

Diffusion models for SU(2) lattice gauge theory in two dimensions

H. Alharazin[✉], J. Yu. Panteleeva[✉], and B.-D. Sun

Institut für Theoretische Physik II, Ruhr-Universität Bochum, D-44780 Bochum, Germany



(Received 15 June 2026; accepted 8 July 2026; published 27 July 2026)

We apply score-based diffusion models to two-dimensional SU(2) lattice pure gauge theory with the Wilson action, extending recent work on U(1) gauge theories. The SU(2) manifold structure is handled through a quaternion parametrization. The model is trained on 10,000 configurations generated via hybrid Monte Carlo at a fixed coupling $\beta_0 = 2.0$ on an 8×8 lattice, augmented to 20,000 samples via random gauge transformations. Through physics-conditioned sampling exploiting the linear β dependence of the score function, we generate configurations at different values of the coupling without retraining; through the fully convolutional U-Net architecture with periodic boundary conditions, we generate configurations on lattices of different spatial extents. We validate our approach by comparing the average plaquette against exact analytical predictions. At the training lattice size (8×8), the model reproduces the exact plaquette with biases $|\Delta| \leq 0.001$ for $\beta \in [1.5, 2.5]$ and $|\Delta| < 0.06$ across $\beta \in [1, 4]$. For lattices sharing the training extent $L = 8$ in at least one direction, biases remain below ~ 0.003 for $\beta \in [1.5, 2.5]$, with larger deviations at higher couplings. This work demonstrates that diffusion models are a promising tool for non-Abelian gauge field generation and motivates further investigation toward higher-dimensional theories.

DOI: [10.1103/n3bh-gs98](https://doi.org/10.1103/n3bh-gs98)

I. INTRODUCTION

Lattice gauge theory provides a rigorous nonperturbative framework for studying quantum chromodynamics (QCD) and other gauge theories [1,2]. The fundamental computational challenge lies in sampling gauge field configurations from the Boltzmann distribution $p(U) \propto e^{-S[U]}$, where S denotes the Euclidean action. Traditional Markov chain Monte Carlo methods, particularly the hybrid Monte Carlo (HMC) algorithm [3], have been the workhorse of lattice QCD for decades. However, these methods face severe limitations: critical slowing down near continuum, topological freezing at fine lattice spacings [4], and the notorious sign problem for finite baryon density [5].

The emergence of machine learning approaches offers promising avenues to address these challenges. Normalizing flows have been successfully applied to scalar field theories [6] and gauge theories [7–9], demonstrating the potential of generative models in lattice field theory. More recently, diffusion models—a class of generative models that learn to reverse a gradual noising process—have achieved remarkable success in image generation [10,11] and have been proposed for lattice gauge theory applications [12–18].

The work of Zhu *et al.* [14,15] established a diffusion model framework for two-dimensional U(1) gauge theory, demonstrating several attractive properties: the ability to generate configurations at different couplings through physics-conditioned sampling, generalization to different lattice sizes via a fully convolutional architecture, and avoidance of topological freezing. A framework with these capabilities, training in a regime where HMC is efficient and extending to regions where it suffers from critical slowing down or topological freezing, is particularly promising and warrants further investigation. However, the extension to non-Abelian gauge theories, essential for QCD applications, remained unexplored. During the preparation of this paper, Aarts *et al.* [18] addressed this extension using an equivariant diffusion framework; a detailed comparison between this work and the work of Ref. [18] is presented in Section V E.

In this work, we develop a diffusion model for two-dimensional SU(2) pure gauge theory. The transition from U(1) to SU(2) introduces several conceptual and technical challenges. First, the configuration space changes fundamentally: U(1) link variables are phases on the circle S^1 , while SU(2) link variables are elements of a three-dimensional manifold isomorphic to the 3-sphere S^3 . Second, the noncommutativity of SU(2) affects the structure of the theory, including the form of the action gradient. We address these challenges through a quaternion parametrization that naturally respects the SU(2) manifold structure. Notably, gauge equivariance is not built into the architecture but must be learned from data; while it seems natural to

Published by the American Physical Society under the terms of the [Creative Commons Attribution 4.0 International](https://creativecommons.org/licenses/by/4.0/) license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI. Funded by SCOAP³.

incorporate gauge invariance from the outset, as done in Ref. [18], it is instructive that a standard convolutional network can learn to approximate the correct gauge-field distributions without this constraint. Moreover, since our uncorrected diffusion approach never evaluates the action during sampling, it may offer a starting point for future explorations of theories with a complex action, where Metropolis-based corrections are not directly applicable.

This paper is organized as follows. Section II reviews the mathematical framework of two-dimensional SU(2) lattice gauge theory. Section III presents the diffusion model formalism adapted to gauge field configurations. Section IV describes the neural network architecture and training procedure. Section V discusses our numerical results, including comparisons with exact analytical predictions and with the equivariant approach of Ref. [18]. Section VI outlines future directions.

II. SU(2) LATTICE GAUGE THEORY: MATHEMATICAL FRAMEWORK

We begin by establishing the mathematical foundations of two-dimensional SU(2) lattice gauge theory, which forms the physical basis for our diffusion model approach.

A. The SU(2) group and quaternion representation

The special unitary group SU(2) consists of all 2×2 complex matrices U satisfying

$$U^\dagger U = \mathbb{1}, \quad \det(U) = 1. \quad (1)$$

As a manifold, SU(2) is isomorphic to the 3-sphere $S^3 \subset \mathbb{R}^4$, which motivates the quaternion parametrization. Every SU(2) element can be uniquely written as

$$U = a_0 \mathbb{1} + i(a_1 \sigma_1 + a_2 \sigma_2 + a_3 \sigma_3), \quad (2)$$

where σ_i are the Pauli matrices,

$$\sigma_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_2 = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \sigma_3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad (3)$$

and the quaternion components $(a_0, a_1, a_2, a_3) \in \mathbb{R}^4$ satisfy the constraint

$$a_0^2 + a_1^2 + a_2^2 + a_3^2 = 1. \quad (4)$$

This constraint defines the 3-sphere S^3 embedded in $\mathbb{R}^4$. In explicit matrix form, Eq. (2) becomes

$$U = \begin{pmatrix} a_0 + ia_3 & a_2 + ia_1 \\ -a_2 + ia_1 & a_0 - ia_3 \end{pmatrix}. \quad (5)$$

The quaternion representation is particularly advantageous for numerical implementations: group multiplication reduces

to quaternion multiplication, the inverse is obtained by negating the vector part $(a_1, a_2, a_3) \rightarrow -(a_1, a_2, a_3)$, and the trace is simply $\text{Tr}(U) = 2a_0$. Moreover, random sampling according to the Haar measure is achieved by drawing four independent Gaussian random numbers and normalizing to unit length, exploiting the uniformity of the Gaussian measure on S^3 .

B. Lattice gauge field configuration

On a two-dimensional Euclidean lattice Λ of size $L_x \times L_t$ with periodic boundary conditions, a gauge field configuration consists of link variables $U_\mu(x) \in \text{SU}(2)$ associated with each oriented link from site x to site $x + \hat{\mu}$, where $\hat{\mu}$ denotes the unit vector in direction $\mu \in \{0, 1\}$ (spatial and temporal directions, respectively). The complete configuration is thus a collection,

$$\{U_\mu(x) : x \in \Lambda, \mu \in \{0, 1\}\}, \quad (6)$$

containing $2 \times L_x \times L_t$ SU(2) matrices. Using the quaternion representation, we store this as a tensor

$$\mathcal{U} \in \mathbb{R}^{L_x \times L_t \times 2 \times 4}, \quad (7)$$

where the last two indices correspond to direction μ and quaternion component i , respectively.

The fundamental gauge-invariant object is the plaquette, the elementary closed loop around a single lattice cell,

$$P_{01}(x) = U_0(x)U_1(x + \hat{0})U_0^\dagger(x + \hat{1})U_1^\dagger(x), \quad (8)$$

where the product is taken around the elementary square in the counter-clockwise direction. Since $P_{01}(x) \in \text{SU}(2)$, it can also be represented as a quaternion $(a_0^{(\text{plaq})}, a_1^{(\text{plaq})}, a_2^{(\text{plaq})}, a_3^{(\text{plaq})})$.

C. Wilson action and observables

The Wilson gauge action for two-dimensional SU(2) lattice gauge theory is defined as

$$S[U] = -\frac{\beta}{2} \sum_{x \in \Lambda} \text{ReTr}[P_{01}(x)] = -\beta \sum_{x \in \Lambda} a_0^{(\text{plaq})}(x), \quad (9)$$

where $\beta = 4/g^2$ is the inverse gauge coupling (with g the bare coupling constant), and $a_0^{(\text{plaq})}(x)$ is the a_0 component of the plaquette quaternion at site x . The partition function is

$$Z = \int \mathcal{D}U e^{-S[U]}, \quad (10)$$

where the integration measure $\mathcal{D}U = \prod_{x,\mu} dU_\mu(x)$ is the product of Haar measures on SU(2). Physical observables are computed as thermal averages,

$$\langle O \rangle = \frac{1}{Z} \int \mathcal{D}U O[U] e^{-S[U]}. \quad (11)$$

The average plaquette is the most fundamental observable,

$$\langle P \rangle = \left\langle \frac{1}{2} \text{ReTr}[P_{01}] \right\rangle = \langle a_0^{(\text{plaq})} \rangle. \quad (12)$$

For two-dimensional SU(2) gauge theory, this has the exact analytical result [2,19],

$$\langle P \rangle = \frac{I_2(\beta)}{I_1(\beta)}, \quad (13)$$

where $I_n(\beta)$ denotes the modified Bessel function of the first kind. Crucially, this result holds exactly on any finite periodic lattice, regardless of L_x and L_t : in two dimensions with periodic boundary conditions, the single-plaquette integrals factorize in the partition function after gauge fixing to a maximal tree, so $\langle P \rangle$ receives no finite-volume corrections [2,19]. On a finite lattice, the average plaquette is computed as

$$\langle P \rangle = \frac{1}{L_x L_t} \sum_{x \in \Lambda} \frac{1}{2} \text{ReTr}[P_{01}(x)] = \frac{1}{L_x L_t} \sum_{x \in \Lambda} a_0^{(\text{plaq})}(x). \quad (14)$$

III. DIFFUSION MODELS: THEORETICAL FRAMEWORK

Diffusion models are a class of generative models that learn to reverse a gradual noising process. In this section, we present the theoretical framework adapted to lattice gauge field configurations.

A. Forward diffusion process

Given an unnoised gauge field configuration represented as $\varphi_0 \in \mathbb{R}^{8 \times L_x \times L_t}$ (reshaped from the quaternion tensor, where 8 channels = 2 directions $\times$ 4 quaternion components), the forward process progressively adds Gaussian noise over T time steps according to the Markov chain,

$$q(\varphi_t | \varphi_{t-1}) = \mathcal{N}(\varphi_t; \sqrt{1 - \beta_t} \varphi_{t-1}, \beta_t \mathbb{1}), \quad (15)$$

where $\mathcal{N}(\varphi_t; \mu, \Sigma)$ denotes a Gaussian distribution over φ_t with mean μ and covariance Σ , and $\{\beta_t\}_{t=1}^T$ is a variance schedule with $0 < \beta_t < 1$.

A key property of Gaussian diffusion is that φ_t can be sampled directly from φ_0 without iterating through intermediate steps,

$$q(\varphi_t | \varphi_0) = \mathcal{N}(\varphi_t; \sqrt{\bar{\alpha}_t} \varphi_0, (1 - \bar{\alpha}_t) \mathbb{1}), \quad (16)$$

where we define

$$\alpha_t = 1 - \beta_t, \quad \bar{\alpha}_t = \prod_{s=1}^t \alpha_s. \quad (17)$$

This allows direct sampling via the reparametrization,

$$\varphi_t = \sqrt{\bar{\alpha}_t} \varphi_0 + \sqrt{1 - \bar{\alpha}_t} \epsilon, \quad \epsilon \sim \mathcal{N}(0, \mathbb{1}). \quad (18)$$

We employ the cosine schedule [20],

$$\bar{\alpha}_t = \cos^2 \left(\frac{\pi}{2} \cdot \frac{t/T + s}{1 + s} \right), \quad (19)$$

with small offset $s = 0.008$. This schedule ensures a smooth progression from $\bar{\alpha}_0 \approx 1$ (pure signal) to $\bar{\alpha}_T \approx 0$ (pure noise), avoiding the sharp transitions of linear schedules that can destabilize training.

B. Reverse diffusion process

The reverse process learns to denoise by modeling the posterior $p_\theta(\varphi_{t-1} | \varphi_t)$. Following the denoising diffusion probabilistic models (DDPM) framework [10], we parametrize this as a Gaussian,

$$p_\theta(\varphi_{t-1} | \varphi_t) = \mathcal{N}(\varphi_{t-1}; \mu_\theta(\varphi_t, t), \beta_t \mathbb{1}), \quad (20)$$

where the mean is predicted by a neural network. Rather than predicting μ_θ directly, we train the network to predict the noise $\epsilon_\theta(\varphi_t, t)$, from which the mean is computed as

$$\mu_\theta(\varphi_t, t) = \frac{1}{\sqrt{\alpha_t}} \left(\varphi_t - \frac{\beta_t}{\sqrt{1 - \bar{\alpha}_t}} \epsilon_\theta(\varphi_t, t) \right). \quad (21)$$

The network is trained to minimize the denoising score matching objective,

$$\mathcal{L} = \mathbb{E}_{t \sim \mathcal{U}(1, T), \varphi_0 \sim p_{\text{data}}, \epsilon \sim \mathcal{N}(0, \mathbb{1})} [\|\epsilon - \epsilon_\theta(\varphi_t, t)\|^2], \quad (22)$$

where φ_t is constructed from φ_0 and ϵ via Eq. (18).

C. Connection to stochastic quantization

Diffusion models have a deep connection to stochastic quantization [21], an alternative approach to quantizing field theories. In stochastic quantization, a field φ evolves in a fictitious Langevin time τ according to the equation

$$\frac{\partial \varphi}{\partial \tau} = -\frac{\delta S[\varphi]}{\delta \varphi} + \sqrt{2} \eta(\tau), \quad (23)$$

where η is Gaussian white noise with $\langle \eta(\tau) \eta(\tau') \rangle = \delta(\tau - \tau')$. As $\tau \rightarrow \infty$, the distribution of φ converges to the Boltzmann distribution $p(\varphi) \propto e^{-S[\varphi]}$.

The reverse diffusion process can be understood as a learned version of Langevin dynamics. The score function,

the gradient of the log probability, is related to the action gradient,

$$\nabla_{\varphi} \log p(\varphi) = -\nabla_{\varphi} S[\varphi]. \quad (24)$$

The neural network ϵ_{θ} learns an approximation to this score function, enabling sampling from the target distribution. This connection provides a physics-based foundation for the diffusion process and suggests natural extensions to non-Abelian gauge theories through the established framework of stochastic quantization for gauge fields [22]. This connection between diffusion models and stochastic quantization in the context of lattice field theory was developed in Refs. [12,13] and exploited for gauge field simulations in Refs. [14,15].

D. Physics-conditioned sampling

A key advantage of our approach is the ability to generate configurations at coupling values β different from the training coupling β_0 , without retraining the model. Theoretically, this follows from the structure of the Wilson action and the connection to stochastic quantization.

Since $S[U] = \beta \tilde{S}[U]$ where $\tilde{S} = -\sum_x a_0^{\text{plaq}}(x)$ is independent of β , the score function scales linearly with the coupling,

$$\nabla_{\varphi} \log p^{(\beta)}(\varphi) = -\beta \nabla_{\varphi} \tilde{S}[\varphi]. \quad (25)$$

A model trained at β_0 learns the score $s^{(\beta_0)} \propto -\beta_0 \nabla \tilde{S}$. To generate configurations at coupling β , we rescale the predicted noise,¹

$$\hat{\epsilon}^{(\beta)} = \frac{\beta}{\beta_0} \hat{\epsilon}^{(\beta_0)}. \quad (26)$$

In practice, when extrapolating to couplings far from β_0 , we stabilize the sampling by interpolating between a constant baseline and the rescaled network prediction,

$$\hat{\epsilon}^{(\beta)} = (1 - \lambda) + \lambda \cdot \frac{\beta}{\beta_0} \hat{\epsilon}^{(\beta_0)}, \quad (27)$$

where $\lambda \in [0.15, 0.5]$ controls the network contribution. Results are robust across this range, with smaller values preferred for $\beta \gg \beta_0$, where the sampling initialization is also adapted accordingly. This allows exploration of various coupling regions without generating new training data, dramatically reducing computational cost.

¹This physics-conditioned approach was introduced in Refs. [14,15], building on the framework of Refs. [12,13].

IV. NEURAL NETWORK ARCHITECTURE AND IMPLEMENTATION

A. Data representation

The gauge field configuration tensor $\mathcal{U} \in \mathbb{R}^{L_x \times L_t \times 2 \times 4}$ is reshaped to $\varphi \in \mathbb{R}^{8 \times L_x \times L_t}$, where the 8 channels correspond to the $2 \times 4 = 8$ quaternion components (2 directions $\times$ 4 components each²). This representation treats the gauge field as an “8-channel image” over the $L_x \times L_t$ lattice, enabling the use of standard convolutional architectures while preserving the full gauge field information.

B. Training data generation and gauge augmentation

Training configurations are generated using the hybrid Monte Carlo algorithm [3] at a fixed coupling β_0 . The HMC algorithm combines molecular dynamics evolution with a Metropolis accept/reject step, providing exact sampling from the Boltzmann distribution while maintaining reasonable acceptance rates.

The gauge symmetry of lattice gauge theory provides a powerful mechanism for data augmentation. Given a configuration U , we generate additional training samples by applying random gauge transformations,

$$U'_{\mu}(x) = \Omega(x) U_{\mu}(x) \Omega^{\dagger}(x + \hat{\mu}), \quad (28)$$

where $\{\Omega(x)\}$ are independent random SU(2) elements sampled according to the Haar measure. In practice, we generate Haar-random SU(2) elements by sampling four independent Gaussian random numbers and normalizing to unit quaternion,

$$\Omega \leftrightarrow \frac{(g_0, g_1, g_2, g_3)}{\sqrt{g_0^2 + g_1^2 + g_2^2 + g_3^2}}, \quad g_i \sim \mathcal{N}(0, 1). \quad (29)$$

All gauge-transformed configurations share identical physical content, the same action, plaquette values, Wilson loops, and any gauge-invariant observable but have different numerical values for the link variables. This augmentation teaches the network the gauge equivalence of configurations and improves generalization. For each of the 10,000 HMC configurations, we generate one gauge-transformed copy, yielding a total of 20,000 training samples.

C. U-Net architecture

The noise prediction network $\epsilon_{\theta}(\varphi_t, t)$ is implemented as a U-Net architecture [23] with specific adaptations for lattice gauge fields. The overall structure of our diffusion model pipeline is illustrated in Fig. 1.

The network consists of an encoder path that progressively downsamples the spatial dimensions while increasing feature channels, a bottleneck, and a decoder path that upsamples

²The channel ordering is $\varphi = [\vec{a}^{(\mu=0)}, \vec{a}^{(\mu=1)}]$.

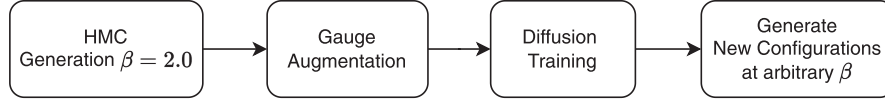


FIG. 1. Complete pipeline of the diffusion model approach. Training data are generated via HMC at coupling β_0 , augmented through random gauge transformations, and used to train the diffusion model. Once trained, the model generates new configurations at arbitrary coupling β and lattice extents without retraining.

back to the original resolution with skip connections from the encoder. The architecture is detailed in Fig. 2.

1. Periodic boundary conditions

Lattice gauge theory is conventionally formulated on a torus with periodic boundary conditions. Standard neural network padding schemes (zero padding, reflection padding) would introduce artificial boundary effects and break translational invariance. We instead use *circular padding*, which wraps the input around so that sites at opposite boundaries are correctly identified as neighbors.

For a convolutional kernel of size k with padding $p = \lfloor k/2 \rfloor$, circular padding extends the input by copying values from the opposite boundary,

$$\varphi_{\text{padded}}[i] = \varphi[(i - p) \bmod L], \quad (30)$$

where L is the lattice extent. This ensures translational invariance is exactly preserved by the convolution operation, which is essential for the fully convolutional architecture to generalize to different lattice sizes.

2. Time embedding

The diffusion time step t is encoded using sinusoidal positional embeddings [24], which map the scalar time step to a high-dimensional vector,

$$\text{PE}^{2i}(t) = \sin\left(\frac{t}{10^{6i/d}}\right), \quad \text{PE}^{2i+1}(t) = \cos\left(\frac{t}{10^{6i/d}}\right), \quad (31)$$

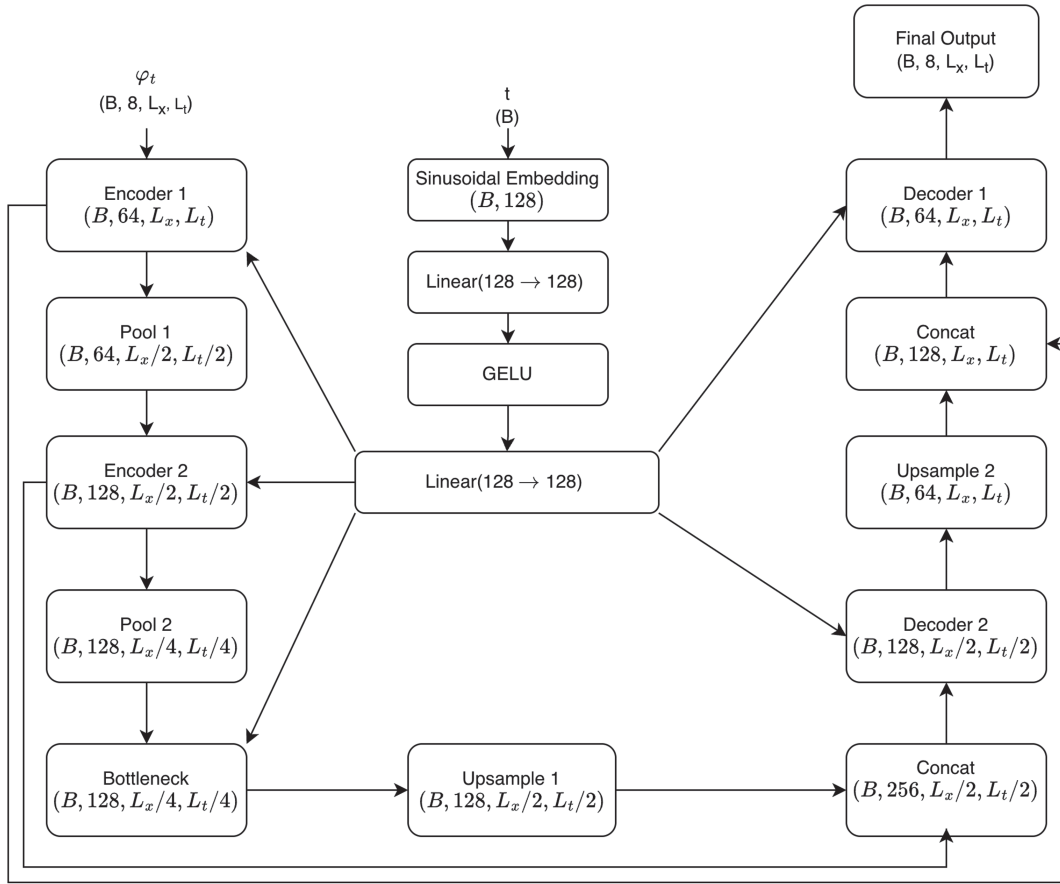


FIG. 2. Detailed architecture of the U-Net neural network ϵ_θ . The network takes the noised configuration φ_t and time step t as inputs and predicts the noise ϵ_{pred} . The encoder-decoder structure with skip connections enables multiscale feature extraction while maintaining spatial resolution in the output.

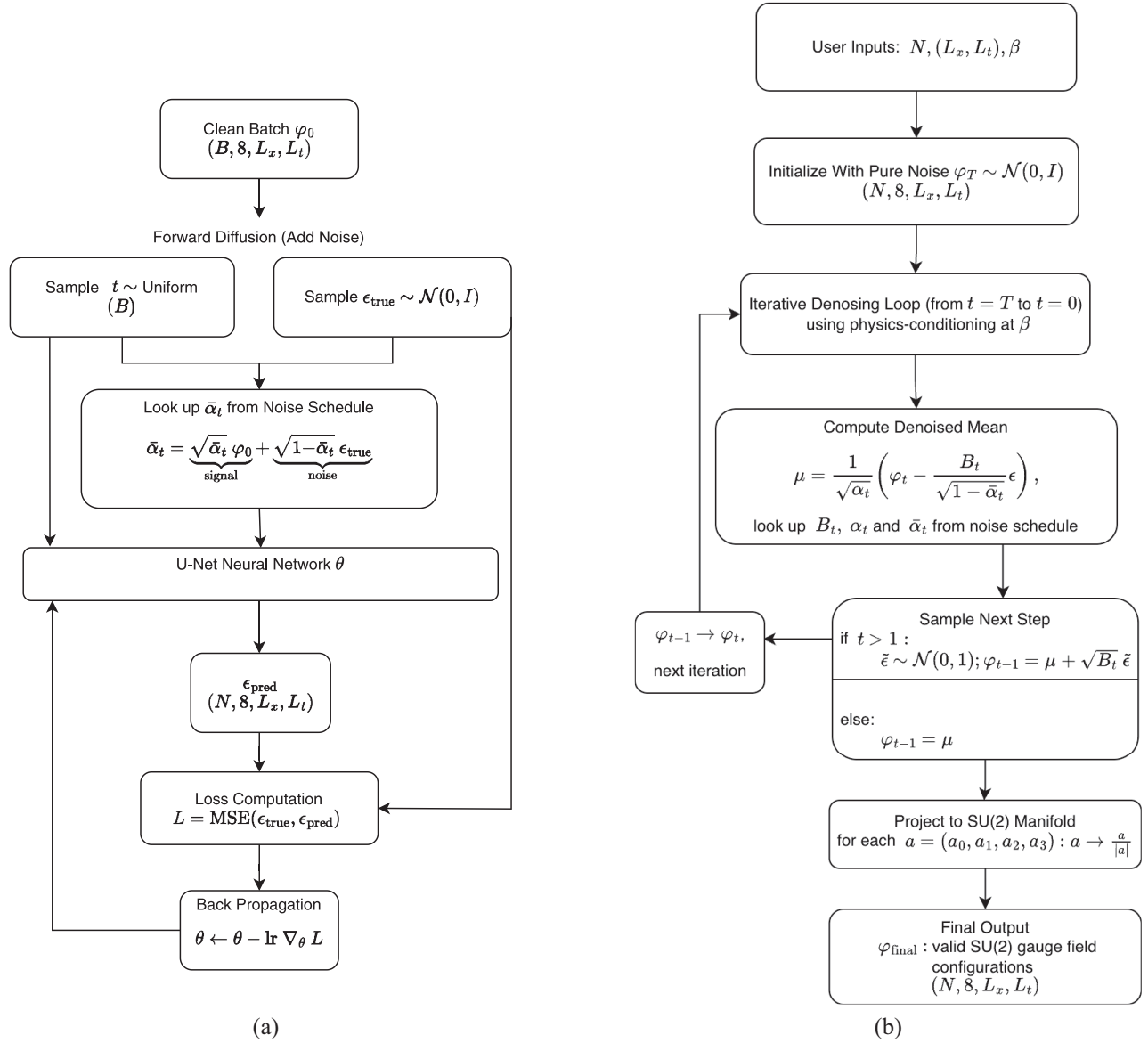


FIG. 3. Overview of the training and sampling procedures. (a) Flowchart of a single training step. The model learns to predict the noise ϵ_{true} that was added to clean configurations, enabling the reverse denoising process during sampling. (b) Sampling algorithm flowchart. Starting from Gaussian noise, the iterative denoising loop applies physics-conditioned noise prediction and gradually recovers a gauge field configuration. The final projection step ensures the output satisfies the SU(2) constraint.

where d is the embedding dimension (we use $d = 128$). This embedding is processed through a multilayer perceptron with the Gaussian error linear unit (GELU) activations, and added to the feature maps in each residual block, allowing the network to condition its behavior on the noise level.

3. Residual blocks

Each encoder and decoder level consists of convolutional residual blocks with the following structure:

- (1) GroupNorm $\rightarrow$ GELU activation $\rightarrow$ Conv2d (circular padding)
- (2) Add time embedding (projected to appropriate channel dimension)
- (3) GroupNorm $\rightarrow$ GELU activation $\rightarrow$ Conv2d (circular padding)
- (4) Residual connection (with 1×1 convolution if channel dimensions differ)

GroupNorm [25] is preferred over BatchNorm for the small batch sizes typical in physics applications, as it normalizes over channel groups rather than requiring large batch statistics.

D. Training procedure

The training procedure follows the standard DDPM approach, as illustrated in Fig. 3(a). For each batch of clean configurations φ_0 :

- (1) Sample random time steps $t^i \sim \mathcal{U}(1, T)$ independently for each configuration i in the batch
- (2) Sample random noise $\epsilon_{\text{true}}^i \sim \mathcal{N}(0, 1)$
- (3) Construct noised configurations $\varphi_t^i = \sqrt{\bar{\alpha}_t} \varphi_0^i + \sqrt{1 - \bar{\alpha}_t} \epsilon_{\text{true}}^i$
- (4) Forward pass through the network to predict $\epsilon_{\text{pred}}^i = \epsilon_\theta(\varphi_t^i, t^i)$
- (5) Compute MSE loss $\mathcal{L} = \frac{1}{B} \sum_i \|\epsilon_{\text{true}}^i - \epsilon_{\text{pred}}^i\|^2$
- (6) Backpropagate and update network parameters

We use the Adam optimizer [26] with learning rate 10^{-4} and cosine annealing schedule. Training is performed for 100 epochs with batch size 32. Gradient clipping with maximum norm 1.0 prevents training instabilities.

E. Sampling algorithm

To generate new configurations, we reverse the diffusion process starting from pure noise, as illustrated in Fig. 3(b).

The final quaternion normalization step projects the generated configuration back onto the SU(2) manifold, correcting any small deviations from unit norm accumulated during sampling. This projection is essential for ensuring the output represents valid SU(2) gauge links.

V. RESULTS

We now present our numerical results, investigating the model's ability to generate configurations that reproduce exact analytical predictions across a range of couplings and lattice sizes.

A. Validation at training coupling

We first validate the model at the training coupling $\beta_0 = 2.0$ on the training geometry 8×8 . Generating 500 configurations, we compute the average plaquette and compare with the exact analytical prediction Eq. (13).

The diffusion model yields $\langle P \rangle_{\text{diff}} = 0.4329 \pm 0.0001(\text{stat})$, where the statistical uncertainty σ_{stat} is the autocorrelation-corrected standard error of the mean from 500 configurations. The exact value is $\langle P \rangle_{\text{exact}} = I_2(2)/I_1(2) = 0.4331$, giving a bias $\Delta \equiv \langle P \rangle_{\text{diff}} - \langle P \rangle_{\text{exact}} = -0.0002$, well within statistical precision.

Throughout this work, since the exact answer is known analytically, we report both σ_{stat} and the bias Δ for each $(\beta, L_x \times L_t)$ point as the most transparent diagnostics of model quality. As established in Eq. (13), the exact plaquette $I_2(\beta)/I_1(\beta)$ holds on any finite periodic lattice, independent of L_x and L_t : in two dimensions, the single-plaquette expectation values factorize after gauge fixing to a maximal tree [2,19]. Consequently, a nonzero Δ at any lattice geometry is a direct measure of model error.

Figure 4(a) shows the plaquette distribution for the training point. The distribution is well centered on the exact value, with the sample mean nearly coinciding with it, confirming that the model has learned the correct equilibrium distribution at $(\beta_0, 8 \times 8)$.

B. Physics-conditioned generation at different couplings

A central capability of our approach is generating configurations at couplings different from the training value. Using physics-conditioned sampling (Sec. III D), we generate 500 configurations for $\beta \in \{1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0\}$ on the *training geometry* 8×8 and compare with exact predictions.

Figure 5(a) shows the average plaquette as a function of β . Near the training coupling, the model achieves excellent agreement: at $\beta = 1.5, 2.0$, and 2.5 , the biases are $|\Delta| \leq 0.0004$. At the boundaries of the explored coupling range, larger biases appear: $|\Delta| = 0.017$ at $\beta = 1.0$ and $|\Delta| = 0.051$ at $\beta = 3.0$, with intermediate values $|\Delta| \approx 0.01\text{--}0.03$ at $\beta = 3.5$ and 4.0 . This demonstrates that physics-conditioned rescaling captures the coupling dependence of the score function well near the training point, with accuracy degrading gradually at larger $|\beta - \beta_0|$.

C. Lattice size generalization

The fully convolutional architecture with periodic padding enables generation on lattices of different sizes without retraining. Since $\langle P \rangle$ is exactly volume independent in two dimensions (Sec. II C), the bias Δ on a nontraining geometry directly measures how well the model has learned to reproduce the correct local gauge-field structure beyond its training domain. We test this by generating configurations on lattices with extents $L_x, L_t \in \{8, 12, 16, 32\}$, all using the model trained on 8×8 configurations, across the full range of couplings $\beta \in \{1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0\}$.

A clear hierarchy emerges, governed by how close the lattice geometry is to the training size:

- (i) $[L_x, L_t] = [8 \times 8]$: biases $|\Delta| \leq 0.001$ for $\beta \in [1.5, 2.5]$, growing to $|\Delta| \approx 0.02\text{--}0.05$ at the extremes of the coupling range.
- (ii) $[L_x, L_t] \in \{[8 \times 12], [12 \times 8], [8 \times 16], [16 \times 8]\}$: biases $|\Delta| \lesssim 0.003$ for $\beta \in [1.5, 2.5]$, growing significantly for $\beta \geq 3$ and also elevated at $\beta = 1.0$ ($|\Delta| \approx 0.02$).
- (iii) $[12 \times 12], [12 \times 16], [16 \times 12]$: moderate biases $|\Delta| \sim 0.02\text{--}0.07$ at low β , reaching $|\Delta| > 0.2$ at higher couplings.
- (iv) $[16 \times 16]$: near-exact agreement ($|\Delta| < 0.002$) for $\beta \in [1.5, 2.5]$, with large deviations at higher couplings and at $\beta = 1.0$ ($|\Delta| \approx 0.04$).
- (v) $[32 \times 32]$: significant biases $|\Delta| > 0.15$ at most couplings.

Figures 5(b) and 5(c) illustrate this progression. For the 12×8 lattice [Fig. 5(b)], which shares the training extent $L = 8$ in one direction, the model tracks the exact curve

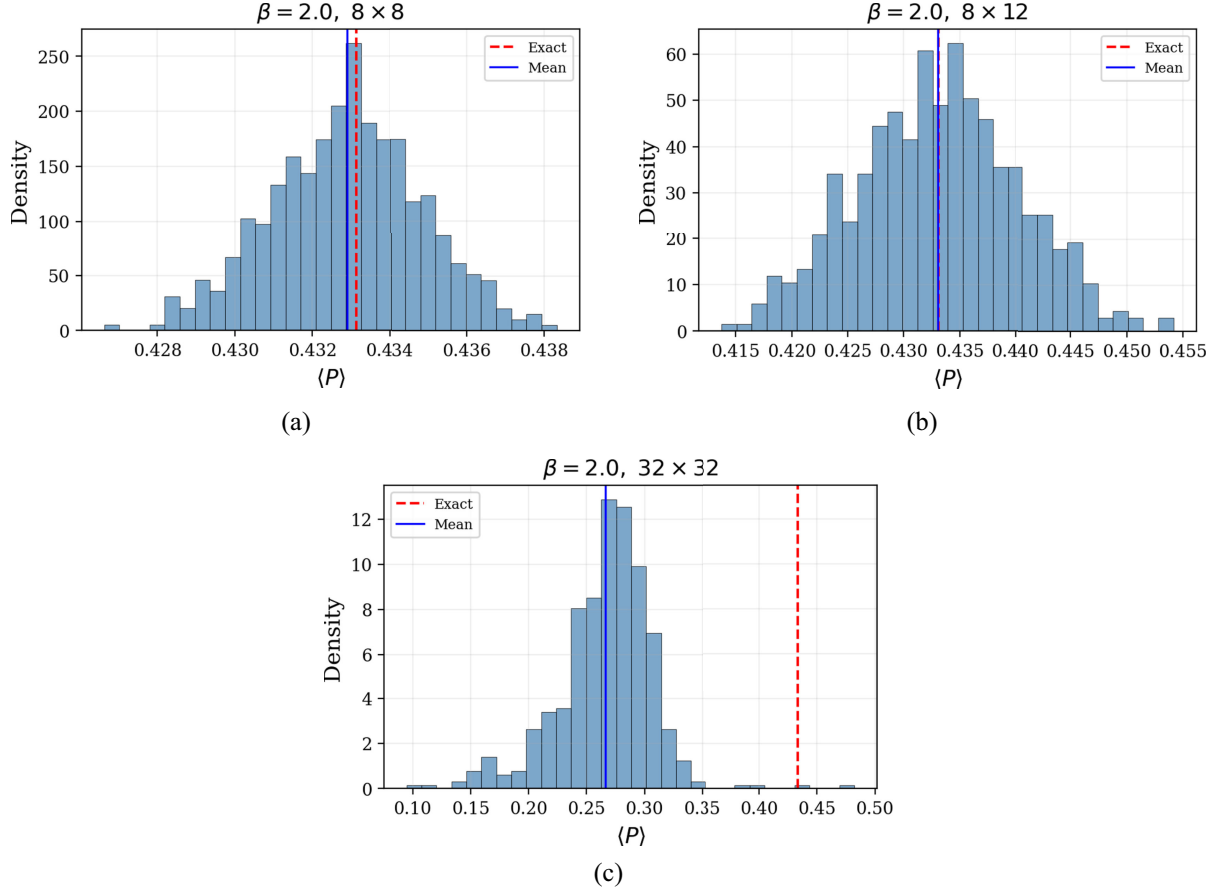


FIG. 4. Plaquette distribution from 500 diffusion-generated configurations at $\beta = 2.0$. Red dashed line: exact value $I_2(\beta)/I_1(\beta)$; blue line: sample mean. (a) 8×8 training lattice. The distribution is well centered on the exact result. (b) 8×12 lattice. The distribution remains centered near the exact result, confirming successful generalization to geometries slightly larger than the training lattice. (c) 32×32 lattice. The distribution is visibly shifted from the exact value, reflecting the difficulty of generalizing to lattice sizes much larger than the 8×8 training geometry.

well up to $\beta \approx 3$. For the 16×16 lattice [Fig. 5(c)], near-exact agreement is achieved for $\beta \in [1.5, 2.5]$, with deviations appearing at higher couplings and at $\beta = 1.0$.

The plaquette distributions further illustrate the volume dependence. Figure 4(b) shows the 8×12 distribution at $\beta = 2.0$, which remains well centered on the exact value. In contrast, Fig. 4(c) for the 32×32 lattice shows a visible shift, with the distribution peaked well below the exact result.

This behavior is expected: the model learns local correlations at the scale of the training lattice, and larger volumes require the network to extrapolate beyond its training domain. Nevertheless, the successful generalization to lattices up to 16×16 , four times the training volume, demonstrates the utility of the fully convolutional approach.

D. HMC autocorrelation analysis

To verify that the HMC-generated configurations are statistically independent, we computed the normalized autocorrelation function $R(t)$ of the average plaquette for

each (β, L_x, L_t) point and extracted the integrated autocorrelation time τ_{int} . Figure 6 shows τ_{int} as a function of β for all lattice sizes. Across the entire parameter grid, $\tau_{\text{int}} < 1$, confirming that consecutive HMC trajectories produce effectively independent configurations, and no thinning is required.

E. Comparison with equivariant diffusion models

During the preparation of this paper, Aarts *et al.* [18] presented a gauge equivariant diffusion framework for two-dimensional U(2) and SU(2) gauge theories. Since both works address the same physical system with diffusion models trained at $\beta_0 = 2$, a detailed comparison is instructive.

The two approaches differ in three fundamental respects. First, the treatment of gauge symmetry: Reference [18] formulates the diffusion process directly on the group manifold via a Stratonovich stochastic differential equation [Eq. (4) therein] and employs lattice gauge equivariant convolutional neural networks (L-CNNs) [27] as the score

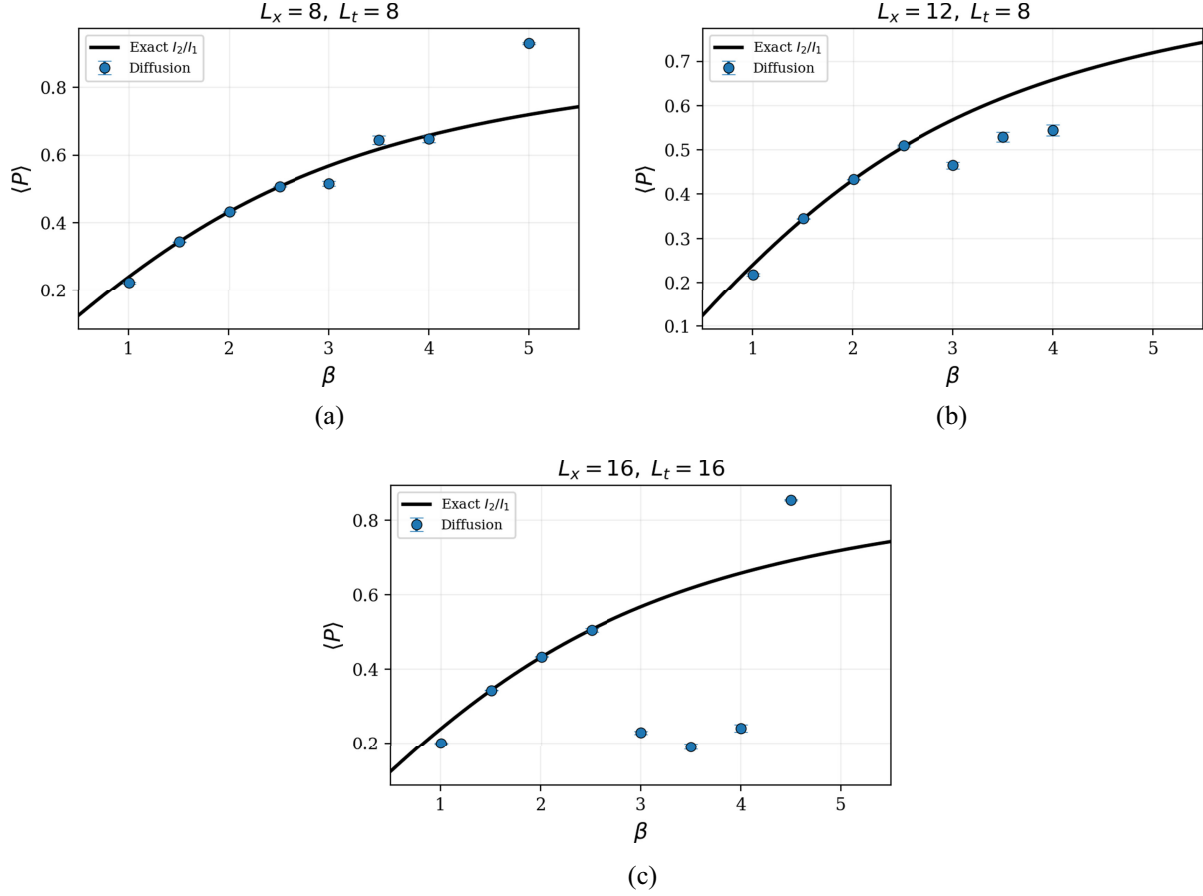


FIG. 5. Average plaquette $\langle P \rangle$ as a function of coupling β . Blue circles: diffusion model results with σ_{stat} error bars; solid line: exact analytical result $I_2(\beta)/I_1(\beta)$. (a) 8×8 training lattice. Biases are $|\Delta| \leq 0.001$ near the training coupling and remain below 0.06 across $\beta \in [1, 4]$. (b) 12×8 lattice, which shares the training extent $L = 8$. The model achieves good agreement across the range $\beta \in [1, 4]$. (c) 16×16 lattice. The model achieves near-exact agreement for $\beta \in [1.5, 2.5]$.

network (Sec. “Gauge equivariant diffusion models” therein), thereby enforcing exact local gauge equivariance at the architectural level through the transformation law of Eq. (6) therein. Our approach instead works in the flat quaternion representation (a_0, a_1, a_2, a_3) of SU(2) and uses

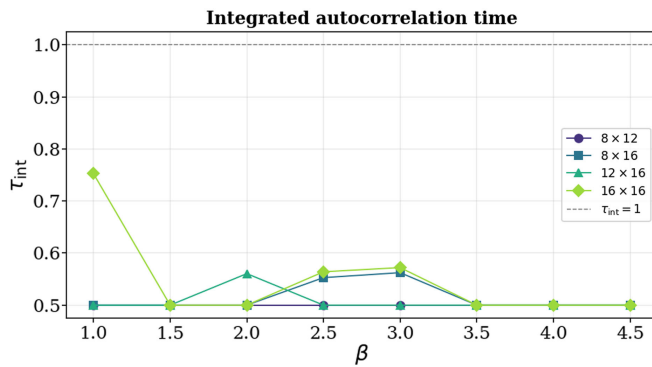


FIG. 6. Integrated autocorrelation time $\tau_{\text{int}} = \frac{1}{2} + \sum_{t=1}^{\infty} R(t)$ of the average plaquette as a function of β for all lattice sizes. All values remain near the theoretical minimum $\tau_{\text{int}} = 0.5$, indicating that consecutive configurations are effectively uncorrelated.

a standard convolutional U-Net, relying on the unit-quaternion projection to return configurations to the group manifold. Gauge equivariance is not enforced but must be learned from data.

Second, the sampling algorithm: Reference [18] employs the Metropolis-adjusted annealed Langevin algorithm [MAALA, Eqs. (11) and (12) therein], which applies a Metropolis accept/reject step based on the target Wilson action during the final 2% of diffusion time steps (End Matter, “MAALA implementation details” therein). This correction guarantees that generated samples are asymptotically distributed according to $\exp(-S_E)$ at the target β , regardless of residual model errors. Our approach uses uncorrected reverse diffusion with physics-conditioned noise rescaling (Sec. III D), so the quality of the generated ensemble depends entirely on the accuracy of the learned score function.

Third, the training setup differs in scale: Reference [18] trains on 10^5 HMC configurations at $\beta_0 = 2$ on a 16×16 lattice using an ensemble of ten independently initialised networks with $\sim 15,000$ parameters each (Secs. “Numerical results” and “Training and model details” therein), while

our model is trained on a smaller set at $\beta_0 = 2$ on an 8×8 lattice with a single network.

Despite these differences, qualitative similarities emerge. Both models successfully generalize to couplings away from the training point using score rescaling: Reference [18] reports accurate Wilson loop predictions up to $\beta \approx 14$ on $L = 16$ (Fig. 3 therein), while our model achieves biases $|\Delta| \leq 0.001$ for $\beta \in [1.5, 2.5]$ on the 8×8 training geometry, growing to $|\Delta| \approx 0.05$ at $\beta = 3$. Both also demonstrate volume generalization through fully convolutional architectures.

The key quantitative differences can be attributed to the Metropolis correction. The MAALA acceptance rates reported in Ref. [18] (Fig. 4 therein) remain above 50% for $\beta \lesssim 12$ at $L = 16$, ensuring that even where the score network becomes inaccurate the generated samples are corrected towards the true distribution. Our uncorrected approach has no such safety net: deviations from the exact plaquette grow with both β and lattice volume for nontraining geometries (Table I), reaching $|\Delta| = 0.17$ at 32×32 . For the SU(2) plaquette at $\beta = 2$ on $L = 16$, Ref. [18] reports $\langle P \rangle = 0.4333(2)$ (Table II therein) against the exact value 0.4331, while our model on the same-volume 16×16 lattice yields $\langle P \rangle = 0.4339 \pm 0.0008(\text{stat})$, with bias $\Delta = +0.0008$, in excellent agreement with the exact value at this coupling, though the comparison is limited since the MAALA correction in Ref. [18] guarantees accuracy across all β values whereas our uncorrected approach degrades at higher couplings. These results highlight complementary strengths. The equivariant approach of Ref. [18] achieves superior accuracy and broader generalization range, particularly at large β and large volumes, owing to the built-in gauge symmetry and the MAALA correction. Our simpler flat-space approach, while more limited in range, demonstrates that even without explicit gauge equivariance or Metropolis adjustment a standard diffusion model can learn the correct physics at moderate couplings and lattice sizes close to the training geometry.

However, the two approaches also differ in a way that becomes significant when considering the long-term challenges of lattice field theory. The MAALA correction

requires evaluating the ratio $\exp(-\Delta S_E)$ of real Boltzmann weights at each Metropolis step. While this guarantees exactness for theories with a real action, the Metropolis correction step itself is not directly applicable to theories with a complex action, such as QCD at finite baryon density, real-time formulations, or theories with a θ -vacuum term, where the Boltzmann weight oscillates. This limitation applies to any Metropolis-based correction and is not specific to the equivariant framework of Ref. [18], which can equally be run without MAALA. More generally, any uncorrected diffusion approach, whether equivariant or not, never evaluates the action during sampling and relies entirely on the learned score function, making it a potential starting point for theories with a complex action, provided suitable training data can be obtained by alternative methods such as reweighting or complex Langevin dynamics.

Taken together, the equivariant framework of Ref. [18] and the simpler flat-space approach presented here represent complementary starting points for developing diffusion-based samplers for lattice gauge theories. The former establishes that high-accuracy generation with broad generalization is achievable when gauge symmetry and Metropolis correction are built in; the latter shows that even minimal architectures can capture the relevant physics in a regime close to training. Both approaches contribute to a broader effort aimed at building generative models that can ultimately address problems beyond the reach of traditional methods.

VI. OUTLOOK

Several directions for extending this work present themselves. On the architectural side, exploring gauge equivariant networks such as L-CNNs in place of the flat-space U-Net could improve accuracy by encoding symmetries directly rather than requiring the model to learn them from data. For near-term benchmarking in theories with a real action, incorporating a Metropolis correction step analogous to MAALA [18] would provide a practical way to correct residual model bias; however, developing action-free sampling strategies that maintain accuracy without such corrections remains the more important goal, as these are essential for eventual application to theories with a sign problem. On the methodological side, training an ensemble of independently initialized models and combining results using random-effects meta-analysis [28], as employed in Ref. [18], would provide a principled decomposition of the uncertainty into within-model statistical error and between-model variance, directly probing sensitivity to training stochasticity.

On the physical side, the extension to four-dimensional SU(2) Yang–Mills theory is currently underway. The key challenges are the increased dimensionality of the link fields (four directions instead of two) and the substantially larger configuration space, which will test both the scalability of

TABLE I. Average plaquette at $\beta = 2.0$ for increasing lattice sizes. σ_{stat} is the autocorrelation-corrected statistical error; $\Delta = \langle P \rangle_{\text{diff}} - \langle P \rangle_{\text{exact}}$ is the bias. The exact result $I_2(\beta)/I_1(\beta) = 0.4331$ is volume independent (see text).

$L_x \times L_t$	$\langle P \rangle_{\text{diff}}$	σ_{stat}	Δ	$ \Delta /\sigma_{\text{stat}}$
8×8	0.4329	0.0001	−0.0002	2.7
8×12	0.4331	0.0003	−0.0001	0.2
12×8	0.4333	0.0002	+0.0001	0.6
12×12	0.3892	0.0041	−0.0439	10.8
16×12	0.3594	0.0037	−0.0738	20.1
16×16	0.4339	0.0008	+0.0008	0.9
32×32	0.2661	0.0015	−0.1670	108

the diffusion framework and the effectiveness of physics-conditioned generation in higher dimensions.

For long-term applications involving four-dimensional SU(3) gauge theory, additional challenges arise. A central one concerns training data efficiency: our model is trained on 2×10^4 configurations, while Ref. [18] uses 10^5 configurations. Both training sets are relatively large for a two-dimensional theory where HMC is inexpensive, but scaling to four dimensions will require training ensembles on lattice volumes where each configuration could involve $\mathcal{O}(10^6)$ or more link variables. Producing comparably large training sets may itself become a computational bottleneck, particularly for expensive actions. Developing architectures and training strategies that can learn effectively from limited training data, for instance, by training on smaller lattice extents and generalizing to larger

volumes, is therefore an important challenge that remains to be investigated.

ACKNOWLEDGMENTS

We thank J. Bulava, P. Klein, J. Gegelia, F. He, L. Meng, F. Romero-López, and W. Sun for valuable discussions and helpful comments on the paper. The calculations of this work were performed on the HPC cluster Elysium of the Ruhr University Bochum, subsidized by the DFG (Grant No. INST 213/1055-1).

DATA AVAILABILITY

The code and data used for this paper are available in Ref. [29].

-
- [1] K. G. Wilson, *Phys. Rev. D* **10**, 2445 (1974).
 - [2] M. Creutz, *Phys. Rev. D* **21**, 2308 (1980).
 - [3] S. Duane, A. D. Kennedy, B. J. Pendleton, and D. Roweth, *Phys. Lett. B* **195**, 216 (1987).
 - [4] L. Del Debbio, H. Panagopoulos, and E. Vicari, *J. High Energy Phys.* **08** (2002) 044.
 - [5] P. de Forcrand, Proc. Sci. LAT2009 (2009) 010 [arXiv:1005.0539].
 - [6] M. S. Albergo, G. Kanwar, and P. E. Shanahan, *Phys. Rev. D* **100**, 034515 (2019).
 - [7] G. Kanwar, M. S. Albergo, D. Boyda, K. Cranmer, D. C. Hackett, S. Racanière, D. J. Rezende, and P. E. Shanahan, *Phys. Rev. Lett.* **125**, 121601 (2020).
 - [8] D. Boyda, G. Kanwar, S. Racanière, D. Jimenez Rezende, M. S. Albergo, K. Cranmer, D. C. Hackett, and P. E. Shanahan, *Phys. Rev. D* **103**, 074504 (2021).
 - [9] R. Abbott *et al.*, *Eur. Phys. J. A* **59**, 257 (2023).
 - [10] J. Ho, A. Jain, and P. Abbeel, Adv. Neural Inf. Process. Syst. **33**, 6840 (2020).
 - [11] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, arXiv:2011.13456.
 - [12] L. Wang, G. Aarts, and K. Zhou, *J. High Energy Phys.* **05** (2024) 060.
 - [13] L. Wang, G. Aarts, and K. Zhou, arXiv:2311.03578.
 - [14] Q. Zhu, G. Aarts, W. Wang, K. Zhou, and L. Wang, arXiv:2410.19602.
 - [15] Q. Zhu, G. Aarts, W. Wang, K. Zhou, and L. Wang, *J. High Energy Phys.* **03** (2026) 111.
 - [16] O. Vega, J. Komijani, A. El-Khadra, and M. Marinkovic, arXiv:2510.26081.
 - [17] G. Kanwar and O. Vega, arXiv:2512.19877.
 - [18] G. Aarts, D. E. Habibi, A. Ipp, D. I. Müller, T. R. Ranner, L. Wang, W. Wang, and Q. Zhu, arXiv:2601.19552.
 - [19] J.-M. Drouffe and J.-B. Zuber, *Phys. Rep.* **102**, 1 (1983).
 - [20] A. Q. Nichol and P. Dhariwal, arXiv:2102.09672.
 - [21] G. Parisi and Y. Wu, *Sci. Sin.* **24**, 483 (1981).
 - [22] D. Zwanziger, *Nucl. Phys.* **B192**, 259 (1981).
 - [23] O. Ronneberger, P. Fischer, and T. Brox, U-Net: Convolutional networks for biomedical image segmentation, in *Medical Image Computing and Computer-Assisted Intervention MICCAI 2015* (Springer, Cham, 2015), p. 234.
 - [24] A. Vaswani *et al.*, Adv. Neural Inf. Process. Syst. **30**, 5998 (2017).
 - [25] Y. Wu and K. He, Group normalization, in *ECCV* (Springer Cham, 2018), p. 3.
 - [26] D. P. Kingma and J. Ba, arXiv:1412.6980.
 - [27] M. Favoni, A. Ipp, D. I. Müller, and D. Schuh, *Phys. Rev. Lett.* **128**, 032003 (2022).
 - [28] R. DerSimonian and N. Laird, *Control. Clin. Trials* **7**, 177 (1986).
 - [29] B.-D. Sun, <https://github.com/sunbaodong/DiffusionModeISU2> (2026).