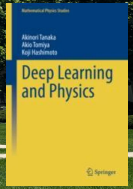
Based on arXiv:2607.12587 + updates

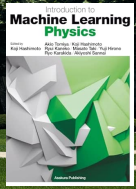
Self-learning Diffusion model and recent developments

Akio Tomiya
Associate Professor, Tokyo Woman's Christian University
Specially Appointed Assoc. Prof., Kyoto Univ. (BOOST, concurrent)
Visiting Researcher, RIKEN R-CCS (concurrent)
Contact: akio@yukawa.kyoto-u.ac.jp



julia





Machine learning for lattice sampling

Selected work on learned proposals, gauge covariance, and generation

2016 **1609.09087** Detection of phase transition via convolutional neural network

Phase transition from latent information
Tanaka–Tomiya, submitted 28 Sep 2016.
Almost 10 years ago.

2017 **1712.03893** **Towards reduction of autocorrelation in HMC by machine learning**

2020 **2010.11900** **Self-learning Monte Carlo for non-Abelian gauge theory with dynamical fermions**

2021 **2103.11965** Gauge covariant neural network for quarks and gluons

2021 **2112.01586** HMC with Normalizing Flows

2022 **2208.08903** GomalizingFlow.jl: A Julia package for Flow-based sampling algorithm for lattice field theory

2023 **2306.11527** Self-learning Monte Carlo with equivariant Transformer

2025 **2501.16955** CASK: A Gauge Covariant Transformer for Lattice Gauge Theory

2026 **2602.20516** Lattice Gauge Theory via LLVM-Level Automatic Differentiation

See also Y. Nagai's talk on Wed afternoon.

2026 **2607.12587** Lattice Configuration Generation with a Self-Learning Diffusion Model

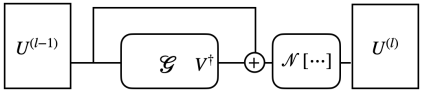
Related review: *Machine Learning for Lattice QCD*, JPSJ 94, 031006 (open access).

Building gauge-covariant neural operators

Neural operators for 4D SU(N) gauge theory with dynamical fermions

Gauge covariance built into each learned map

- Smearing combines a link with nearby paths that have the same endpoints.
- Promote the smearing parameters to trainable weights; stack these maps as network layers.
- Each output remains a gauge-covariant link. A gauge-invariant action can then be built from the output links.



Generalized smearing as a network layer (Fig. 2).

Example: a trainable stout layer

$$U_{x\mu}^{(\ell+1)} = e^{iQ_\theta^{(\ell)}[U^{(\ell)}]_{x\mu}} U_{x\mu}^{(\ell)}.$$

Q_θ: a learned traceless Hermitian matrix, constructed from staples and the local link.

Neural surrogate: a learned effective action

The surrogate generates HMC proposals. The original action enters the Metropolis test.

$$\mathcal{F}_\theta[U^\Omega]_{x\mu} = \Omega_x \mathcal{F}_\theta[U]_{x\mu} \Omega_{x+\hat{\mu}}^\dagger.$$

Gauge-covariant neural operators provide building blocks for learned action surrogates.

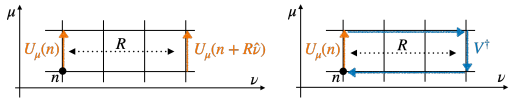
Nagai and Tomiya, *Gauge covariant neural network for quarks and gluons*, arXiv:2103.11965; Fig. 2 and Secs. II–III. Demonstration: 4D SU(2) with dynamical staggered fermions.

CASK (2025): 4D SU(N) gauge theory with dynamical fermions

Covariant Attention with Stout Kernel

- Attention lets the local gauge field determine which paths receive more weight.
- Query, key and value are link fields produced by 3 independently trainable stout maps.
- Gauge-invariant weights come from traced loops connecting the links, not naive dot products of separated links.
- These weights combine value-field staples in a gauge-covariant stout update.

In this implementation: sparse attention using 1×1, 1×2 and 1×3 rectangles.



Connect separated links before taking the trace (Fig. 1).

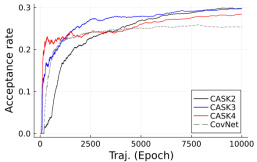


Fig. 2 (left): SU(2), $\beta = 2.7$, with dynamical staggered fermions.

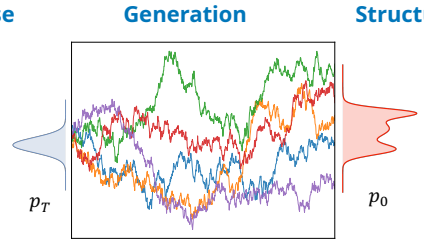
CASK extends our gauge-covariant surrogate architecture with attention.

Nagai, Ohno and Tomiya, *CASK: A Gauge Covariant Transformer for Lattice Gauge Theory*, arXiv:2501.16955, Figs. 1–2, Secs. 2.3–4.

Lattice configuration generation with diffusion

This talk: self-learning diffusion with Metropolis correction

A diffusion model learns stochastic motion that builds structure from noise.



Curves: different stochastic trajectories. Side profiles: probability distributions.

Here, each sample is a lattice field ϕ.

Learned proposals and Metropolis correction target $\pi(\phi) \propto e^{-S[\phi]}$.

Concept: Wang, Aarts and Zhou, *Diffusion Models as Stochastic Quantization in Lattice Field Theory*, arXiv:2309.17082, Fig. 2 (reverse process).

Self-learning sampler: A. Tomiya, arXiv:2607.12587.

Talk outline

Sampling, action-weighted score learning, and the XY model

I **Lattice field theory and HMC**

Target measures, slow modes, HMC, and Metropolis correction

II **Self-learning diffusion: general formulation**

Training steps, generation updates, and action-weighted self-learning

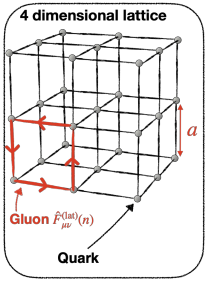
III **XY model: implementation and numerical tests**

Model, training and generation settings, and physical measurements

Summary

I. Lattice field theory and HMC

The computational object is an ensemble of fields



a: lattice spacing.
Red loop: a plaquette.

$$\langle \mathcal{O} \rangle = \frac{1}{Z} \int \mathcal{D}\phi \, \mathcal{O}[\phi] e^{-S[\phi]} \approx \frac{1}{N} \sum_{n=1}^{N} \mathcal{O}[\phi_n].$$

- Integrate over all lattice degrees of freedom.
- A Markov chain supplies correlated field configurations.

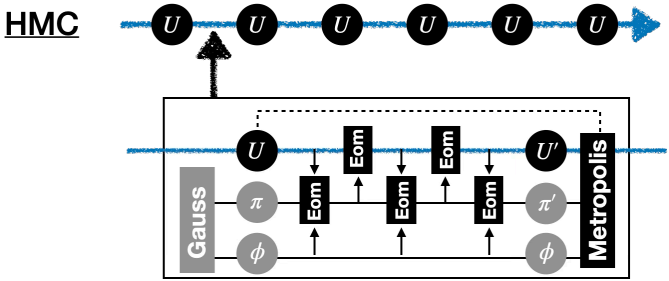
Sampling is part of the physics calculation

The fields ϕ_n must explore the target measure, including rare sectors and long-distance fluctuations.

- Lattice field theory directly evaluates the defining path integral of a regularized quantum field theory, rather than imitating physical phenomena.
- **A first-principles calculation must account for its fluctuations at all scales from UV to IR.**

HMC is the de facto standard for lattice field theory

Hamiltonian trajectories generate global proposals and MH correction preserves the target



Gauge theory with dynamical fermions.

Notation in the diagram

U: gauge links.
π: fictitious momenta.
ϕ: pseudofermion field, fixed during each trajectory.
Eom: equations of motion.

$$H = S_{\mathrm{LQCD}}(U;\phi) + \tfrac{1}{2}\sum_x \pi_x^2.$$

S_{LQCD}: gauge and pseudofermion actions.
The force updates all links.

Reversible, volume-preserving integration and the Metropolis test preserve the exact target distribution. Critical and topological directions can nevertheless remain slow.

HMC: Duane et al., Phys. Lett. B 195 (1987) 216.

Autocorrelation, not acceptance, controls information

The observable chooses the relevant notion of mixing

$$\rho_{\mathcal{O}}(t) = \frac{\langle \delta\mathcal{O}_n \delta\mathcal{O}_{n+t} \rangle}{\langle (\delta\mathcal{O})^2 \rangle},$$

$$\tau_{\mathrm{int},\mathcal{O}} = \frac{1}{2} + \sum_{t>0} \rho_{\mathcal{O}}(t).$$

$$N_{\mathrm{eff},\mathcal{O}} \simeq \frac{N}{2\tau_{\mathrm{int},\mathcal{O}}}.$$

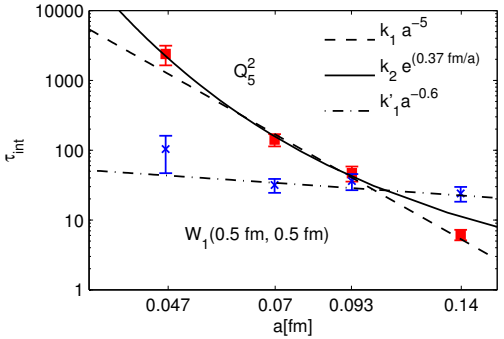
- High acceptance with tiny moves may give large τ_{int}.
- Energy, topology, and long-distance correlators can mix on different scales.
- Acceptance, squared jump distance, and several observable-specific autocorrelations are needed together.

Optimization targets proposal motion; physics validation targets **observable histories**.

Autocorrelation convention in this work: positive, self-consistent $c = 5$ window with block jackknife.

Large systems stress both proposal geometry and statistics

Slow modes can decouple from otherwise well-behaved observables



In DD-HMC, topological charge develops a much longer autocorrelation time as the lattice spacing decreases, while a smeared Wilson loop is far less affected.

- **Exactness**: the invariant distribution must remain the target measure.
- **Efficiency**: proposals must move in the physical directions seen by slow observables.
- **Scaling**: both action mismatch and collective barriers worsen with dimension or volume.

Acceptance can look healthy while one scientifically important mode remains slow.

The goal is therefore **exact sampling** with proposal geometry that can be repaired from data.

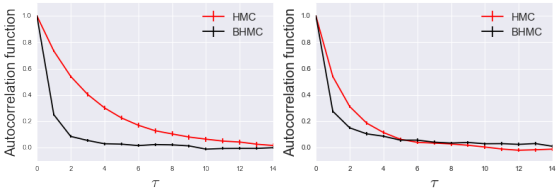
Schaefer, Sommer, Virotta, "Critical slowing down and error analysis in lattice QCD simulations", arXiv:1009.5228, Fig. 4.

See also L. Verzichelli's talk on Tue.

Beyond HMC

Learned collective proposals in HMC

2017 proof of concept — useful motion was visible before the exact formulation



Condensate ACFs in the symmetric and broken phases: the RBM-assisted chain (black) decorrelates faster than HMC (red) in this early study.

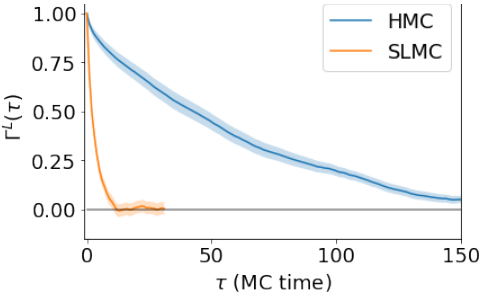
- A neural model supplied a collective update that ordinary HMC did not discover.
- The observable histories, not the loss alone, revealed the improvement.
- The original algorithm was not yet fully corrected with the true action and showed discrepancies near criticality.

The durable idea was the **learned proposal**; exactness still needed a cleaner boundary.

Tanaka–Tomiya, "Towards reduction of autocorrelation in HMC by machine learning", arXiv:1712.03893, Fig. 1.

Self-learning Monte Carlo with dynamical fermions

2020 self-learning Monte Carlo for gauge theory with dynamical fermions



Polyakov-loop ACF: $\tau_{\rm int}^{\rm HMC} = 62(24)$ versus $\tau_{\rm int}^{\rm SLMC} = 4.0(4)$.

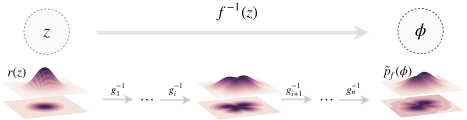
- Learn a cheap effective action from an exact chain.
- Use it only to construct a long-range proposal.
- Accept or reject with the original action.

model error $\longrightarrow$ acceptance and mixing, not bias.

SLDiffusion keeps this separation, but replaces the effective action by a noise-conditioned score and removes target-ensemble supervision.

Nagai–Tanaka–Tomiya, "Self-learning Monte Carlo for non-Abelian gauge theory with dynamical fermions", arXiv:2010.11900, Fig. 1.

The known action trains an invertible generative map



(a) Normalizing flow between prior and output distributions

$$z \sim r(z), \qquad \phi = f_\theta^{-1}(z),$$
$$q_\theta(\phi) = r(f_\theta(\phi)) \left| \det \frac{\partial f_\theta}{\partial \phi} \right|.$$

Learning directly from the action

Target: $\pi(\phi) = Z^{-1} e^{-S(\phi)}$.

$$\mathcal{L}(\theta) = \left\langle \log q_\theta(\phi) + S(\phi) \right\rangle_{q_\theta}$$
$$= D_{\mathrm{KL}}(q_\theta \| \pi) - \log Z.$$

Generate fields with the flow, evaluate S, and differentiate $\mathcal{L}$ to update the network.

Training needs neither target samples nor Z.

Exact sampling

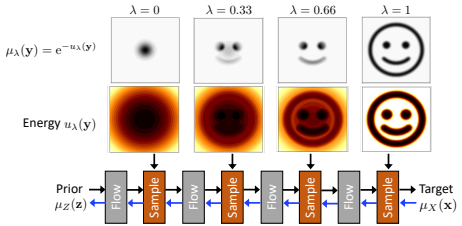
Use the learned density q_θ and the original action S in an independence Metropolis test.

Figure and construction: Albergo, Kanwar and Shanahan, 1904.12072, Fig. 1(a), Eqs. (6), (8), (13)–(14). Scalar fields: Del Debbio et al., 2105.12481, Hackett et al., 2107.00734.
Gauge-equivariant flows: Kanwar et al., 2003.06413, Boyda et al., 2008.05456, Abbott et al., 2305.02402.
Fermions: Albergo et al., 2106.05934. QCD: Abbott et al., 2208.03832.
Stochastic flows: Caselle, Cellini, Nada and Panero, 2201.08862. Continuous flows: Gerdes et al., 2207.00283. All identifiers are arXiv.
Julia implementation: **GomalizingFlow.jl**, Tomiya–Terasaki, arXiv:2208.08903. github.com/AtelierArith/GomalizingFlow.jl.

Stochastic normalizing flows

A. Tomiya

Learned invertible maps combined with stochastic sampling updates



λ interpolates from the simple prior to the target. u_λ is the intermediate action, with density $\mu_\lambda \propto e^{-u_\lambda}$.

Gray: a learned invertible map transports fields.

Orange: MCMC or Langevin updates explore the intermediate distribution.

Wu, Köhler and Noé, *Stochastic Normalizing Flows*, NeurIPS 2020, arXiv:2002.06707, Fig. 2.

Training from the known action

Draw from the prior and generate complete paths. Train the maps by minimizing $-\langle \log w \rangle_{\mathrm{paths}}$. No pre-generated target ensemble is required.

Exact importance weights on paths

The weight w uses the endpoint actions, map Jacobians, and reverse/forward transition ratios.

$$\langle O \rangle_\pi = \frac{\langle w\, O(\phi_{\mathrm{final}}) \rangle_{\mathrm{paths}}}{\langle w \rangle_{\mathrm{paths}}}.$$

Reweighting recovers target expectation values in the infinite-sample limit.

See also talks by E. Cellini and L. Verzichelli on Tue.

Caselle, Cellini, Nada and Panero, *Stochastic normalizing flows as non-equilibrium transformations*, JHEP 07 (2022) 015, arXiv:2201.08862, Eqs. (23)–(27).

Prior work: diffusion sampling in lattice field theory

Scalar fields and gauge theories with learned, Metropolis-corrected proposals

2023: Scalar-field sampling

Wang, Aarts and Zhou, arXiv:2309.17082

Reverse diffusion learns global proposals for 2D ϕ^4 theory from HMC ensembles.

2025: Physics-conditioned $U(1)$ sampling

Zhu et al., arXiv:2502.05504

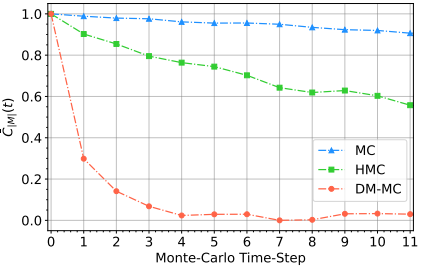
Metropolis-adjusted annealed Langevin algorithm (MAALA): coupling and volume transfer.

2026: Non-Abelian gauge theories

Aarts et al., arXiv:2601.19552

Gauge-equivariant MAALA for 2D $U(2)$ and $SU(2)$.

Shorter correlations in 2D ϕ^4



$|M|$ ACF, 32×32. Wang, Aarts and Zhou, Fig. 8.

Monte Carlo ensembles provide training data for these diffusion-based samplers.

See also T. Ranner's talk on Mon.

Wang, Aarts and Zhou, *Diffusion Models as Stochastic Quantization in Lattice Field Theory*, JHEP 05 (2024) 060, Fig. 8; Zhu et al., arXiv:2502.05504; Aarts et al., arXiv:2601.19552.

II. Self-learning diffusion: general formulation

Diffusion with self-generated training data

Building on G. Aarts, L. Wang and collaborators

We employ the MAALA method. [Zhu, Aarts, W. Wang, Zhou & L. Wang, 2025]

Use the action to learn ahead.

Let the sampler supply the data.

Samples from an accessible reference ensemble.

Action-weighted training for the next β.

MH-corrected samples for the next training stage.

Original MAALA applies MH in the final low-noise region. Here we also apply MH at high noise and repeat the corrected sequence. In both cases, the MH-corrected updates preserve the exact physical target distribution.

Wang, Aarts and Zhou, arXiv:2309.17082. Zhu, Aarts, W. Wang, Zhou and L. Wang, arXiv:2502.05504v2, Sec. 4.2 and Algorithm 2. A. Tomiya, arXiv:2607.12587.

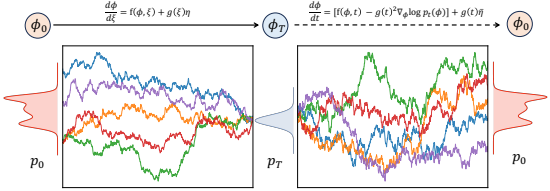
Diffusion model: reverse noising with a learned score

The score supplies a direction field at every noise level

$$dW_t \propto \eta_t, \qquad \eta_t \sim \mathcal{N}(0, I),$$
$$d\phi_t = f(\phi_t, t)\, dt + g(t)\, dW_t,$$
$$d\phi_t = \left[f - g^2 s_\omega(\phi_t, t)\right] dt + g\, d\bar{W}_t,$$
$$s_\omega(\phi_t, t) \simeq \nabla_{\phi_t} \log p_t(\phi_t).$$

1. the forward process destroys structure with known noise;
2. the network learns the score of the noisy distribution;
3. the reverse process uses that score to reconstruct fields.

For lattice sampling, reverse dynamics proposes.
The action supplies the exact correction.



Forward trajectories approach a simple p_T; score-driven reverse trajectories reconstruct p_0.

Wang, Aarts, Zhou, "Diffusion Models as Stochastic Quantization in Lattice Field Theory", arXiv:2309.17082, Fig. 2.

Training and generation update different objects

The score network connects two separate calculations

	Training	Generation
Input	Stored fields, added noise, target coupling	Current field, fixed score, target action
Network evaluation	Predict the score of a noisy field	Compute a drift for a trial move
Updated object	Network parameters ω	Markov-chain configuration ϕ_n
Update rule	Gradient of the score-matching loss	Random proposal and MH accept/reject
Output	A trained score s_{ω_*}	A sequence of corrected configurations

$$\underbrace{\omega_j \longmapsto \omega_{j+1}}_{\text{one training step}} \qquad \underbrace{\phi_n \longmapsto \phi_{n+1}}_{\text{one generation update, } \omega_* \text{ fixed}} .$$

Self-learning uses the generated configurations as input to a later training stage.

Song and Ermon, arXiv:1907.05600. A. Tomiya, arXiv:2607.12587.

Denoising score matching at a fixed target coupling β

Training example

$$\phi_i \sim p_\beta, \qquad \widetilde{\phi}_i = \phi_i + \sigma_i \eta_i, \qquad \eta_i \sim \mathcal{N}(0, I).$$

$$\underbrace{s_\omega(\widetilde{\phi}_i; \beta, \sigma_i)}_{\text{network prediction}} \qquad \underbrace{-\eta_i/\sigma_i}_{\text{known noise target}} .$$

Minibatch loss

$$\widehat{\mathcal{L}}_B(\omega) = \frac{1}{|B|} \sum_{i \in B} \frac{\|\sigma_i s_\omega(\widetilde{\phi}_i; \beta, \sigma_i) + \eta_i\|^2}{d}.$$

d: number of field components. The factor σ_i balances the noise scales.

1. Draw a minibatch of stored clean fields and noise scales.
2. Add fresh noise and evaluate the network.
3. Backpropagate $\nabla_\omega \widehat{\mathcal{L}}_B$ and update ω with the optimizer.
4. Repeat with another minibatch.

The clean fields remain training examples. Only the network parameters change during the optimizer step.

The noise target is known, while averaging over physical fields teaches the noisy physical score.

Gaussian-noise formulation: Song and Ermon, arXiv:1907.05600. For compact fields, the noise density and its score are periodic.

The trained parameters ω_* stay fixed

Trial field at proposal scale ℓ

$$\phi' = \phi + h_\ell s_{\omega_*}(\phi; \beta, \sigma_\ell) + \sqrt{2h_\ell}\,\eta.$$

$$q_\ell(\phi' \mid \phi) = \mathcal{N}(\phi'; \phi + h_\ell s_{\omega_*}, 2h_\ell I)\,.$$

Exact accept/reject probability

$$\alpha_\ell = \min\left[1,\, e^{-S_\beta(\phi')+S_\beta(\phi)}\,\frac{q_\ell(\phi \mid \phi')}{q_\ell(\phi' \mid \phi)}\right].$$

$$\phi_{\text{next}} = \begin{cases} \phi', & u < \alpha_\ell, \\ \phi, & u \geq \alpha_\ell, \end{cases} \qquad u \sim \text{Uniform}(0,1).$$

1. Evaluate the fixed score at the current field.
2. Draw fresh noise and propose ϕ'.
3. Evaluate the score at ϕ' for the reverse probability.
4. Evaluate the target action difference and accept or retain the old field.

$h_\ell > 0$ sets the step size. This update changes the configuration without a training loss or an optimizer step.

MAALA: Metropolis-adjusted annealed Langevin algorithm. Zhu, Aarts, W. Wang, Zhou and L. Wang, arXiv:2502.05504. Compact fields require the corresponding periodic proposal density.

Generation repeats the corrected updates across noise scales

A complete sweep changes proposal scales at one fixed physical coupling

$$\sigma_1 > \sigma_2 > \cdots > \sigma_m, \qquad p_\beta K_\ell = p_\beta \quad \text{for each MH-corrected kernel } K_\ell.$$

$$\underbrace{\phi_n \xrightarrow{K_1} \phi_{n,1} \xrightarrow{K_2} \cdots \xrightarrow{K_m} \phi_{n+1}}_{\text{one complete generation sweep}}.$$

Within a sweep

- Each level evaluates the same fixed network at its noise scale.
- Every MH test uses the same physical action S_β.
- Every accepted or retained field enters the next level.

Between sweeps

- The final field becomes the next sweep's initial field.
- Discard warmup and record subsequent fields.
- Keep repeated fields after rejection with equal measurement weights.

This produces a persistent Markov chain whose fixed kernel preserves p_β.

See also G. Kanwar's talk on Mon.

A. Tomiya, arXiv:2607.12587.

The action relates a source ensemble to the next target

Training fields at β_s can inform the score at a nearby β_t

$$p_\beta(\phi) = Z_\beta^{-1} e^{-S_\beta(\phi)}.$$

$$w(\phi) = e^{-S_{\beta_t}(\phi) + S_{\beta_s}(\phi)}.$$

$$\langle f \rangle_{\beta_t} = \frac{\langle w f \rangle_{\beta_s}}{\langle w \rangle_{\beta_s}}.$$

β_s: source coupling.
β_t: target coupling.
f: the training loss at β_t in this work.

We reweight the training loss, not physical observables

Give each source configuration the weight appropriate to the target action. Then train a score conditioned on β_t.

Physical observables use equal-weight averages over the MH-corrected target ensemble.

Nearby distributions can share useful configurations even when their probabilities differ.

The identity is exact for ensemble averages. Finite-sample ratios introduce statistical error and bias.

The known action changes the training weights before target sampling begins.

A. Tomiya, SLDiffusion-MAALA (2026).

The source fields stay at β_s, while the network learns at β_t

$$w_i = e^{-S_{\beta_t}(\phi_i)+S_{\beta_s}(\phi_i)}, \qquad \bar{w}_i = \frac{w_i}{\sum_j w_j}, \qquad \phi_i \sim p_{\beta_s}.$$

$$\widehat{\mathcal{L}}_{s\to t}(\omega) = \sum_i \bar{w}_i \, \mathbb{E}_{\sigma,\eta}\left[\underbrace{\frac{\|\sigma s_\omega(\phi_i + \sigma\eta; \beta_t, \sigma) + \eta\|^2}{L^2}}_{\text{denoising loss conditioned on the target coupling}} \right].$$

One weighted training step

1. Draw source indices with probabilities $\bar{w}_i$.
2. Add fresh noise to those clean fields.
3. Backpropagate the minibatch loss at β_t and update ω.

Weights belong to the clean source fields. Sampling indices in proportion to the weights implements the objective without multiplying by them again.

The trained model is then frozen before generation at β_t begins.

A. Tomiya, SLDiffusion-MAALA (2026).

The self-learning cycle alternates training and generation

A corrected target ensemble becomes the source for the next coupling

Stage	Calculation	Output
Starting ensemble	Draw fields from an accessible reference distribution p_{β_s}	Source fields $\mathcal{R}_s$
Training at β_t	Action-weight the source fields. Repeat noise addition, score loss, and optimizer steps	Target score $s_{\omega_*}(\cdot;\beta_t,\sigma)$
Model selection	Evaluate a held-out validation set and freeze the selected parameters	Fixed ω_*
Generation at β_t	Repeat the corrected sweeps, discard warmup, and retain the chain	Corrected fields $\mathcal{R}_t$
Next coupling	Use $\mathcal{R}_t$ as the source for the next action-weighted fit	A new training stage

Learning changes the proposal between generation stages. Within each generation stage, the fixed MH kernel preserves the chosen physical target.

A. Tomiya, SLDiffusion-MAALA (2026).

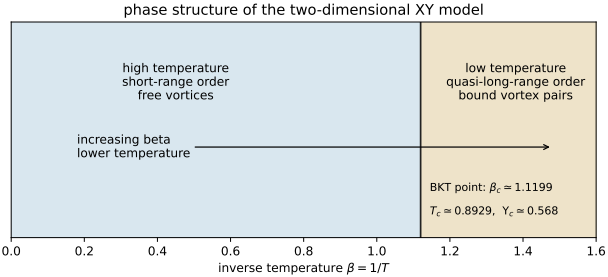
III. The XY model: implementation and tests

Proof of concept: self-generated sampling in a compact-field model

$$\theta_x \in (-\pi, \pi], \qquad S_\beta[\theta] = -\beta \sum_{x,\mu=1,2} \cos(\theta_{x+\hat{\mu}} - \theta_x), \qquad \beta = 1/T.$$

- compact field and global $U(1)$ symmetry;
- vortices from integer plaquette winding;
- BKT point $\beta_c \simeq 1.1199$;
- $0 \leq \beta \leq 0.50$, $L = 4, 6, 8, 12$; periodic square lattices.

At $\beta = 0$, every angle is independent. Increasing β creates the correlations that the score must learn.

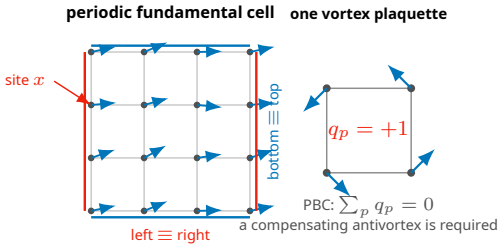


Increasing $\beta = 1/T$ binds vortices and produces quasi-long-range order. The tested interval ends well to the left of the BKT point.

M. Hasenbusch, arXiv:cond-mat/0502556.

Energy, vortices, and helicity probe distinct physics

One sampler must reproduce local, defect, and response observables



Grid intersections are lattice sites, and each spin arrow starts there. Periodic boundaries identify matching edge spins. The inset shows the oriented winding of one plaquette.

Energy per site: local alignment

$$e = \frac{H}{V} = -\frac{1}{V}\sum_{x,\mu} \cos \Delta_\mu \theta_x .$$

Vortex density: topological defects

$$\rho_v = \frac{1}{V}\sum_p |q_p|, \qquad q_p = \frac{1}{2\pi}\sum_{\ell \in \partial p} \mathrm{wrap}(\Delta\theta_\ell)\,.$$

Helicity modulus: response to a twist

$$\overline{\Upsilon} = \tfrac{1}{2}(\Upsilon_1 + \Upsilon_2).$$

Spin stiffness: gently twist the spin angles across the lattice. A larger $\overline{\Upsilon}$ means a greater free-energy cost.

A. Tomiya, arXiv:2607.12587, Sec. 2. Spin stiffness: Khairnar–Vojta, arXiv:2312.04468.

One ladder step: learn, generate, pass on the ensemble

$\beta_0 \to \beta_1$ is one step. Repeating this cycle builds the ladder.

Stage	Input and operation	Output
1. Learn	Configurations at β_0 let us evaluate the training loss for β_1. Minimize this loss to learn the β_1 score at each noise width σ.	Scores $s_\omega(\cdot;\beta_1,\sigma)$
2. Generate	Freeze the model. Run MAALA with MH targeting the physical distribution p_{β_1}. Discard warmup.	Corrected fields at β_1
3. Repeat	Use the generated β_1 configurations to evaluate the loss for β_2, learn its scores, and generate there.	Next step: $\beta_1 \to \beta_2$

XY example: start with independent uniform angles at $\beta = 0$.

$0 \to 0.30$: evaluate the loss for 0.30, learn its scores, then generate with MAALA.
$0.30 \to 0.35$: use the generated fields to learn the scores for 0.35, then generate again.

The reweighting identity holds for population averages. Finite data and a finite network give an approximate score. MH preserves the exact physical target.

A. Tomiya, SLDiffusion-MAALA (2026).

The XY starting ensemble: uniform angles at $\beta = 0$

$L = 4, \beta_s = 0, \beta_t = 0.30$

$$S_\beta = \beta H, \qquad \theta_x \overset{\text{iid}}{\sim} \text{Uniform}(-\pi, \pi].$$

$$w_i = e^{-0.30 H(\theta_i)}, \qquad \bar{w}_i = \frac{w_i}{\sum_j w_j}.$$

20,000 independent uniform configurations supply the first training set. No $\beta = 0$ neural network is required.

Low-energy configurations receive larger weights for the interacting target.

How concentrated are the weights?

$$N_{\text{weight}} = \frac{1}{\sum_i \bar{w}_i^2} = 3,841.$$

19.2% of the source count.
Largest normalized weight: 0.00620.
Top 1% of configurations: 14.7% of total weight.

Noise is added after assigning weights to the clean configurations.

Uniform samples and the action provide a training objective for the $\beta = 0.30$ score.

A. Tomiya, SLDiffusion-MAALA (2026).

32 periodic MH levels per recorded trajectory

$$\theta' = \mathrm{wrap}\left[\theta + aD_\ell s_\omega(\theta;\beta_t,g_\ell) + \sqrt{2aD_\ell}\eta\right], \qquad \omega \text{ fixed.}$$

$$\alpha_\ell = \min\left[1,\ e^{-S_{\beta_t}(\theta')+S_{\beta_t}(\theta)}\frac{q_\ell(\theta \mid \theta')}{q_\ell(\theta' \mid \theta)}\right].$$

32 MH updates per recorded trajectory

$g_\ell : 0.80 \to 0.05$, $D_\ell = g_\ell^2/2$, $a = 2$.

At each level, accept θ' or keep θ. Every level targets p_{β_t}.

Retain the chain, including rejected stays

The last field of a sweep starts the next sweep.

Discard warmup and save the chain with equal measurement weights.

Training weights improve the proposal. Physical observables come from the corrected chain.

A. Tomiya, SLDiffusion-MAALA (2026).

Source configurations and target-score models have different couplings

$$\beta = 0 \longrightarrow 0.30 \longrightarrow 0.35 \longrightarrow 0.40 \longrightarrow 0.45 \longrightarrow 0.50$$

Example: $0.30 \to 0.35$

1. Start with corrected $\beta = 0.30$ fields.
2. Weight them by $e^{-0.05H}$.
3. Fit the score conditioned on $\beta = 0.35$.
4. Freeze that score and sample at $\beta = 0.35$.

The generated fields supply the next stage

After thermalization and distribution checks, 10,000 corrected fields become the next training source.

$$\mathcal{R}_k \xrightarrow[\text{action weights}]{\text{fit at } \beta_{k+1}} s_{\omega_{k+1}} \xrightarrow{\text{MH}} \mathcal{R}_{k+1}.$$

No separately generated target-HMC ensemble enters this loop.

A. Tomiya, SLDiffusion-MAALA (2026).

Volume transfer and retraining at $\beta = 0.50$

The source configurations live on the new lattice size

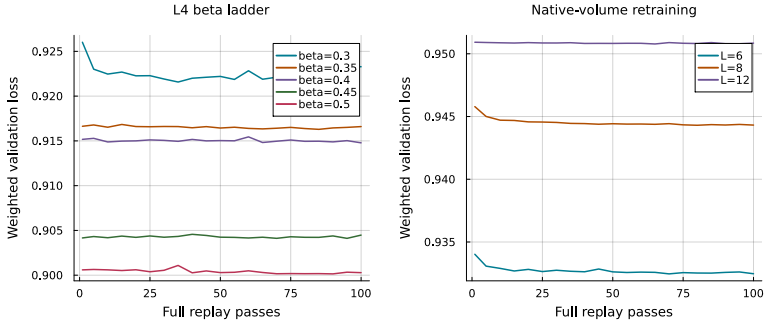
Stage	Calculation at the target lattice size L
Weight transfer	$L = 6, 8$ inherit the $L = 4$ score. $L = 12$ inherits the retrained $L = 8$ score.
Source generation	Initialize angles on the new lattice. Run the transferred fixed score with the new-size action, discard warmup, and collect 50,000 corrected fields.
Training	Train on these same-β, same-L fields with equal weights. Select the model using held-out validation loss.
Independent test	Freeze the retrained score. Run a separate chain with 20,000 warmup and 50,000 measurement trajectories.

Convolutional parameters transfer between lattice sizes. The sampler generates the training configurations directly on the target lattice.

A. Tomiya, SLDiffusion-MAALA (2026).

Training uses a fixed validation ensemble

100 passes, batch size 64, Adam learning rate 2×10^{-4}



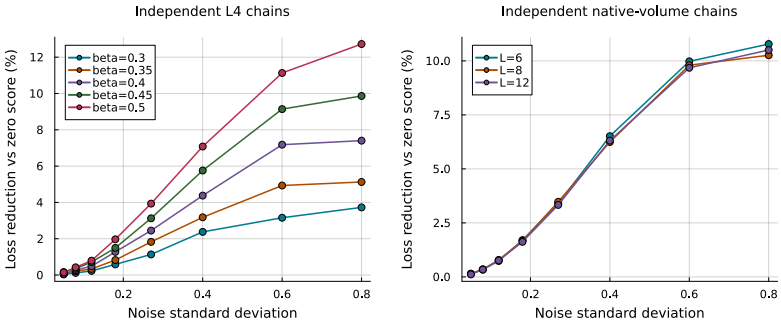
A pass consists of weighted draws with replacement, equal in number to the training set. 1,024 held-out source fields define the fixed validation loss. The selected model minimizes this loss.

The right panel shows retraining on corrected fields generated at each lattice size.

A. Tomiya, SLDiffusion-MAALA (2026).

Independent fields test the learned scores

The test ensemble does not select the model



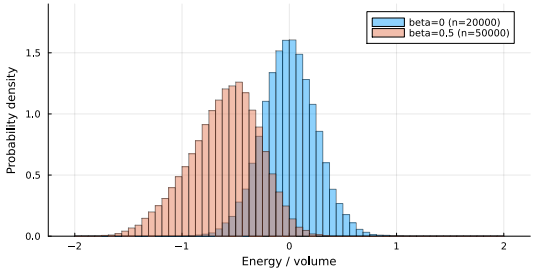
Vertical axis: $100(1 - \mathcal{L}_{\mathrm{learned}}/\mathcal{L}_{\mathrm{zero}})$ at each noise scale. 4,096 draws per scale from independent measurement chains. The right panel uses models retrained at each L.

At $L = 12$, the learned score lowers the loss at all 8 noise scales, by 10.50% at $\sigma = 0.80$.

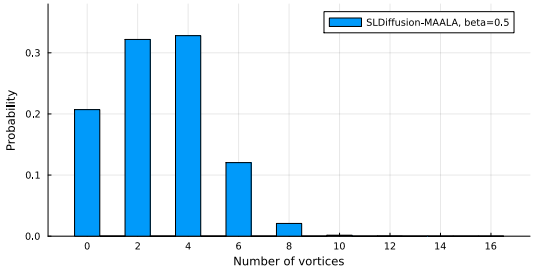
A. Tomiya, SLDiffusion-MAALA (2026).

Interactions reshape both alignment and defects

$L = 4$: independent uniform fields and corrected $\beta = 0.50$ fields



Energy density $e = H/V$. The interacting ensemble favors locally aligned spins.



The $\beta = 0.50$ vortex distribution has discrete support. Periodic boundaries impose zero net winding charge.

The self-generated ensemble develops nontrivial structure from a uniform starting distribution.

A. Tomiya, SLDiffusion-MAALA (2026).

Independent chains compare physics and decorrelation

$\beta = 0.50$, $L = 4, 6, 8, 12$

	SLDiffusion–MAALA	HMC
Fixed sampler	Trained score; 32 MH levels	MH-corrected Hamiltonian trajectories
Measured acceptance ($L = 4, 6, 8, 12$)	70.77%, 60.45%, 50.49%, 37.21% per MH level	88.28%, 82.47%, 76.54%, 65.08% per trajectory
One recorded trajectory	32 full-lattice proposals	Trajectory length 1
Warmup trajectories	20,000	20,000
Measurement fields	50,000, no thinning	500,000, no thinning
Error estimates	Contiguous-block jackknife, block length 200	

$$\tau_{\mathrm{int}}(\mathcal{O}) = \frac{1}{2} + \sum_{t=1}^{W} \rho_{\mathcal{O}}(t), \qquad \tau_{\mathrm{int}} = \frac{1}{2} \text{ for independent records.}$$

Autocorrelation times below are in each sampler's recorded-trajectory units. Their ratio is not a wall-clock speedup.

A. Tomiya, SLDiffusion-MAALA (2026).

Configurations can supply their own temperature check

A Schwinger–Dyson identity for $S_\beta = \beta H$

$$0 = \left\langle \sum_x \left[\partial_{\theta_x}^2 H - \beta (\partial_{\theta_x} H)^2 \right] \right\rangle .$$

$$\boxed{\beta_{\mathrm{SD}} = \frac{\left\langle \sum_x \partial_{\theta_x}^2 H \right\rangle}{\left\langle \sum_x (\partial_{\theta_x} H)^2 \right\rangle}}$$

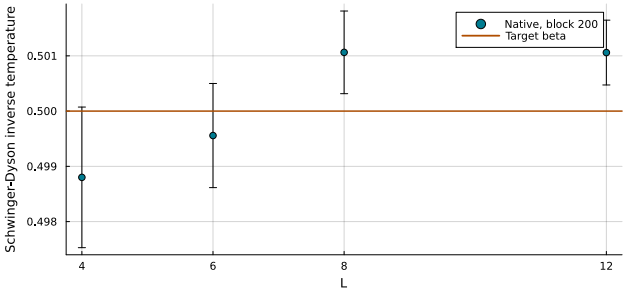
Integration by parts on periodic angles relates measurable derivatives of the Hamiltonian. This check uses the sampled fields and the action, without an HMC reference.

Several ways to estimate temperature from configurations

Here we use a Schwinger–Dyson identity; another choice is Creutz's demon method. See also the configurational-thermometer approach to probing diffusion models in **D. E. Habibi's talk on Mon.**

A. Gonzalez-Arroyo and M. Okawa, *Renormalized coupling constants by Monte Carlo methods*, Phys. Rev. D 35, 672 (1987).
M. Creutz, *Microcanonical Monte Carlo Simulation*, Phys. Rev. Lett. 50, 1411 (1983).
A. Kagimura, A. Tomiya and R. Yamamura, *Effective lattice action for the configurations smeared by the Wilson flow*, arXiv:1508.04986.

Independent measurement chains at $\beta = 0.50$

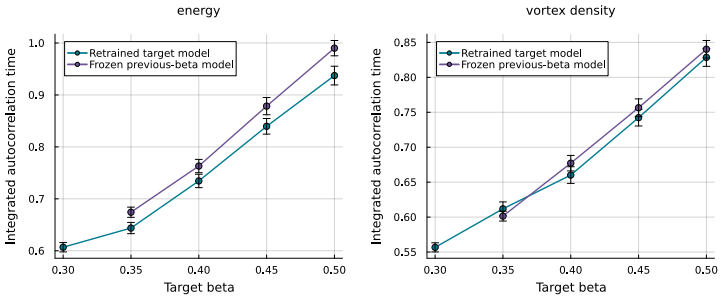


“Native”: score trained at the measured lattice size. Errors: block jackknife of the ratio, block length 200.

MH correction preserves the exact target distribution in exact arithmetic. The finite-chain temperature estimate can fluctuate away from the input value: at $L = 12$, $\beta_{\mathrm{SD}} = 0.50106(59)$ differs from 0.50000 by 1.8 standard errors.

A. Tomiya, SLDiffusion-MAALA (2026).

$L = 4$: the target score is fitted before the target chain is generated



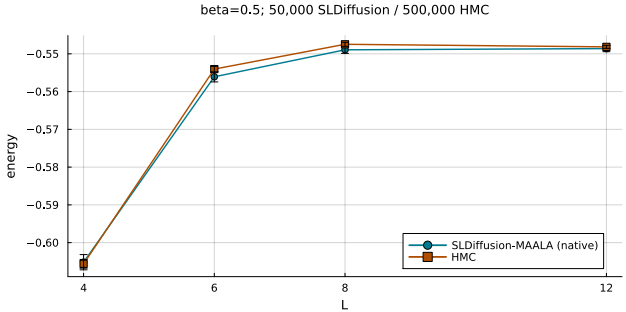
Target model: fit preceding-β fields with action weights and target-β conditioning. Frozen previous model: retain both the preceding weights and conditioning, changing only the MH target action.

For $0.30 \to 0.35$, the learned 0.35 score uses weighted 0.30 fields. The separate control uses the fixed 0.30 score at MH target 0.35.

A. Tomiya, SLDiffusion-MAALA (2026).

The energy density agrees with independent HMC

Local alignment across four lattice sizes at $\beta = 0.50$



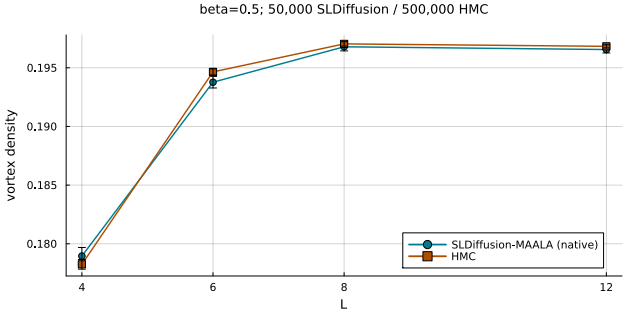
$e = H/V$. SLDiffusion: 50,000 fields; HMC: 500,000 fields. “Native” means the score was trained on corrected fields at that lattice size.

The largest energy difference is 1.35 combined standard errors, at $L = 8$.

A. Tomiya, SLDiffusion-MAALA (2026).

The vortex density also agrees with independent HMC

The same sampler reproduces the density of topological defects



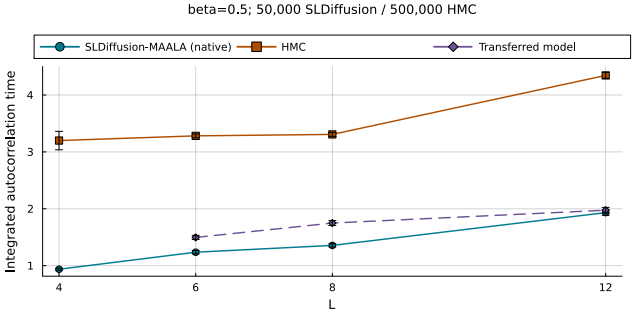
$\rho_v = V^{-1}\sum_p |q_p|$. "Native" denotes training at the measured lattice size. All quoted errors include time correlations through block jackknife.

All energy and vortex means lie within 1.58 combined standard errors of HMC.

A. Tomiya, SLDiffusion-MAALA (2026).

Energy correlations stay short as the lattice grows

$\tau_{\mathrm{int}}(e) < 2$ through $L = 12$



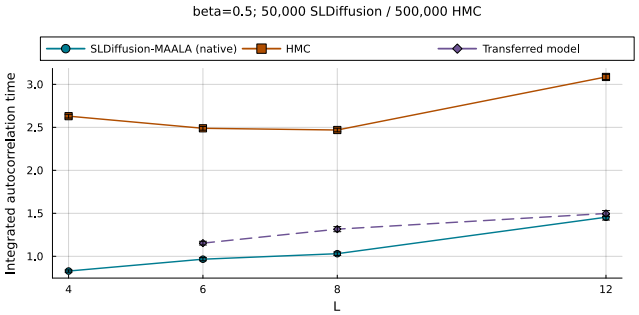
"Transfer": weights from a smaller lattice. "Native": retrained on corrected fields at the measured size. Error bars are block-jackknife errors with a reselected correlation window.

$\tau_{\mathrm{SLD}}/\tau_{\mathrm{HMC}} = 0.293(16)$ at $L = 4$ and $0.445(12)$ at $L = 12$, in recorded-trajectory units.

A. Tomiya, SLDiffusion-MAALA (2026).

Vortex correlations remain short too

Defect-density autocorrelations across $L = 4, 6, 8, 12$



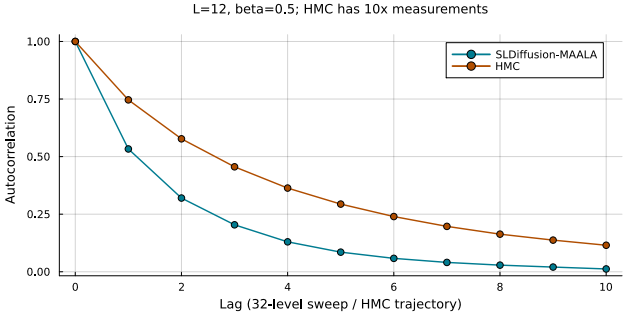
"Native" means retraining at the measured size; "transfer" uses a smaller-lattice score. One SLDiffusion record contains 32 MH levels; one HMC record has trajectory length 1.

$L = 12$: $\tau_{\mathrm{int}}(\rho_v) = 1.455(33)$ for SLDiffusion and $3.086(31)$ for HMC.

A. Tomiya, SLDiffusion-MAALA (2026).

The $L = 12$ energy autocorrelation decays within a few records

The full correlation function shows the origin of the shorter integrated time



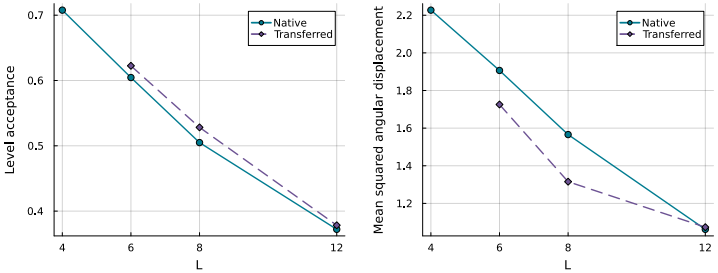
$\rho_e(t) = \langle (e_n - \bar{e})(e_{n+t} - \bar{e}) \rangle / \langle (e_n - \bar{e})^2 \rangle$. The horizontal axis counts recorded trajectories. Curves are point estimates.

The integrated times are $1.932(46)$ for the score retrained at $L = 12$ and $4.344(52)$ for HMC.

A. Tomiya, SLDiffusion-MAALA (2026).

Volume transfer changes acceptance and movement

Transfer the convolutional weights, then train on the new lattice's corrected fields



Acceptance is per MH level. Mean squared jump distance is $\langle V^{-1} \sum_x \mathrm{wrap}(\theta_x^{n+1} - \theta_x^n)^2 \rangle$ between recorded fields. "Native" means retrained at that lattice size.

At $L = 6, 8$, retraining increases movement even though acceptance decreases. Autocorrelation measures the net gain.

A. Tomiya, SLDiffusion-MAALA (2026).

Retrain at fixed $\beta = 0.50$ using 50,000 corrected configurations

L	Transferred score	$\tau_e^{\text{retrained}}/\tau_e^{\text{transfer}}$	$\tau_{\rho_v}^{\text{retrained}}/\tau_{\rho_v}^{\text{transfer}}$
6	From $L = 4$	$0.826(28)$	$0.838(22)$
8	From $L = 4$	$0.775(25)$	$0.783(23)$
12	From the retrained $L = 8$ model	$0.978(34)$	$0.971(31)$

$L = 6, 8$: a resolved improvement

Autocorrelation times decrease by about 16–22%. The sampler supplies its own adaptation data.

$L = 12$: similar autocorrelations

The inherited $L = 8$ score already gives short correlations. The further gain is unresolved within the errors.

A. Tomiya, SLDiffusion-MAALA (2026).

An exact sampler can learn from its own corrected history

A. Tomiya

arXiv:2607.12587

Action-weighted learning, Metropolis correction, and volume transfer

1. **Use the action to learn ahead.**
   Weighted source fields train the next-coupling score. Starting from independent $\beta = 0$ angles, the self-learning cycle reaches $\beta = 0.50$.

2. **Use Metropolis correction to generate physical fields.**
   A fixed learned score guides the proposal; the exact action determines the invariant measure. Energy and vortex means agree with independent HMC through $L = 12$.

3. **Use the generated fields to adapt the score.**
   Retraining reduces autocorrelation times by 16–22% at $L = 6, 8$. Measured energy and vortex autocorrelation times remain below 2 through $L = 12$.

The learned proposal supplies efficient moves; its corrected samples supply the next training set.

A. Tomiya, SLDiffusion-MAALA (2026).



Backup: learning, sampling, and statistical tests

Phase structure and learned sampling (2016–2023)

2016 **1609.09087** Detection of phase transition via convolutional neural network

Phase transition from latent information
A. Tanaka and A. Tomiya; first submitted 28 Sep 2016.
Almost 10 years ago.

2017 **1712.03893** Towards reduction of autocorrelation in HMC by machine learning

2018 **1802.08313** Deep learning and the AdS/CFT correspondence

2018 **1809.10536** Deep Learning and Holographic QCD

2018 **1812.01522** Phase transition encoded in neural network

2020 **2010.11900** Self-learning Monte Carlo for non-Abelian gauge theory with dynamical fermions

2021 **2103.11965** Gauge covariant neural network for quarks and gluons

2021 **2112.01586** HMC with Normalizing Flows

2022 **2208.08903** GomalizingFlow.jl: A Julia package for Flow-based sampling algorithm for lattice field theory

2023 **2306.11527** Self-learning Monte Carlo with equivariant Transformer

2023 **2310.13222** Equivariant transformer is all you need

2023 **2311.15233** Sparse modeling approach to extract spectral functions with covariance of Euclidean-time correlators of lattice QCD

Related review: *Machine Learning for Lattice QCD*, JPSJ 94, 031006 (open access).

Lattice observables and generative methods (2024–2026)

2024 **2410.23567** Sparse modeling study to extract spectral functions from lattice QCD data

2024 **2411.18170** Machine Learning Estimation on the trace of inverse Dirac operator using the Gradient Boosting Decision Tree Regression

2025 **2501.16955** CASK: A Gauge Covariant Transformer for Lattice Gauge Theory

2025 **2509.10386** Sparse modeling study of extracting charmonium spectral functions from lattice QCD at finite temperature

2026 **2602.20516** Lattice Gauge Theory via LLVM-Level Automatic Differentiation

See also Y. Nagai's talk on Wed.

2026 **2602.21617** Machine Learning-Based Estimation of Cumulants of Chiral Condensate via Multi-Ensemble Reweighting with Deborah.jl

2026 **2602.23731** A Machine Learning Approach for Lattice Gauge Fixing

2026 **2603.16329** Parameter Optimization of Domain-Wall Fermion using Machine Learning

2026 **2607.12587** Lattice Configuration Generation with a Self-Learning Diffusion Model

2026 **2608.30416** Bias-Corrected Machine-Learning Estimation of Chiral Condensate Cumulants: A Retrospective Lattice QCD Case Study

Related review: *Machine Learning for Lattice QCD*, JPSJ 94, 031006 (open access).

Exact correction enables self-generated training data

The true action judges every trial field before it enters the replay

$$\underbrace{\phi_n}_{\text{current field}} \xrightarrow{\text{learned drift + noise}} \underbrace{\phi_{\text{trial}}}_{\text{approximate ML move}} \xrightarrow{\text{exact accept/reject using } \Delta S \text{ and proposal asymmetry}} \underbrace{\phi_{n+1}}_{\text{accepted trial or old field}} .$$

repeat with the network frozen $\implies$ $\mathcal{D}_{\text{self}} = \{\phi_n\}_{\text{after warmup}}$ samples $\pi(\phi) \propto e^{-S[\phi]}$.

ML determines efficiency

The learned force chooses the direction and size of the trial move. A poor approximation lowers acceptance or slows mixing.

The action determines the ensemble

The exact action enters every accept/reject decision. After warmup, the retained fields may train the next proposal.

The sampler can improve from its own corrected history without a separately generated target ensemble.

A. Tomiya, arXiv:2607.12587.

Data-free diffusion: learning signals and replay

Both methods reuse generated fields without an external target ensemble

	Nakajima et al. (ϕ^4)	SLDiffusion (XY)
Training fields	Current diffusion model fills a replay buffer	Exactly sampled $\beta = 0$ fields, then MH-corrected chain replay
Score target	Blend of the noise-kernel score and the action force $-\nabla S$ (control variates)	Noise-kernel score learned from fields weighted for the target coupling
Training progression	Repeated generation and score updates at the chosen target parameters, without importance weights	Coupling ladder: weight source fields by $e^{-S_{\beta_t}+S_{\beta_s}}$, train at β_t, then generate with MH
Measurements	Reweight generated samples for physical observables	MH keeps the target invariant. Equal-weight chain measurements

For Nakajima et al., omitting importance weights in training does not exclude reweighting observables.

See also S. Nakajima's talk on Tue.

S. Nakajima et al., *Data Free Training of Diffusion Models for Lattice Simulations*, ECT*, 8 Sep 2026, lecture slides, pp. 26, 31, 40, 48–50, 57 (ϕ^4 study).

A. Tomiya, arXiv:2607.12587.

The network predicts a score field at each noise scale

A periodic convolutional network with 690,194 trainable parameters

$$(\widetilde{\theta}, \beta, \sigma) \longmapsto u_\omega = \sigma s_\omega.$$

- 11 input channels encode local angle differences and physical features.
- 48 feature channels and 8 residual blocks.
- Noise-dependent feature scales and shifts: FiLM.

$$u_\omega(x) = \sum_{\mu=1}^{2} \left[J_{\omega,\mu}(x) - J_{\omega,\mu}(x - \hat{\mu}) \right].$$

The network predicts 2 link fields $J_{\omega,\mu}$. Their lattice divergence gives a site score with zero spatial sum.

Shared convolutional weights allow the same architecture to act on different lattice sizes.

FiLM: feature-wise linear modulation. The input includes local-force features, while the output is learned without adding an analytic force.

A. Tomiya, SLDiffusion-MAALA (2026).

Adam learning rate 2×10^{-4}, batch size 64

1. Draw a minibatch of up to 64 clean source fields with probability proportional to their action weights.
2. Draw a noise scale and fresh Gaussian noise for each field.
3. Wrap the noisy angles and evaluate $u_\omega = \sigma s_\omega$ at the target β_t.
4. Backpropagate the periodic denoising loss and apply one Adam update to ω.

Known periodic training target

$$\widetilde{\theta}_i = \mathrm{wrap}(\theta_i + \sigma_i \eta_i),$$

$$y_i = \sigma_i P \nabla_{\widetilde{\theta}_i} \log K_{\sigma_i}(\widetilde{\theta}_i \mid \theta_i).$$

$$\widehat{\mathcal{L}}_B = \frac{1}{|B|V} \sum_{i \in B} \|u_\omega - y_i\|^2.$$

K_σ: wrapped Gaussian density.
P: removal of the spatially constant component.
The optimizer updates the 690,194 network parameters.

100 passes, each containing as many weighted draws as training fields. A fixed set of 1,024 held-out source fields selects the minimum-validation-loss model.

A. Tomiya, SLDiffusion-MAALA (2026).

The training weight belongs to the clean field

$$\widetilde{\theta} = \mathrm{wrap}(\theta_i + \sigma\eta).$$

$$\widehat{\mathcal{L}}(\omega) = \sum_{i\in\mathcal{T}} \bar{w}_i^{(\mathcal{T})} \mathbb{E}_{\sigma,\eta}\left[\frac{1}{V}\left\| u_\omega - \sigma P \nabla_{\tilde{\theta}} \log K_\sigma(\widetilde{\theta} \mid \theta_i)\right\|^2\right].$$

$u_\omega = \sigma s_\omega(\widetilde{\theta}; \beta_t, \sigma)$: the network's normalized score output.
K_σ: known wrapped Gaussian noise density.
P: removal of the spatially constant component.

1. Draw clean fields with probability proportional to w_i.
2. Add noise and condition the network on β_t.
3. Update the network parameters ω.

Weighted sampling with replacement implements this objective. There is no second weight factor in the training minibatch.

A. Tomiya, SLDiffusion-MAALA (2026).

HMC enters only after the diffusion model is fixed

Stage	Fields	Role
Score training	Weighted source configurations	Update ω
Fixed validation	1,024 held-out source configurations	Select the score model
Corrected replay	Thermalized chain at the target coupling	Supply the next training stage
Measurement chain	50,000 fields after 20,000 warmup trajectories	Measure physics and autocorrelations
Independent HMC	500,000 fields after 20,000 warmup trajectories	External validation only

The final score test and HMC comparison do not change model parameters, training duration, or checkpoint selection.

HMC integration: leapfrog with step size $\epsilon = 0.5$, 2 steps per trajectory, trajectory length 1.

A. Tomiya, SLDiffusion-MAALA (2026).

Nearby couplings preserve useful importance weights

The clean-field action ratio controls the overlap available for learning

$$\log w = -(\beta_t - \beta_s)H, \qquad \mathrm{Var}(\log w) = (\Delta\beta)^2 \,\mathrm{Var}_{\beta_s}(H).$$

Coupling step	$0 \to 0.30$	$0.30 \to 0.35$	$0.35 \to 0.40$	$0.40 \to 0.45$	$0.45 \to 0.50$
Source count	20,000	10,000	10,000	10,000	10,000
N_{weight}/N	0.192	0.950	0.944	0.939	0.935

Each corrected ensemble refreshes the source distribution before the next action-weighted fit. The ladder avoids keeping the original uniform source for every target.

Kish weight count measures weight concentration, not time correlations or whether rare configurations have been adequately represented.

See also E. Cellini's talk on Tue.

A. Tomiya, SLDiffusion-MAALA (2026).

The score network uses periodic local features and noise conditioning

690,194 parameters; the architecture does not depend on the lattice size

Component	Specification
11 inputs	$\sin \Delta_\mu \theta, \cos \Delta_\mu \theta$ for 2 directions (4 channels); local energy, force, force squared, vortex, absolute twist; β, σ
Entrance	5×5 convolution, 11 to 48 channels; periodic padding
Noise embedding	$\log \sigma$; 24 fixed Gaussian Fourier frequencies (48 components); dense $48 \to 96 \to 48$, swish
8 residual blocks	2 convolutions each, alternating 3×3 and 5×5 blocks; noise-conditioned feature scales and shifts (FiLM)
Normalization	GroupNorm, 8 groups, $\epsilon = 10^{-5}$
Output	GroupNorm, swish, 3×3 convolution to 2 link fields; lattice divergence gives $u_\omega = \sigma s_\omega$
Precision	Network Float32; configurations and MH arithmetic Float64

Translation-equivariant and invariant under a common angle shift. The divergence output has zero spatial sum, matching the global-shift symmetry.

A. Tomiya, SLDiffusion-MAALA (2026).

Adam 2×10^{-4}, batch size 64, 100 weighted-sampling passes

Target model	Source configurations	Optimizer updates	Selected pass
$L = 4,\ \beta = 0.30$	20,000	29,700	35
$L = 4,\ \beta = 0.35$	10,000	14,100	93
$L = 4,\ \beta = 0.40$	10,000	14,100	67
$L = 4,\ \beta = 0.45$	10,000	14,100	3
$L = 4,\ \beta = 0.50$	10,000	14,100	66
$L = 6,\ \beta = 0.50$	50,000	76,600	70
$L = 8,\ \beta = 0.50$	50,000	76,600	73
$L = 12,\ \beta = 0.50$	50,000	76,600	65

1,024 source fields are held out with fixed validation noise. Training and validation weights are normalized separately, without clipping. Each new stage warm-starts the network and resets Adam's moments.

A. Tomiya, SLDiffusion-MAALA (2026).

Training noise and sampling noise serve different purposes

The noisy training target does not replace the physical MH target

Training

8 noise scales, sampled with equal probability:
$\sigma \in \{0.05, 0.08, 0.12, 0.18,$
$0.27, 0.40, 0.60, 0.80\}$.

The target is the wrapped-noise score, multiplied by σ and projected to zero spatial mean.

Sampling

32 geometric scales, $g_\ell : 0.80 \to 0.05$.
$D_\ell = g_\ell^2/2$, $a = 2$.
Drift: $aD_\ell s_\omega$.
Noise variance: $2aD_\ell$.

1 full-lattice proposal and 1 MH test at every level.

The same β appears in all 32 MH target actions. The noise ladder changes proposal scales, not the physical temperature.

A. Tomiya, SLDiffusion-MAALA (2026).

Detailed balance uses both directions of the learned transition

$$\alpha(\phi, \phi') = \min\left[1, \frac{e^{-S(\phi')} q_\omega(\phi \mid \phi')}{e^{-S(\phi)} q_\omega(\phi' \mid \phi)}\right].$$

$$\pi(\phi) K(\phi' \mid \phi) = \pi(\phi') K(\phi \mid \phi').$$

Exactness
follows from the transition kernel.

Efficiency
follows from how well q_ω matches the target geometry.

Learning may improve mixing; correction protects the measure.

A. Tomiya, arXiv:2607.12587.

Each noise level is a Markov kernel for the same physical target

The ladder changes proposal scale, not the invariant distribution

$$q_i(\theta' \mid \theta) = \mathcal{N}_{2\pi}(\theta';\ \mathrm{wrap}(\theta + aD_i s_\omega(\theta, \sigma_i))\ ,\ 2aD_i\, I)\ ,$$

$$A_i(\theta \to \theta') = \min\left[1, e^{-S_\beta(\theta')+S_\beta(\theta)} \frac{q_i(\theta \mid \theta')}{q_i(\theta' \mid \theta)}\right].$$

$\pi_\beta(\theta) q_i(\theta' \mid \theta) A_i(\theta \to \theta') = \pi_\beta(\theta') q_i(\theta \mid \theta') A_i(\theta' \to \theta)$.

Composition of the 32 invariant kernels leaves π_β invariant; the trained drift controls mixing.

A. Tomiya, arXiv:2607.12587.

Compact angles require a wrapped proposal density

Both directions of the MH ratio include all periodic images

$$\mathcal{N}_{2\pi}(\theta';\mu,vI)=\prod_x\sum_{n\in\mathbb{Z}}\frac{\exp[-(\theta'_x-\mu_x+2\pi n)^2/(2v)]}{\sqrt{2\pi v}}.$$

Forward proposal

Evaluate the score at θ and center the wrapped Gaussian at $\mathrm{wrap}(\theta + aD_\ell s_\omega(\theta))$.

Reverse probability

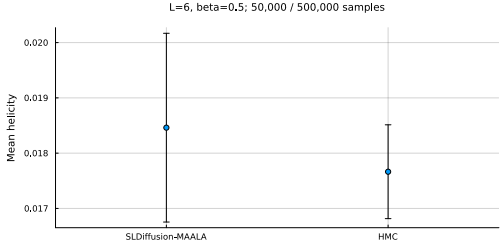
Evaluate the score again at the candidate θ'. Use its drift to compute $q_\ell(\theta \mid \theta')$.

The implemented sum uses images $n = -3,\dots,3$. Extending to $n = -7,\dots,7$ agrees numerically over the tested proposal variances.

A. Tomiya, SLDiffusion-MAALA (2026).

$L = 6, \beta = 0.50$: an observable beyond energy and vortex count

$$\Upsilon_\mu = \frac{1}{V}\sum_x \cos \Delta_\mu \theta_x - \frac{\beta}{V}\left(\sum_x \sin \Delta_\mu \theta_x\right)^2, \qquad \overline{\Upsilon} = \tfrac{1}{2}(\Upsilon_1 + \Upsilon_2).$$

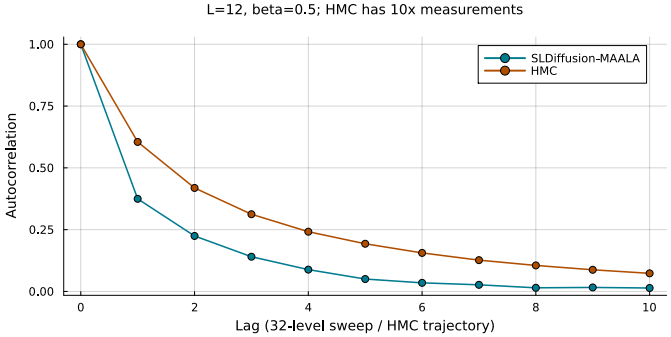


Mean: $0.01846(171)$ for the score trained at $L = 6$, versus $0.01766(85)$ for HMC. The helicity difference is 0.42 combined standard errors.

A. Tomiya, SLDiffusion-MAALA (2026).

Vortex-density correlations also decay rapidly at $L = 12$

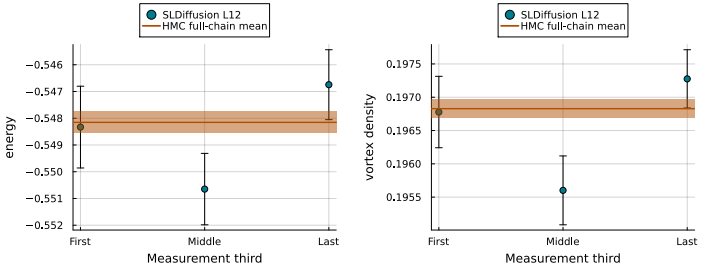
A topological-defect observable in the same independent chains



The SLDiffusion score is retrained at $L = 12$. Lags count recorded trajectories. $\tau_{\mathrm{int}}(\rho_v) = 1.455(33)$ for SLDiffusion, $3.086(31)$ for HMC. Curves are point estimates.

A. Tomiya, SLDiffusion-MAALA (2026).

Split the $L = 12$ measurement series into three contiguous parts



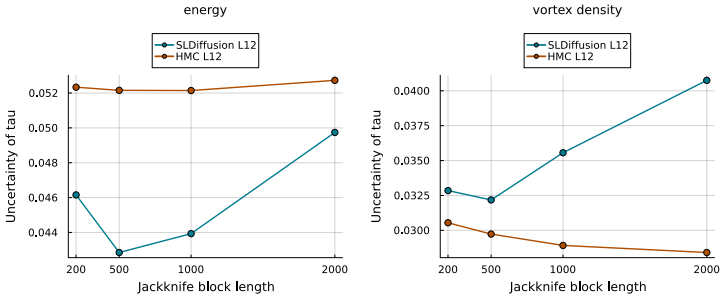
The model is held fixed throughout the measurement chain. The largest pairwise differences between thirds are 2.09 standard errors for energy and 2.47 for vortex density.

Finite-series checks probe the measured observables. They do not turn formal target invariance into a guarantee of exhaustive finite-time exploration.

A. Tomiya, SLDiffusion-MAALA (2026).

Autocorrelation errors are checked against the block length

Contiguous delete-one-block jackknife, with the window reselected in each replicate



Primary block length: 200; checks: 500, 1,000, 2,000. The point estimate uses the full chain, not a bias-corrected jackknife central value.

FFT autocorrelations, maximum lag 5,000, positive self-consistent window with $c = 5$. The same procedure is applied to SLDiffusion and HMC.

A. Tomiya, SLDiffusion-MAALA (2026).

Independent ensemble means at $\beta = 0.50$

Parentheses denote correlated-data standard errors

L	Observable	SLDiffusion	HMC	Difference / combined SE
4	e	$-0.60518(202)$	$-0.60561(115)$	$+0.19$
6	e	$-0.55608(135)$	$-0.55404(73)$	-1.33
8	e	$-0.54894(95)$	$-0.54748(52)$	-1.35
12	e	$-0.54862(81)$	$-0.54815(39)$	-0.51
4	ρ_v	$0.17896(73)$	$0.17825(41)$	$+0.84$
6	ρ_v	$0.19377(49)$	$0.19465(26)$	-1.58
8	ρ_v	$0.19679(35)$	$0.19704(19)$	-0.62
12	ρ_v	$0.19656(29)$	$0.19683(14)$	-0.84

A. Tomiya, SLDiffusion-MAALA (2026).

Integrated autocorrelation times at $\beta = 0.50$

Recorded-trajectory units; error bars condition on each fixed trained model

L	Observable	SLDiffusion	HMC	SLDiffusion / HMC
4	e	$0.937(18)$	$3.199(162)$	$0.293(16)$
6	e	$1.237(29)$	$3.282(33)$	$0.377(10)$
8	e	$1.356(29)$	$3.307(34)$	$0.410(10)$
12	e	$1.932(46)$	$4.344(52)$	$0.445(12)$
4	ρ_v	$0.829(13)$	$2.630(22)$	$0.315(6)$
6	ρ_v	$0.966(18)$	$2.489(22)$	$0.388(8)$
8	ρ_v	$1.030(20)$	$2.469(21)$	$0.417(9)$
12	ρ_v	$1.455(33)$	$3.086(31)$	$0.471(12)$

A. Tomiya, SLDiffusion-MAALA (2026).

$L = 4$, $\beta = 0.30$, matched 50,000-record measurements

Proposal	Acceptance	$\tau_{\mathrm{int}}(e)$	$\tau_{\mathrm{int}}(\rho_v)$
Learned score, 32 levels	0.8090	$0.607(9)$	$0.557(7)$
Analytic force, 32 levels	0.9389	$0.550(7)$	$0.529(5)$
Analytic force, 1 level	0.7697	$3.179(106)$	$2.277(70)$

The learned score gives short correlations within the 32-level corrected construction. The analytic-force control shows that the update schedule itself contributes substantially.

Acceptance is per MH proposal. A 32-level record contains 32 updates; a 1-level record contains 1. These are autocorrelation comparisons in recorded-trajectory units.

A. Tomiya, SLDiffusion-MAALA (2026).

Equal training weights do not make a Markov-chain ensemble independent

$$N_{\mathrm{weight}} = \frac{(\sum_i w_i)^2}{\sum_i w_i^2}.$$

At fixed β, the volume-retraining source already targets the correct action. All training weights are equal, so $N_{\mathrm{weight}} = 50,000$.

$$N_{\mathrm{corr}}(\mathcal{O}) \simeq \frac{N}{2\tau_{\mathrm{int}}(\mathcal{O})}.$$

For the $L = 12$ transfer-chain source:

$N_{\mathrm{corr}}(e) \simeq 12,659$,
$N_{\mathrm{corr}}(\rho_v) \simeq 16,687$.

The effective count depends on the observable.

The importance-weight identity is a population statement. Finite weighted training sets still have sampling error; MH correction fixes the invariant target, not the training sample size.

A. Tomiya, SLDiffusion-MAALA (2026).

Flow-based sampling: selected references

Action-based learning and extensions to lattice field theories

Albergo, Kanwar and Shanahan 1904.12072
Flow-based generative models for Markov chain Monte Carlo in lattice field theory

Kanwar et al. 2003.06413
Equivariant flow-based sampling for lattice gauge theory

Boyda et al. 2008.05456
Sampling using $SU(N)$ gauge equivariant flows

Del Debbio, Marsh Rossney and Wilson 2105.12481
Efficient Modelling of Trivializing Maps for Lattice ϕ^4 Theory Using Normalizing Flows: A First Look at Scalability

Albergo et al. 2106.05934
Flow-based sampling for fermionic lattice field theories

Hackett et al. 2107.00734
Flow-based sampling for multimodal and extended-mode distributions in lattice field theory

Caselle, Cellini, Nada and Panero 2201.08862
Stochastic normalizing flows as non-equilibrium transformations

Gerdes et al. 2207.00283
Learning Lattice Quantum Field Theories with Equivariant Continuous Flows

Abbott et al. 2208.03832
Sampling QCD field configurations with gauge-equivariant flow models

Abbott et al. 2305.02402
Normalizing flows for lattice gauge theory in arbitrary space-time dimension