible paths through space of distributions:

- **Freely learnable**
  Rezende & Mohamed. PMLR 37 (2015)
  - Normalizing flow models

- **Explicit fixed path**
  Neal. physics/9803008 (1998); Lüscher. CMP293 (2010)
  - Annealing $S_t(U)$ for $t \in [0,1]$
  - Lüscher's trivializing flow, out-of-equilibrium simulation

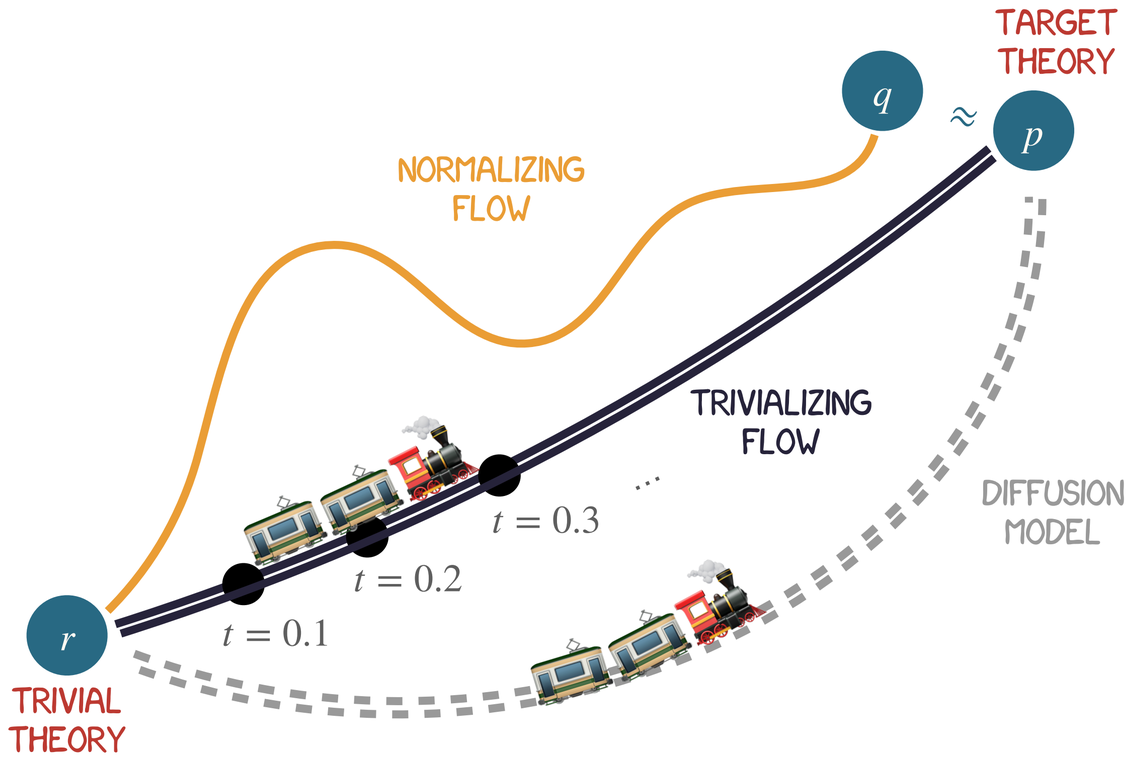


Measure transport strategies

Various possible paths through space of distributions:

- **Freely learnable**
  Rezende & Mohamed. PMLR 37 (2015)
  - Normalizing flow models
- **Explicit fixed path**
  Neal. physics/9803008 (1998); Lüscher. CMP293 (2010)
  - Annealing $S_t(U)$ for $t \in [0,1]$
  - Lüscher's trivializing flow, out-of-equilibrium simulation
- **Implicit fixed path**
  - Defined by action on samples
  - Diffusion models, flow matching

    Song, et al. 2011.13456 (2020)
    Albergo, et al. 2209.15571 (2022)
    Lipman, et al. 2210.02747 (2022)



Continuous-time transport

Flows (deterministic)

ODE for samples:

$$\partial_t U = v_t(U)$$

PDE on density:

$$\partial_t p_t(U) = -\nabla \cdot \{v_t(U)\, p_t(U)\}$$

(continuity equation)

Continuous-time transport

ℹ️ Brownian noise:
$\langle \eta_t^a \eta_{t'}^b \rangle = \delta(t - t')\delta^{ab}$

Flows (deterministic)

ODE for samples:

$$\partial_t U = v_t(U)$$

PDE on density:

$$\partial_t p_t(U) = -\nabla \cdot \{v_t(U)\, p_t(U)\}$$

(continuity equation)

Diffusions (stochastic)

SDE for samples:

$$\partial_t U = v_t(U) + \eta_t^a T^a$$

PDE on density:

$$\partial_t p_t(U) = -\nabla \cdot \{v_t(U)\, p_t(U)\} + \frac{1}{2}\nabla^2 p_t(U)$$

(Fokker-Planck equation)

Learned measure transport

Flows (deterministic)

ODE for samples:

$$\partial_t U = v_t(U)$$

Diffusions (stochastic)

SDE for samples:

$$\partial_t U = v_t(U) + \eta_t^a T^a$$

In both cases need to **parameterize a "flow field"**

$$v : \mathbb{R} \times \mathcal{G}^{n_{\text{link}}} \to \mathfrak{g}^{n_{\text{nlink}}}$$

Learned measure transport

Flows (deterministic)

ODE for samples:

$$\partial_t U = v_t(U)$$

Diffusions (stochastic)

SDE for samples:

$$\partial_t U = v_t(U) + \eta_t^a T^a$$

In both cases need to **parameterize a "flow field"**

$$v : \mathbb{R} \times \mathcal{G}^{n_{\text{link}}} \to \mathfrak{g}^{n_{\text{nlink}}}$$

Several options:

- Analytical approximation (Lüscher's trivializing flow)
- Ansatz in physical basis (e.g. Wilson loops)
- **Use a neural network**

See e.g.:
Lüscher. CMP293 (2010)
Engel, Schaefer. CPC182 (2011)
Bacchio, et al. PRD107 (2023)

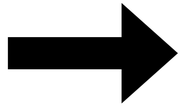
Exactness

The model is never perfect $q(U) \neq p(U)$, but we can exactly correct it if…

For each U_i drawn from the model, we know $q(U_i)$ and $p(U_i)$

Model should provide this.

Known in terms of the lattice action.



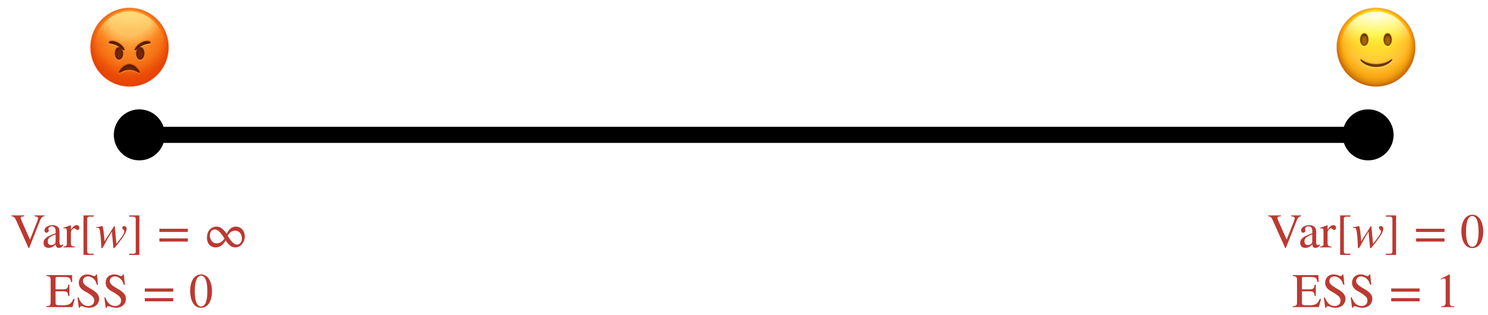
Metropolis or reweighting

$$\langle \mathcal{O} \rangle_p = \frac{\langle \mathcal{O}(U)\, p(U)/q(U) \rangle_q}{\langle p(U)/q(U) \rangle_q}$$

Measuring effectiveness of transport

Variance of reweighting factors $\mathrm{Var}[w]$, where $w(U) := p(U)/q(U)$

Effective sample size, $\mathrm{ESS} := \dfrac{1}{\langle w^2 \rangle_q} = \dfrac{1}{1 + \mathrm{Var}[w]}$





Explicit paths: Annealing methods

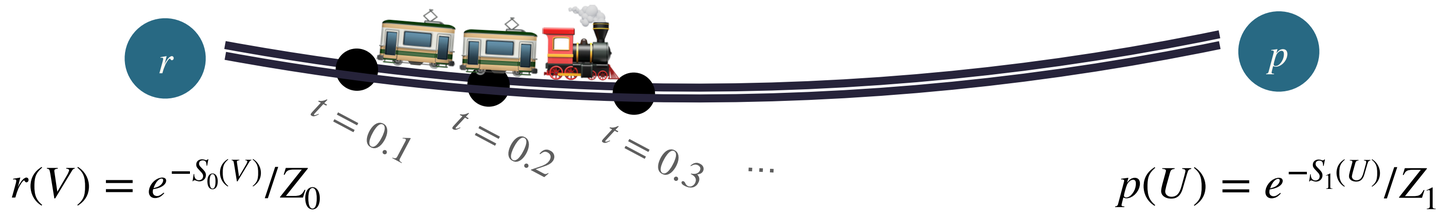
(or “non-equilibrium methods”)

Jarzynski. PRL78 (1997)
Neal. physics/9803008 (1998)
Lüscher. CMP293 (2010)

Annealing paths

Choose any continuous sequence of actions S_t connecting an initial theory (t=0) to a target theory (t=1).

- Usually choose trivial S_0
- Often simple interpolation $S_t = tS_1 + (1-t)S_0$

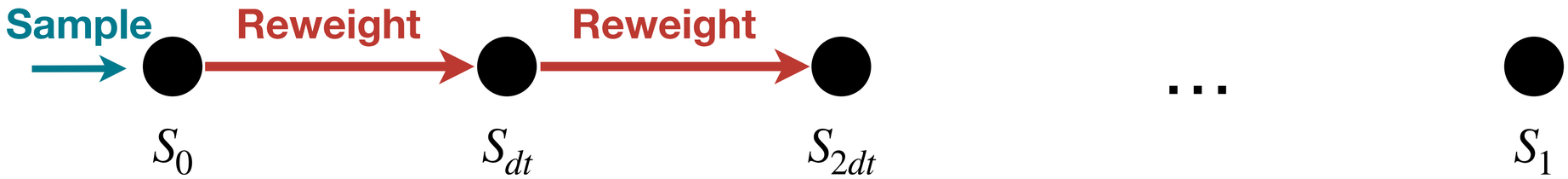


What does this buy you?

Specifying the annealing path gives a zero-noise limit (the quasi-static limit).

Annealed importance sampling:

Jarzynski. PRL78 (1997); Neal. physics/9803008 (1998)



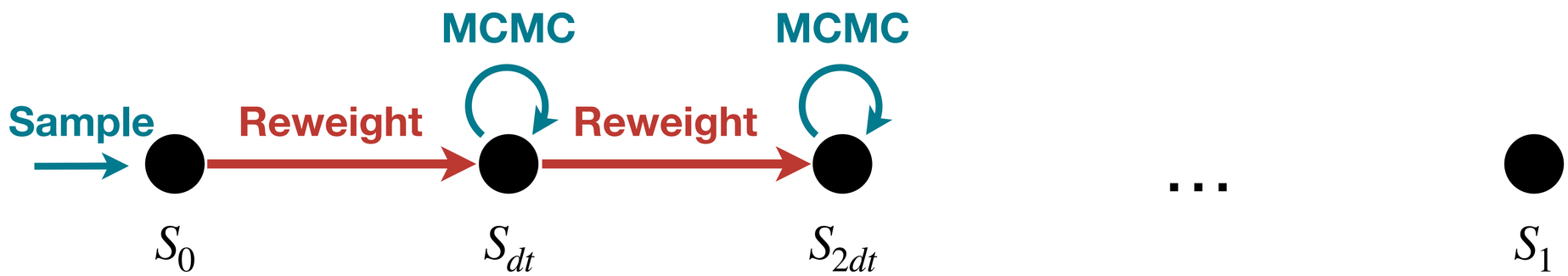
- Unbiased estimates $\langle O(x) \rangle_t = \langle O(x_t) w_t \rangle_{\text{path}}$ throughout the path

What does this buy you?

Specifying the annealing path gives a zero-noise limit (the quasi-static limit).

Annealed importance sampling:

Jarzynski. PRL78 (1997); Neal. physics/9803008 (1998)



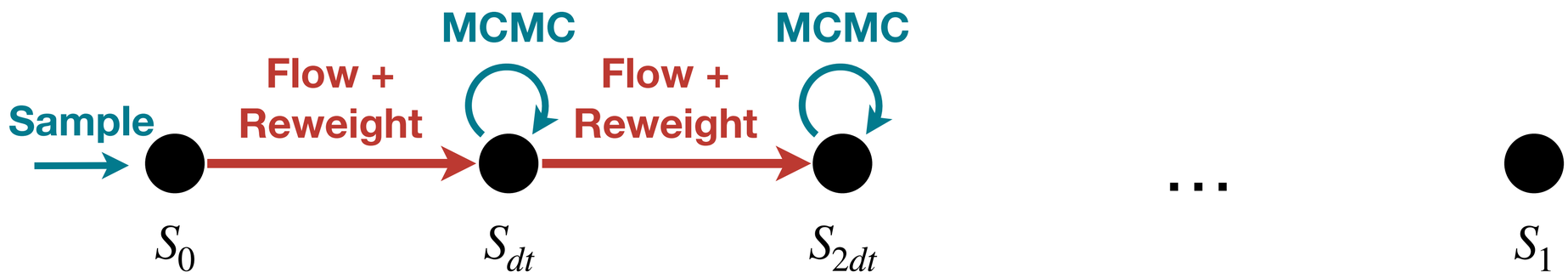
- Unbiased estimates $\langle O(x) \rangle_t = \langle O(x_t) w_t \rangle_{\text{path}}$ throughout the path
- **Decorrelated weights:** $\log w = \sum \log \Delta w_i$ ➡ $\text{Var}[\log w_t] \sim \sum \text{Var}[\log \Delta w_i] \sim 1/n_{\text{step}}$

Learned transport to improve prefactors

Reweighting noise can be reduced by adding learned transport.

Deterministic transport field should satisfy continuity equation

$$\partial_t p_t = -\nabla \cdot (v_t p_t)$$



- Train using path KL (“stochastic normalizing flows”) or using PDE (“NETS”)

Wu, et al. 2002.06707 (2020)

Albergo & Vanden-Eijnden. 2410.02711 (2024)

Annealing summary

- Explicit path through distribution space S_t
- Difficult to learn the transport (often)
- Exactness via weights:
  - Zero-noise quasi-static limit
  - Zero-noise training limit

Implicit paths: Diffusion models



Diffusion paths

Define a path of distributions implicitly by the action of an SDE:

$$\partial_\xi U = f_\xi(U) + g_\xi \eta_\xi^a T^a$$

Image credit: Song et al, *ICLR* (2021)

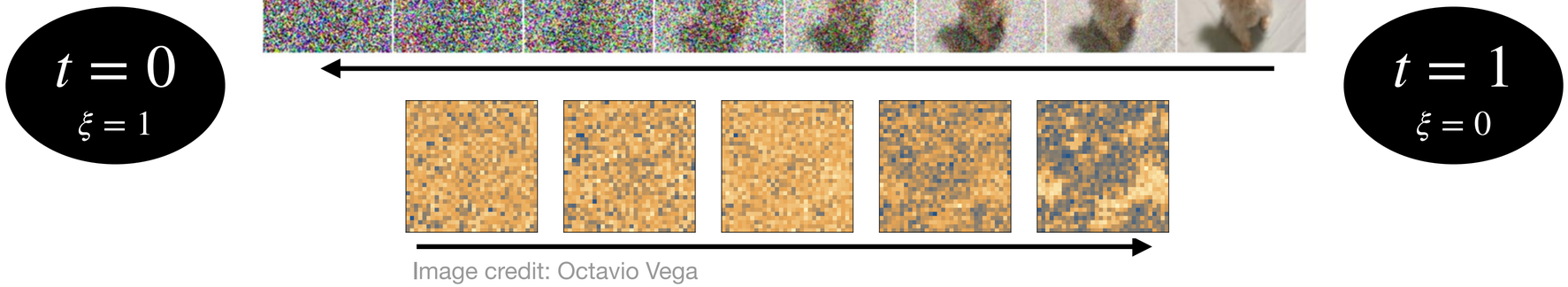


Image credit: Octavio Vega

Diffusion paths

Define a path of distributions implicitly by the action of an SDE:

$$\partial_\xi U = f_\xi(U) + g_\xi \eta_\xi^a T^a$$

Image credit: Song et al, *ICLR* (2021)

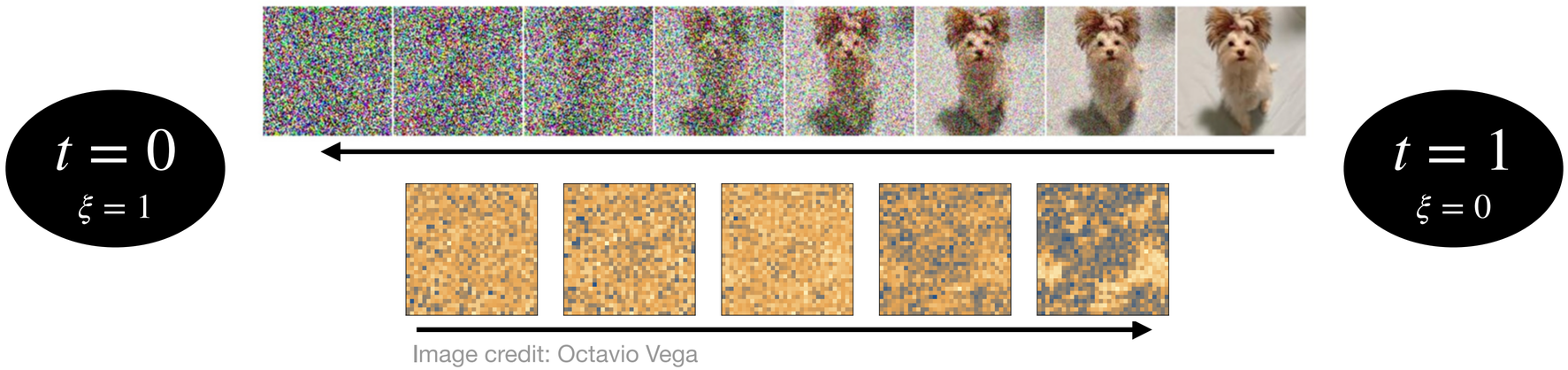


Image credit: Octavio Vega

Probably densities follow Fokker-Planck!

$$\partial_\xi p_\xi = -\nabla \cdot (f_\xi p_\xi) + \frac{1}{2} g_\xi^2 \nabla^2 p_\xi \quad \Longleftrightarrow \quad \partial_t p_t = \nabla \cdot (f_t p_t) - \frac{1}{2} g_t^2 \nabla^2 p_t$$

Diffusion paths

Define a path of distributions implicitly by the action of an SDE:

$$\partial_\xi U = f_\xi(U) + g_\xi \eta_\xi^a T^a$$

Image credit: Song et al, *ICLR* (2021)

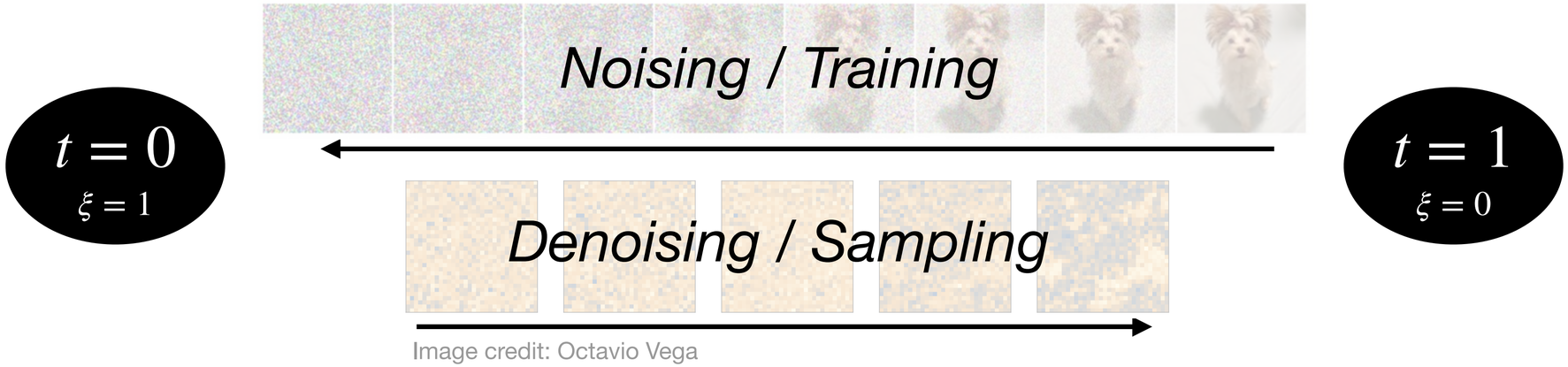


Image credit: Octavio Vega

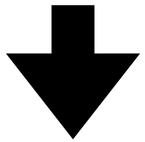
Probably densities follow Fokker-Planck!

$$\partial_\xi p_\xi = -\nabla \cdot (f_\xi p_\xi) + \frac{1}{2} g_\xi^2 \nabla^2 p_\xi \quad \Longleftrightarrow \quad \partial_t p_t = \nabla \cdot (f_t p_t) - \frac{1}{2} g_t^2 \nabla^2 p_t$$

Sampling using the reverse process

We can reproduce the denoising Fokker-Planck equation using transport:

$$\partial_t p_t = \nabla \cdot (f_t p_t) - \frac{1}{2} g_t^2 \nabla^2 p_t \stackrel{?}{=} -\nabla \cdot (v_t p_t)$$

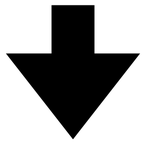


$$\partial_t U = v_t(U) \equiv -f_t(U) + \frac{1}{2} g_t^2 \nabla \log p_t(U)$$

Sampling using the reverse process

We can reproduce the denoising Fokker-Planck equation using transport:

$$\partial_t p_t = \nabla \cdot (f_t p_t) - \frac{1}{2} g_t^2 \nabla^2 p_t \stackrel{?}{=} -\nabla \cdot (v_t p_t)$$



$$\partial_t U = v_t(U) \equiv -f_t(U) + \frac{1}{2} g_t^2 \nabla \log p_t(U)$$

(Can also add noise and counteract with more drift.)

$$\partial_t U = -f_t(U) + (\frac{1}{2} g_t^2 + \frac{1}{2} \hat{g}_t^2) \nabla \log p_t + \hat{g}_t \eta_t^a T^a$$

Score-matching

Sampling the denoising process is easy, but we need $\overbrace{\nabla \log p_t}^{\text{“SCORE”}}$ *…*

$$\partial_t U = v_t(U) \equiv -f_t(U) + \frac{1}{2} g_t^2 \nabla \log p_t(U)$$

Use machine learning:

- Regression to unbiased, noisy estimator for score
- Option 1: denoising-score identity $\nabla \log p_t = \mathbb{E}[\nabla \log p_t(V \mid U)]_{U|V}$ Song, et al. 2011.13456 (2020)
- Option 2: target-score identity $\nabla \log p_t = \mathbb{E}[\nabla \log p_1(U)]_{U|V}$ De Bortoli, et al. 2402.08667 (2024) Phillips, et al. 2402.06320 (2024)
- Later numerical examples: Option 1 for small t, Option 2 for large t

Aside: SU(N) score matching

Unbiased DSI estimator of the score using the **heat kernel** $K_t(U \mid V)$:

$$\nabla \log p_t(U) = \mathbb{E}_{V|U}[\nabla \log K_t(U \mid V)]$$

Using the heat kernel on the SU(N) manifold, we get a usable loss function!

$$\mathcal{L} = \int_0^1 \mathbb{E}_{U_0 \sim p_0} \mathbb{E}_{V \sim K_t} \left[\sigma_t^2 \left\| s_t(U) - P^\dagger \operatorname{diag}[\nabla \log K_t(\theta)] P \right\|^2 \right] dt.$$

Lou, et al. 2310.20030 (2023)
GK & Vega. 2512.19877 (2025)
Aarts, et al. 2601.19552 (2026)
Komijani, et al. 2605.06134 (2026)

Exactness

By default no reweighting factors with denoising process. 😡

Several proposed solutions:

- Treat denoising process as a super-efficient thermalization for MCMC.
  Aarts, et al. 2601.19552 (2026)

- Use path weights referencing reverse diffusion path.
  Zhang, et al. 2505.21005 (2025)
  Wang. 2607.21436 (2026)
  Chen, et al. 2606.13790 (2026)
  B. LUCINI (MON 15:45)

- This talk: Connect diffusion to annealing.

Diffusion model summary

- Implicit path through distribution space (Fokker-Planck)
- Easy to train with regression loss
- Exactness via path weights
  - Zero-noise training limit
- Exactness via annealing
  - Zero-noise training limit
  - Zero-noise quasi-static limit

From implicit to explicit: Annealed diffusion models

Work in progress with Octavio Vega



Octavio Vega
(UIUC)

Overview

Instead of learning the score, learn the density itself (energy-based model).

- Parameterize $\hat{p}_t \sim \exp(-\hat{E}_t)$, score-matching on $-\nabla \hat{E}_t$
- Fix endpoints $\hat{E}_0 = S_0$ and $\hat{E}_1 = S_1$

OuYang, et al. 2409.09787 (2024)
Shi, et al. 2410.15336 (2024)
Habibi, et al. (LATTICE2025)

J. LI (TUE 15:45)

Overview

Instead of learning the score, learn the density itself (energy-based model).

- Parameterize $\hat{p}_t \sim \exp(-\hat{E}_t)$, score-matching on $-\nabla \hat{E}_t$
- Fix endpoints $\hat{E}_0 = S_0$ and $\hat{E}_1 = S_1$

OuYang, et al. 2409.09787 (2024)
Shi, et al. 2410.15336 (2024)
Habibi, et al. (LATTICE2025)
J. LI (TUE 15:45)

Why is this useful?

Related ideas explored in:
Mate, et al. 2406.02313 (2024)
Mate, et al. 2410.15815 (2024)
Phillips, et al. 2402.06320 (2024)
Ou, et al. 2505.22524 (2025)
Lee & Jarzynski. 2606.30451 (2026)

1. Annealed sampling along path (reweight + MCMC) — zero-noise quasi-static limit!
2. $\hat{E}_t$ close to diffusion path after training — automatically has a *good transport field*!

$$\underbrace{\partial_t \hat{p}_t \approx \nabla \cdot (f_t \hat{p}_t) - \frac{1}{2} g_t^2 \nabla^2 \hat{p}_t}_{\text{DIFFUSION FPE}} \implies \underbrace{\partial_t \hat{p}_t \approx -\nabla \cdot (v_t \hat{p}_t), \quad v_t \equiv -f_t - \frac{1}{2} g_t^2 \nabla \hat{E}_t}_{\text{ANNEALED TRANSPORT}}$$

Algorithm

1. Parameterize energy function $\hat{E}_t$.
2. Optimize $\hat{E}_t$ using score-matching loss.
3. Sample with n_{step} annealing steps, using flow $v_t = -f_t - \frac{1}{2} g_t^2 \nabla \hat{E}_t$

- Apply MCMC with respect to $\hat{E}_t$
- Flow from current t to $t + dt$, accumulate log-weight

$$\Delta \log w = -\hat{E}_{t+dt} + \hat{E}_t - \int_t^{t+dt} dt' \nabla \cdot v_t$$

REPEAT

Benchmark: free scalar field theory

Solvable via Fourier transform.

$$S(\tilde{\phi}) = \sum_k (k^2 + m^2) \, |\, \tilde{\phi}(k) \,|^2 \Rightarrow p(\tilde{\phi}) \sim \prod_k e^{-(k^2+m^2)|\tilde{\phi}(k)|^2}$$

Diffusion SDE:

$$\partial_t \tilde{\phi}_t(k) = -\tilde{\phi}_t(k) + g_t \eta_t, \quad \Longrightarrow \quad \tilde{\phi}_t(k) = \alpha_t \tilde{\phi}_0(k) + \sigma_t \theta, \quad \theta \sim \mathcal{N}(0, \mathbb{I})$$

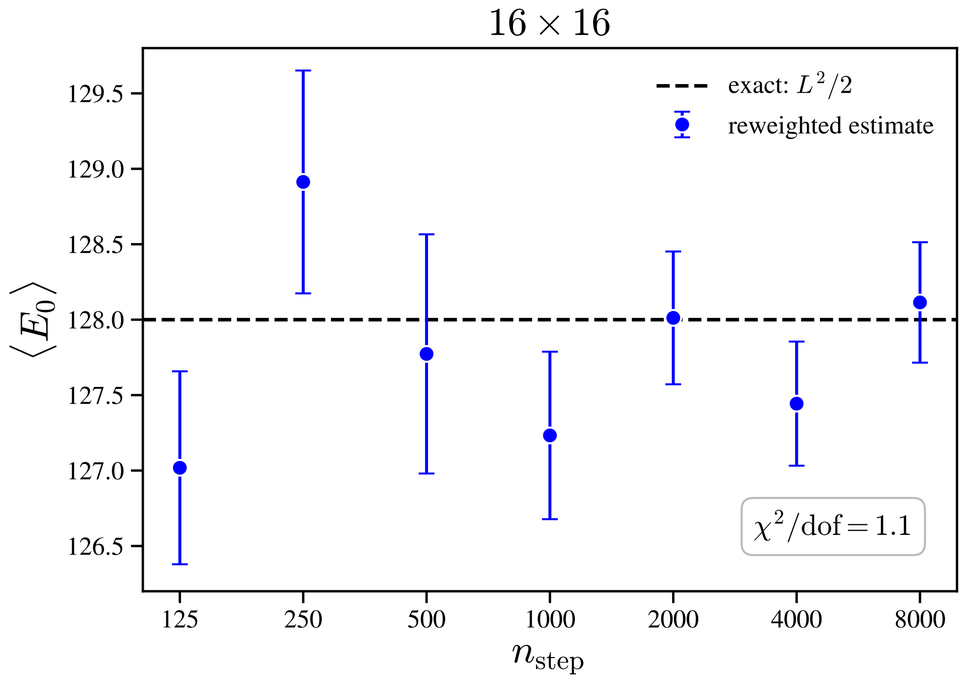
Exact solution to FPE (used for cross-checks):

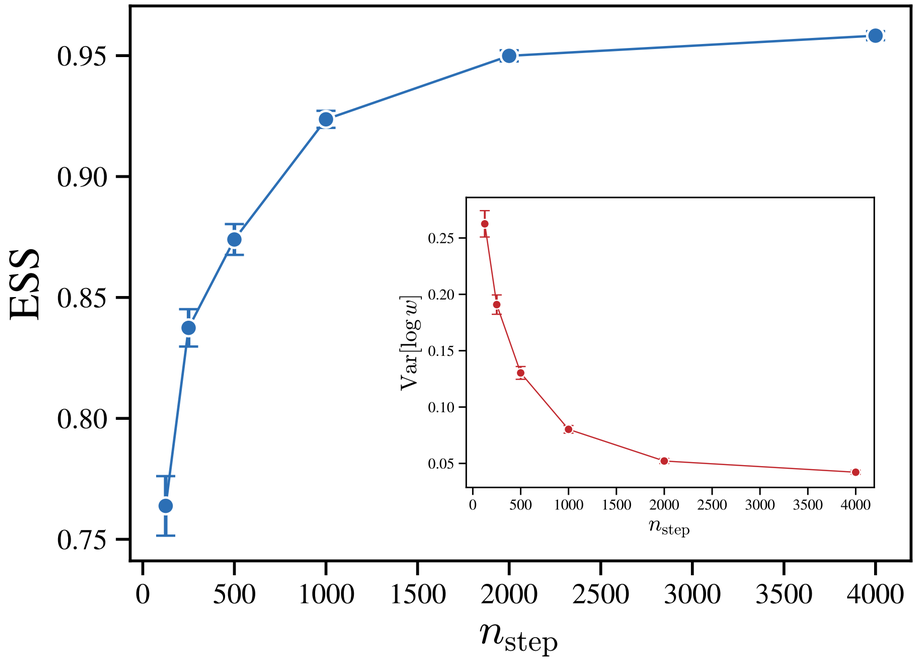
$$E_t = \frac{1}{2} \frac{k^2 + m^2}{\alpha_t^2 + \sigma_t^2 (k^2 + m^2)} \, |\, \tilde{\phi}_t(k) \,|^2$$

Free scalar theory

Local energy function trained via score matching

$$\hat{E}_t = \sum_x \hat{E}_x^{\text{local}}, \quad \hat{E}_x^{\text{local}} \text{ computed from patch of } \phi \text{ around } x$$

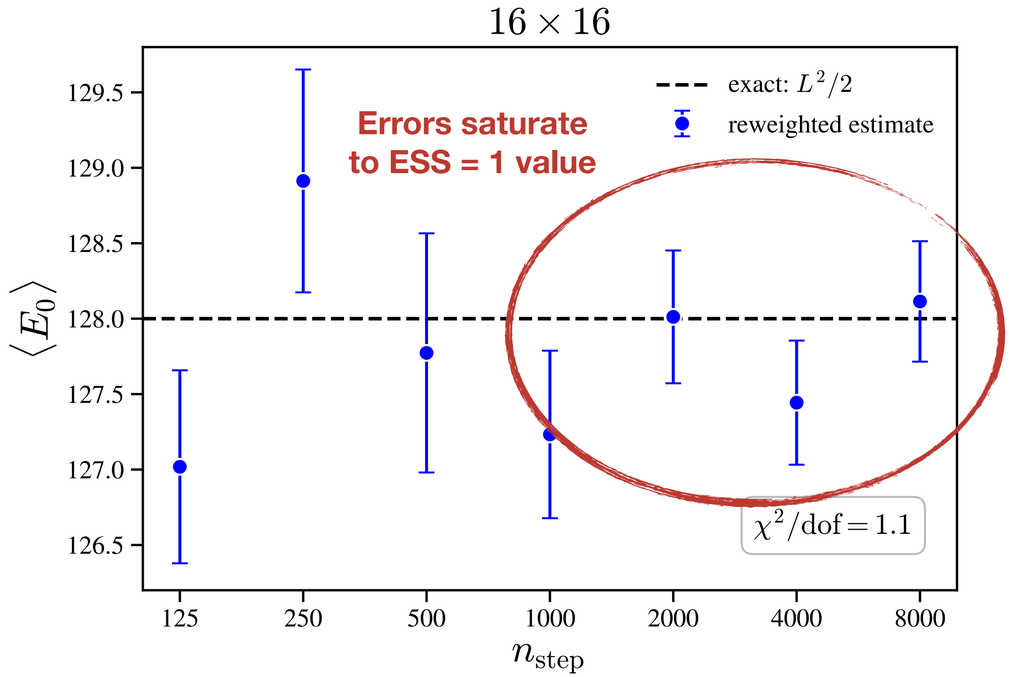


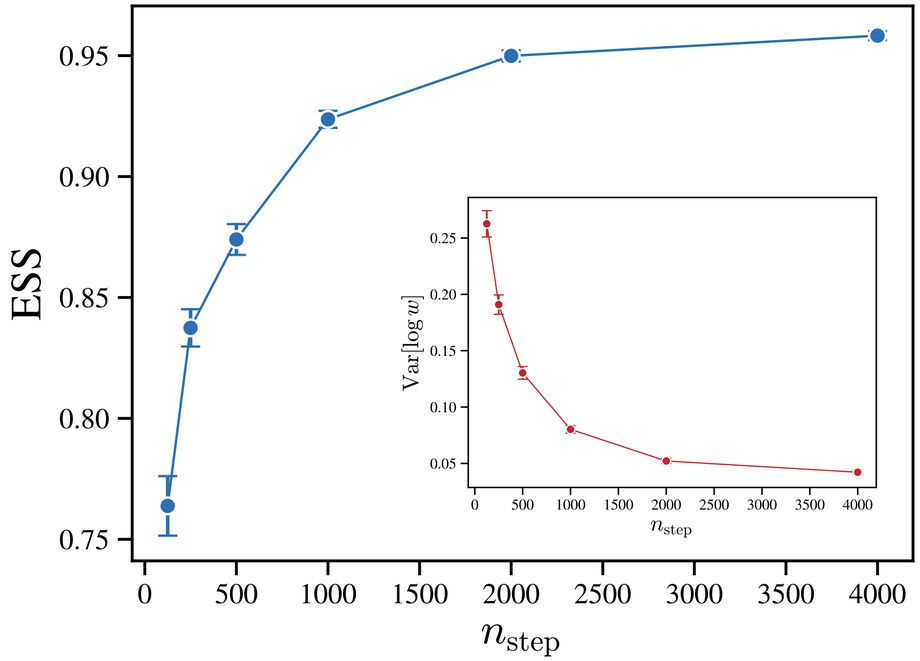


Free scalar theory

Local energy function trained via score matching

$$\hat{E}_t = \sum_x \hat{E}_x^{\text{local}}, \quad \hat{E}_x^{\text{local}} \text{ computed from patch of } \phi \text{ around } x$$

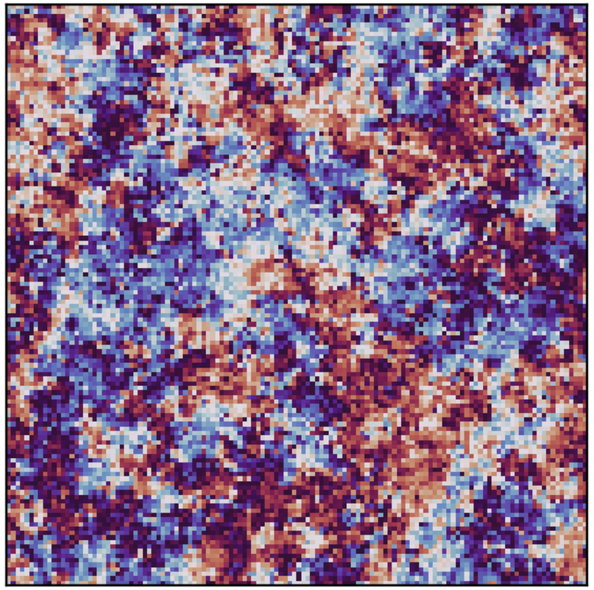




2D O(3) non-linear sigma model

$$S(\phi) = -\beta \sum_{x} \sum_{\mu} \vec{\phi}(x) \cdot \vec{\phi}(x+\hat{\mu}), \quad \vec{\phi}(x) \in \mathbb{S}^2$$

- Asymptotic freedom + topological sectors
- Compact geometry
  - SDE needs to be specialized for O(3)
  - Uniform prior distribution $S_0 = 0$

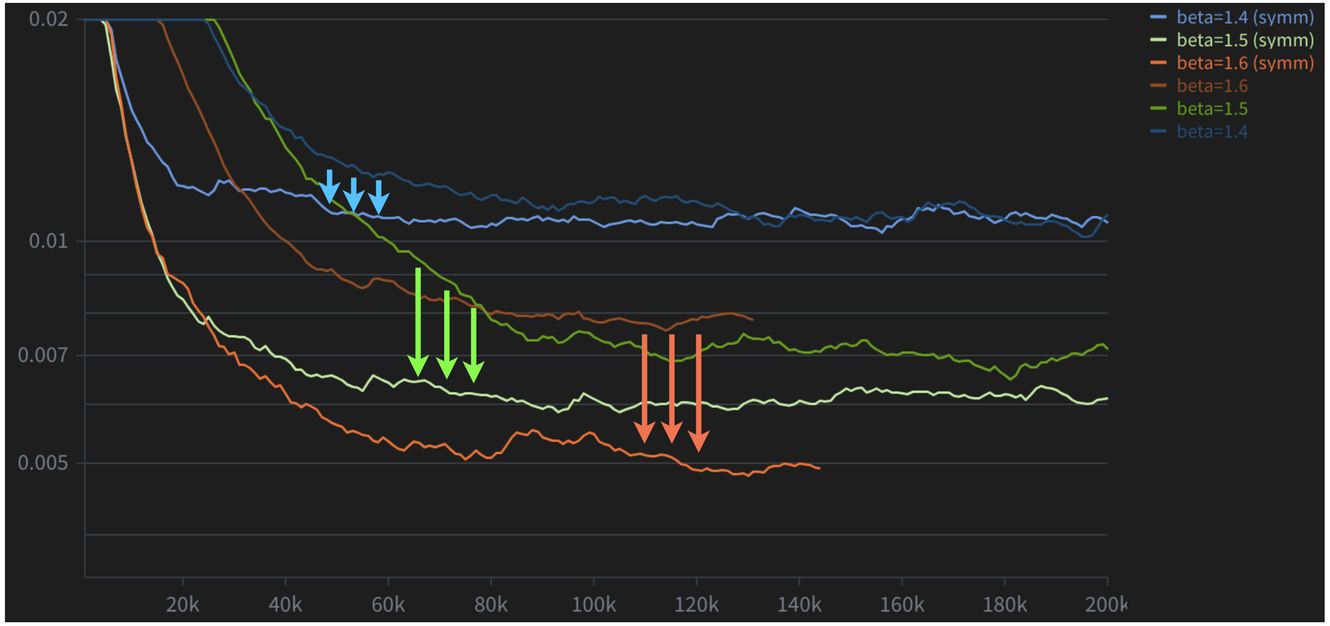


Symmetries

In practice, O(3)-symmetric actions perform better.

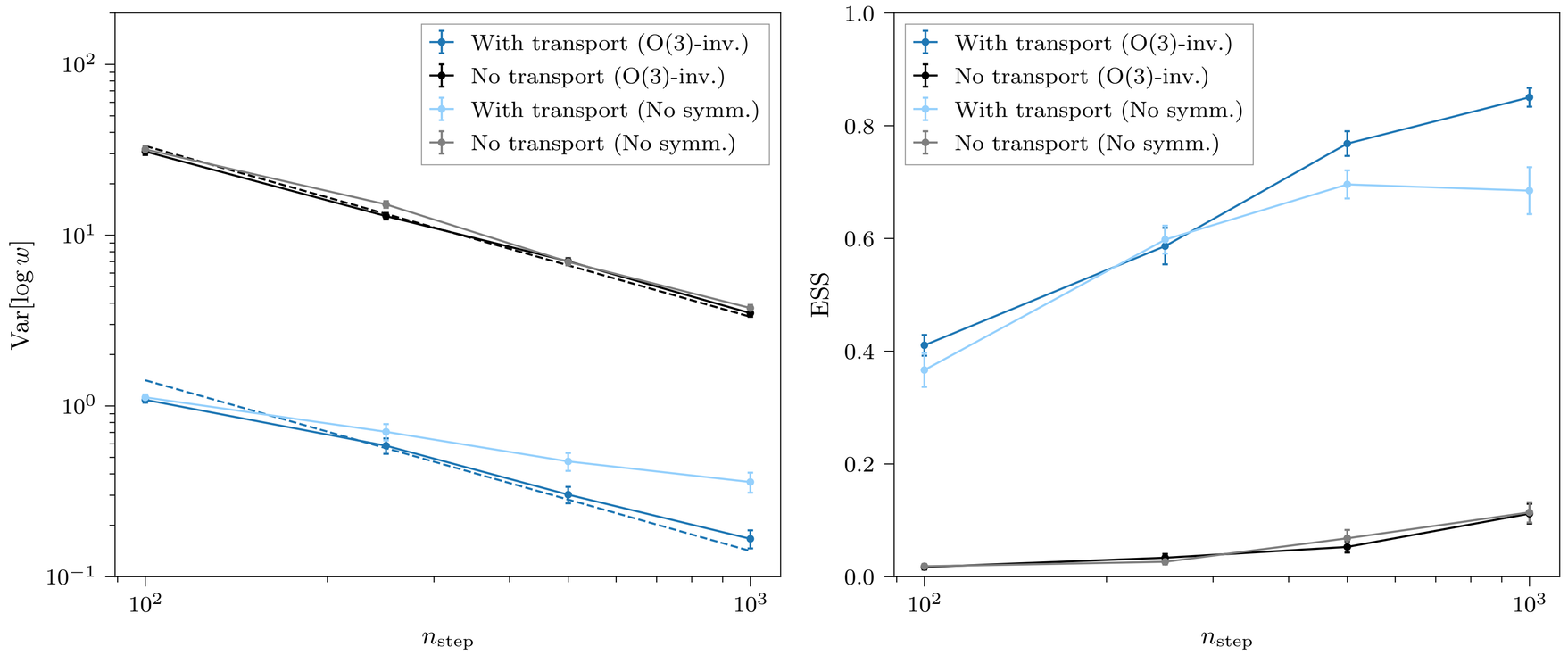
- Parameterized by passing invariants $\phi_x \cdot \phi_y$
- More effect at smaller lattice spacing / more lattice sites.

Error between noising and denoising diffusion paths



Zero-noise quasi-static limit

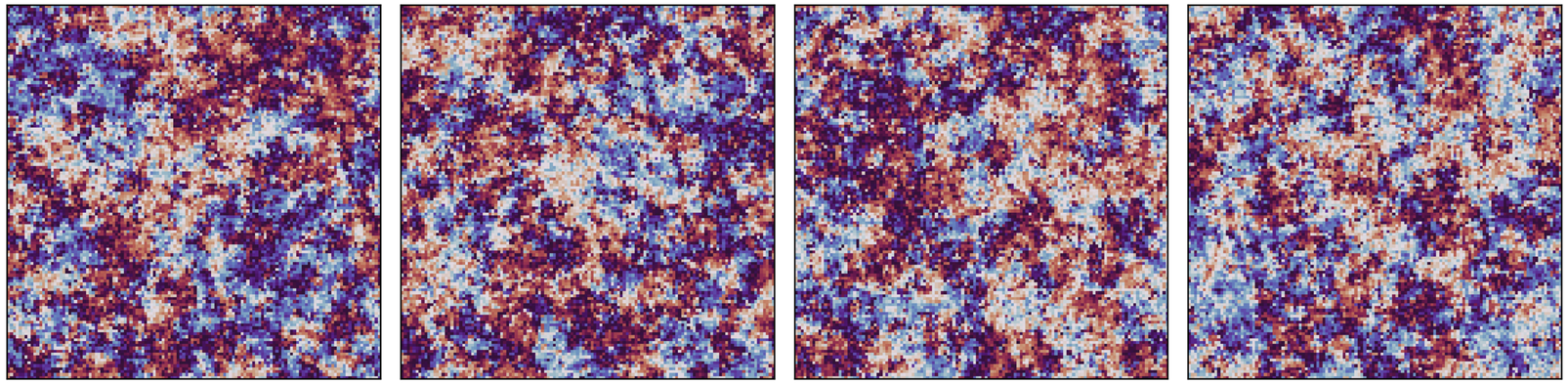
Reweighting noise always decreases with n_{step} and added transport greatly improves the prefactor.



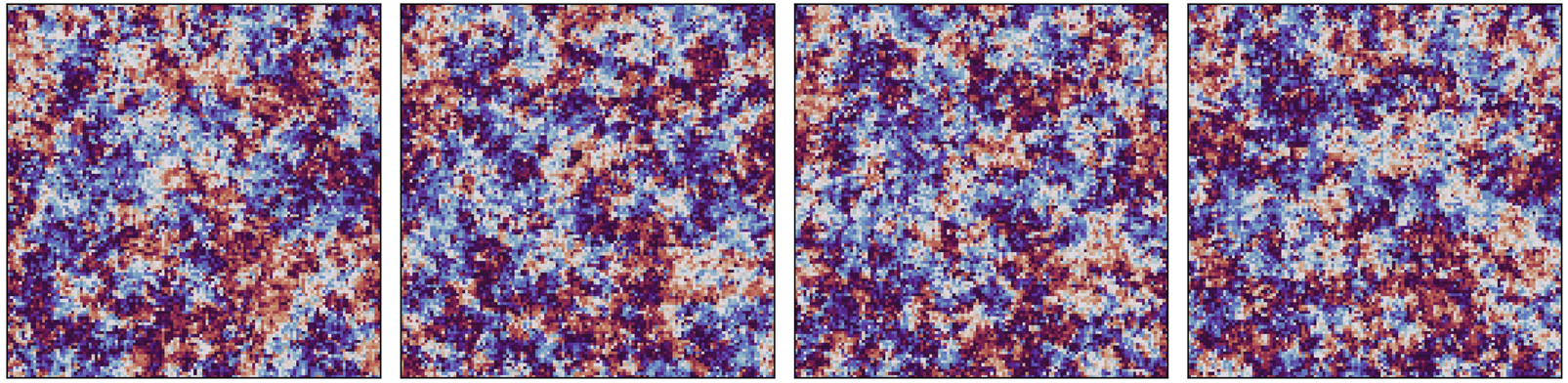
Volume transfer

Models trivially transfer to larger volumes.

DIFFUSION



MCMC



- **Trained at** $L = 16$,
- **Evaluated at** $L = 128$
- **ESS ~ 3% @** $n_{\text{step}} = 250$
- **ESS ~ 15% @** $n_{\text{step}} = 5000$

Caveats and open questions

1. Energy-based models can be harder to train.
2. Constrained endpoints necessary for true zero-noise limit.
3. **Does not address volume scaling of RW factors.**
   - Need new ideas for this. See talks on RG approaches!

A. HASENFRATZ (TUE 9:30) A. SINGHA (TUE 15:45)
E. CELLINI (TUE 11:30) Y. ASHIDA (THUR 9:30)

4. Important to use flows with exact Jacobian (use good integrator).
   - Zero-noise scaling is weaker when using stochastic div/jac estimators?

Conclusions & Outlook

- Learned transport has the potential to **solve critical slowing down**

  ✓ Could gain precision, **better systematic control**

- Sampling problem is not solved yet, but…

  ✓ **Reusable components** (gauge fields + fermions)

  ✓ **New variance reduction** method

  ✓ **New directions** including annealing, diffusion

Conclusions & Outlook

- Learned transport has the potential to **solve critical slowing down**

  ✓ Could gain precision, **better systematic control**

- Sampling problem is not solved yet, but…

  ✓ **Reusable components** (gauge fields + fermions)

  ✓ **New variance reduction** method

  ✓ **New directions** including annealing, diffusion



Elia Cellini



Satria Widyanto



Octavio Vega

1. UNBIASED SAMPLING WITH NOISE
2. SCALABLE ANNEALING METHODS
3. RENORMALIZATION GROUP

Backup slides

Neural networks for drift fields

Neural networks provide $f_\omega : \mathbb{R}^n \to \mathbb{R}^m$.

Learned force field: $\mathscr{F} : \mathbb{R} \times \mathscr{G}^{n_{\text{link}}} \to \mathfrak{g}^{n_{\text{nlink}}}$

- Embed $\mathscr{G}^{n_{\text{link}}} \subset \mathbb{R}^{n-1}$ for sufficiently large n
- Use isomorphism $\mathfrak{g}^{n_{\text{link}}} \simeq \mathbb{R}^m$ (e.g. $\mathfrak{g} = \mathfrak{su}(N_c) \implies m = n_{\text{link}}(N_c^2 - 1)$)

Learned potential: $\mathscr{F}_t(U) = \nabla \varphi_t(U)$, where $\varphi : \mathbb{R} \times \mathscr{G}^{n_{\text{link}}} \to \mathbb{R}$

- Restricts the kinds of $\mathscr{F}_t$ we can have, but usually not meaningfully (zero curl is OK)
- Embed $\mathscr{G}^{n_{\text{link}}} \subset \mathbb{R}^{n-1}$ in the same way