MULTILEVEL GENERATIVE SAMPLERS FOR INVESTIGATING CRITICAL PHENOMENA

Anonymous authors
Paper under double-blind review

ABSTRACT

Investigating critical phenomena, i.e., phase transitions, is of high interest in physics and chemistry. However, Monte Carlo (MC) simulations, a crucial tool for numerically analyzing macroscopic properties of given systems, are often hindered by the emerging *scale invariance at criticality* (SIC)—a divergence of the correlation length, which causes the system to behave the same at any length scale, as can be shown with renormalization group techniques. Many existing sampling methods suffer from SIC: long-range correlations cause critical slowing down in Markov chain Monte Carlo (MCMC), and require intractably large receptive fields for generative samplers. In this paper, we propose a Renormalization-informed Generative Critical Sampler (RiGCS)—a novel sampler specialized for near-critical systems, where SIC is leveraged as an advantage rather than a nuisance. Specifically, RiGCS builds on MultiLevel Monte Carlo (MLMC) with Heat Bath (HB) algorithms, which perform ancestral sampling from low-resolution to high-resolution lattice configurations with site-wise-independent conditional HB sampling. Although MLMC-HB is highly efficient under exact SIC, it suffers from a low acceptance rate under slight SIC violation. Notably, SIC violation always occurs in finite systems, and may induce long-range and higher-order interactions in the renormalized distributions, which are not considered by independent HB samplers. RiGCS enhances MLMC-HB by replacing a part of the conditional HB samplers with generative models that capture those residual interactions and improve the sampling efficiency. Our experiments show that the effective sample size of RiGCS is a few orders of magnitude higher than state-of-the-art generative model baselines in sampling configurations for 128 × 128 two-dimensional Ising systems. SIC also allows us to adopt a specialized sequential training protocol with model transfer, which significantly accelerates training.

1 INTRODUCTION

Monte Carlo (MC) simulations, where samples from a Boltzmann distribution are used to estimate macroscopic properties, are ubiquitous in many fields of science, ranging from chemistry (Metropolis et al., 1953), statistical physics (Hastings, 1970; Creutz et al., 1983), and quantum field theory (Creutz et al., 1979; Wilson, 1980) to biology (Huelsenbeck et al., 2001) and financial analysis (Doucet et al., 2001). In MC simulations, the ability to effectively sample from unnormalized distributions in a high-dimensional space poses crucial challenges. Standard algorithms such as Monte Carlo Markov Chain (MCMC) methods (Metropolis & Ulam, 1949; Robert & Casella, 2004) are often plagued by, e.g., slow convergence (Cowles & Carlin, 1996), energy barriers and local minima (Cérou et al., 2012), and critical slowing down (Wolff, 1990; 2004; Schaefer et al., 2011). This paper specifically tackles the problem of critical slowing down around critical regimes. In the broader context of physical sciences, the term *criticality* refers to situations where a system undergoes a sharp behavioral change, often associated with phase transitions (Nishimori & Ortiz, 2011). At criticality, physical systems typically exhibit self-similarity with respect to the change of scale, i.e., the physics of the coarse-grained system is similar to that of the fine-grained one. This phenomenon, called *scale invariance at criticality* (SIC), requires us to deal with arbitrarily long-range correlations for which standard MCMC samplers based on local moves undergo critical slowing down, meaning that the auto-correlation time becomes arbitrarily large with system size. Although many highly specialized cluster algorithms with non-local moves have been developed in the context of spin systems (Wolff, 1989b;a), critical slowing down still represents one of the major shortcomings of MCMC methods.

For efficient sampling around criticality, *MultiLevel* (or Multiscale) Monte Carlo with heat bath (MLMC-HB) algorithms (Schmidt, 1983; Faas & Hilhorst, 1986; Jansen et al., 2020) were developed, based on *Renormalization Group Theory (RGT)* (Kadanoff, 1966; Wilson, 1971; Wilson & Kogut, 1974; Cardy, 1996). RGT systematically analyzes how macroscopic features emerge when the system is *coarse-grained* to larger length scales by marginalizing fine degrees of freedom, and provides crucial insights into critical phenomena in statistical mechanics (Wilson, 1971; Fisher, 1973) and condensed matter physics (Shankar, 1994; Cardy, 1996). Furthermore, RGT also established the foundation for lattice quantum field theory (Wilson, 1974; Kogut & Susskind, 1975). An important outcome of RGT is the emergence of SIC over the coarse-grained and fine-grained lattices. Adopting the block-spin transformations (Kadanoff, 1966) for lattice site grouping, Schmidt (1983) proposed MLMC-HB that performs ancestral sampling from the coarsest lattice sites to the finest ones. The key advantage is that the conditional distributions between consecutive resolution levels can be factorized into independent distributions under SIC, for which sampling can be efficiently performed by HB algorithms. In MLMC-HB, the long-range correlations are captured in the low resolution lattice, which is much easier than capturing them in the original high resolution lattice.

Machine learning techniques are also seen as potential candidates to, either partially or fully, overcome the shortcomings of MCMC algorithms. In particular, generative models with accessibility to the exact sampling probability—such as normalizing flows (Rezende & Mohamed, 2015; Kobyzev et al., 2020; Papamakarios et al., 2021) and autoregressive models (van den Oord et al., 2016c;b; Salimans et al., 2017)—offer efficient independent sampling and unbiased MC estimation via importance sampling, showing notable success across various domains. Such applications include statistical physics (Wu et al., 2019; Nicoli et al., 2020), quantum many-body systems (Hibat-Allah et al., 2020) quantum chemistry (Noé et al., 2019; Gebauer et al., 2019), string theory (Caselle et al., 2024), and lattice field theory (Albergo et al., 2019; Nicoli et al., 2021; Caselle et al., 2022; Cranmer et al., 2023; Abbott et al., 2024). However, since capturing long-range interactions in large lattice systems may require intractably large receptive fields, generative models tend to struggle to generate samples around criticality, except for a few recent works whose goal was to mitigate critical slowing down (Pawlowski & Urban, 2020; Białas et al., 2023).

In this work, we propose a Renormalization-informed Generative Critical Sampler (RiGCS), which enhances MLMC-HB algorithms by mitigating their major weakness—i.e., the HB samplers in MLMC-HB ignore long-range and higher-order interactions that may exist in renormalized systems when SIC is slightly violated. RiGCS approximates the renormalized distributions with generative models with large enough receptive fields that can capture the greater part of those residual interactions. In our experiments, RiGCS drastically improves the sampling efficiency of MLMC-HB, and achieves an effective sample size a few orders of magnitude better than the previous state-of-the-art generative sampler (Białas et al., 2023) for the two-dimensional Ising model. Furthermore, we propose a specialized sequential training procedure with *warm starts* by transferring model parameters between different resolution levels, which substantially improves the training efficiency. Our contributions include:

- **Renormalization-informed Multilevel Sampling**: We propose a novel method that leverages both SIC and generative modeling to efficiently draw samples from Boltzmann distributions around the criticality.
- **Sequential Training with Warm Start by Model Transfer**: We propose a sequential training procedure starting from solving a small scale system to a (target) large scale system, where the model parameters trained for smaller systems are transferred to larger ones for initialization. This particular warm start strategy accelerates training significantly.

Like most works developing generative neural samplers for simulating physical systems on the lattice, we do not claim that our method outperforms state-of-the-art MCMC samplers, such as cluster methods for the Ising model, which remain unmatched by any generative neural sampler for general observable estimation in large scale systems.

Related Work Renormalization Group Theory (RGT) (Wilson, 1971; Wilson & Kogut, 1974; Kadanoff, 1966) has profoundly influenced the study of critical behavior in statistical systems and quantum field theory. Leveraging results of RGT, Schmidt (1983) proposed MLMC-HB for near-critical systems, which showed notable improvements in sampling efficiency for one- and two-dimensional Ising models. MLMC-HB adopts a particular partitioning of the lattice sites, called *block-spin transformations* (Kadanoff, 1966), and draws samples hierarchically by site-wise independent conditional HB sampling, based on the renormalized systems at different scales. Faas & Hilhorst (1986) further enhanced MLMC-HB by incorporating long-range interactions. Recently, Jansen et al. (2020) introduced a low variance MC estimator by leveraging the correlations between the lattices with consecutive resolution levels, which further advanced MLMC-HB.

A variety of generative models, including Generative Adversarial Networks (GANs) (Pawlowski & Urban, 2020; Singha et al., 2022), Variational Auto Encoders (VAEs) (D'Angelo & Böttcher, 2020), and energy-based models (D'Angelo & Böttcher, 2020; Torlai & Melko, 2016), have been used as independent MC samplers for lattice systems. Generative models with accessible sampling probability are particularly useful for MC simulations because they allow for unbiased MC estimation with importance sampling or Metropolis-Hastings rejection (Nicoli et al., 2020). Specifically, Variational Autoregressive Networks (VANs) (Wu et al., 2019; Nicoli et al., 2020) and normalizing flows (Albergo et al., 2019) for discrete and continuous systems, respectively, have proven to be highly effective. Recent works (Singha et al., 2023a;b; Gerdes et al., 2023) introduced conditional normalizing flows for scalar and gauge theories, and showed that models trained away from criticality can be well interpolated for drawing samples near criticality. Nicoli et al. (2021) demonstrated that generative models are particularly useful for estimating thermodynamic observables, e.g., free energy and entropy, which cannot be directly estimated with standard MCMC methods.

Recently, hierarchical sampling approaches have been integrated with generative modeling both for (discrete) statistical systems (Li & Wang, 2018; Białas et al., 2022) and (continuous) lattice field theories (Finkenrath, 2024; Abbott et al., 2024). Neural Network Renormalization Group (NeuralRG) (Li & Wang, 2018) uses a hierarchical bijective mapping to learn a renormalization transform, and was applied to the Ising model using a continuous relaxation technique. Hierarchical Autoregressive Network (HAN) (Białas et al., 2022)—a state-of-the-art generative sampler for discrete physical systems—uses a recursive domain decomposition (Cè et al., 2016), and performs independent conditional sampling for separate regions with trained VANs. The HAN approach has shown improved sampling efficiency compared to MLMC-HB in two-dimensional Ising models. We refer to Appendix A for an extended review of related works.

2 BACKGROUND

This paper focuses on MC simulations of hypercubic lattice systems around criticality. We refer to a (row vector) *sample* $\boldsymbol{s} \in \mathcal{S}^V$ to be a *configuration* on the $V = N^D$ grid points in the D dimensional lattice, where $\mathcal{S}$ denotes the domain of the random variable at each site, and N denotes the lattice size (per dimension). Given a Hamiltonian (or energy) $H(\boldsymbol{s})$ describing the interactions between the lattice sites, MC simulations draw samples from the Boltzmann distribution

$$p(\boldsymbol{s}) = \tfrac{1}{Z} e^{-\beta H(\boldsymbol{s})}, \tag{1}$$

where β is the inverse temperature and Z is the (typically unknown) partition function. With a sufficient number M of samples, physical observables $\mathcal{O}$, e.g., energy, magnetization, and susceptibility, can be estimated by averaging over the sample configurations $\langle \mathcal{O} \rangle \approx \frac{1}{M} \sum_{m=1}^{M} \mathcal{O}(\boldsymbol{s}_m)$, thus revealing macroscopic physical properties and phenomena, like phase transitions. Below, we introduce common sampling methods with and without machine learning techniques.

2.1 MARKOV CHAIN MONTE CARLO (MCMC) METHODS AROUND CRITICALITY

Markov Chain Monte Carlo (MCMC) methods are a class of algorithms used to sample from unnormalized distributions. Since the partition function is not analytically computable in most physical systems, MCMC methods are fundamental tools for performing MC simulations, e.g., in statistical mechanics and lattice quantum field theory. A crucial challenge for MCMC sampling around criticality is to cope with long-range correlations. When distant regions in the lattice become strongly correlated, the general MCMC methods, relying on local updates, struggle to move from a low-energy state to another low-energy state. This is because local updates generally ignore the correlations, and thus, the proposed trials tend to be rejected in the Metropolis-Hastings rejection step

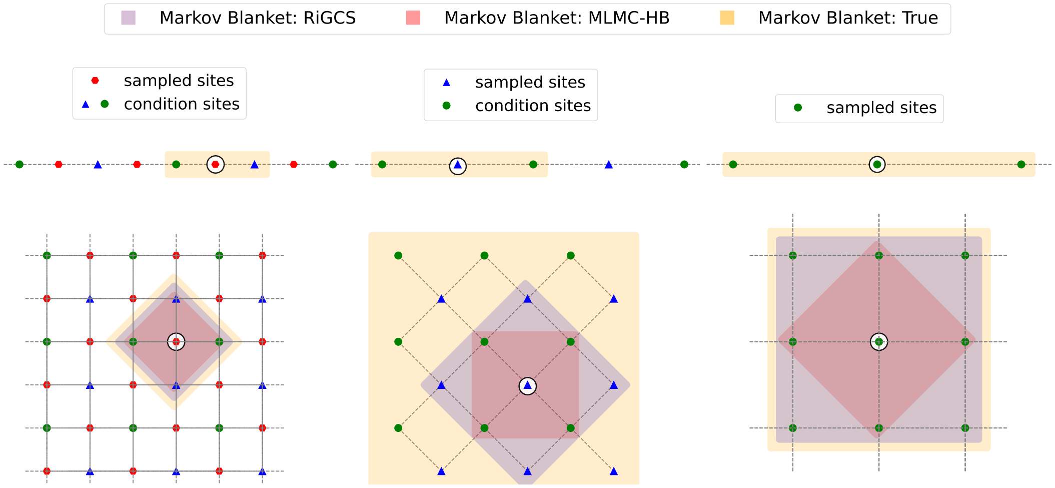

Figure 1: Site partitionings based on the block-spin transformation (Kadanoff, 1966). The ancestral sampling is performed from the coarsest level (right) to the finest level (left), namely, in the order of $\boldsymbol{s}^{L-2}$ (green), $\boldsymbol{s}^{L-1}$ (blue), and $\boldsymbol{s}^{L}$ (red). In one-dimensional case (upper-low), the marginal distribution at each resolution level has only NN interactions, and therefore, the true Markov blanket (yellow shadows) of the highlighted site (by a black circle) contains only NN condition sites, which allows accurate independent HB sampling. In two-dimensional case (lower row), the marginal distributions also have long-range and higher-order interactions. Therefore, except at the finest level, the true Markov blanket (yellow shadows) contain all other sites. MLMC-HB with NN Markov blankets (red shadows) can still be used as an approximate trial sampler, but suffers from low acceptance rates for large lattice sizes. Our RiGCS with wider Markov blankets (purple shadows), induced by the receptive field of generative models, can capture the long-range and higher-order interactions, and thus generate more accurate samples. Note that the Markov blankets of RiGCS we used for the intermediate resolutions are of the size 11×11 (i.e., much larger than the one shown in this figure). $\boldsymbol{s}_{L-2}$ (green) can be further partitioned until the dimension of $\boldsymbol{s}^{0}$ gets sufficiently small.

due to the increased energy. This results in a long autocorrelation time for the Markov chain—a phenomenon known as *critical slowing down* (Wolff, 1990). The two approaches described below were developed to mitigate critical slowing down.

Cluster algorithms Cluster algorithms (Wolff, 1989a; Swendsen & Wang, 1987) perform global updates by identifying clusters of correlated lattice sites and flipping them collectively. These global updates efficiently reduce the autocorrelation time and mitigate critical slowing down. Appendix B introduces two variants of cluster methods, including the Wolff algorithm (Wolff, 1989a) which we refer to as the *cluster algorithm* throughout the rest of the paper. Cluster methods are not seen as universal remedies against critical slowing down because they are not generally applicable to arbitrary continuous variable systems, although generalizations to specific continuous systems was proposed (Kent-Dobias & Sethna, 2018) with less efficiency in reducing autocorrelation times.

Multilevel Monte Carlo with Heat Bath (MLMC-HB) We denote by $\boldsymbol{J}^{\mathrm{NN}} \in \mathbb{R}^{V \times V}$ a homogeneous $2 \cdot D$ nearest neighbor (NN) interaction matrix that satisfies

$$(\boldsymbol{J}^{\mathrm{NN}})_{v,v'} = \begin{cases} J & \text{if } (v, v') \text{ are } 2 \cdot D \text{ nearest neighbor pairs,} \\ 0 & \text{otherwise,} \end{cases} \tag{2}$$

for $J \in \mathbb{R}$. In some important physical systems, including the Ising model (Onsager, 1944) and the XY model (Kosterlitz, 1974), the Hamiltonian can be written as

$$H(\boldsymbol{s}) = -\boldsymbol{s}\boldsymbol{J}^{\mathrm{NN}}\boldsymbol{s}^{\top} \tag{3}$$

with $\boldsymbol{J}^{\mathrm{NN}} \in \mathbb{R}^{V \times V}$.[1] The MultiLevel Monte Carlo with Heat Bath (MLMC-HB) algorithm (Schmidt, 1983)—a protocol inspired by RGT—was developed for sampling in such physical systems around the criticality.

In MLMC-HB, one partitions the lattice sites into $L+1$ levels $\boldsymbol{s} = (\boldsymbol{s}^{L}, \boldsymbol{s}^{L-1}, \ldots, \boldsymbol{s}^{0})$ so that all $2 \cdot D$ nearest neighbors of each entry of $\boldsymbol{s}^{l}$ (for $l = 1, \ldots, L$) belong to the coarser level groups $(\boldsymbol{s}^{l-1}, \ldots, \boldsymbol{s}^{0})$. Figure 1 shows examples of site partitionings based on the block-spin transformations (Kadanoff, 1966) for $(D=1)$- and $(D=2)$-dimensional lattices, where the sites with the same color (green, blue, or red) belong to the same groups $(\boldsymbol{s}^{L-2}, \boldsymbol{s}^{L-1}, \boldsymbol{s}^{L})$. For compact descriptions, we use inequalities to express subsets of the groups, e.g., $\boldsymbol{s}^{\leq l} = (\boldsymbol{s}^{l}, \ldots, \boldsymbol{s}^{0})$, and $\boldsymbol{s}^{>l} = (\boldsymbol{s}^{L}, \ldots, \boldsymbol{s}^{l+1})$. We denote by V_l the dimension of $\boldsymbol{s}^{\leq l}$, i.e., $\boldsymbol{s}^{\leq l} \in \mathbb{R}^{V_l}$ and $\boldsymbol{s}^{>l} \in \mathbb{R}^{V - V_l}$. The marginal distribution of the l-th level lattice is given as

$$p(\boldsymbol{s}^{\leq l}) = \textstyle\int p(\boldsymbol{s}) \mathcal{D}[\boldsymbol{s}^{>l}] \equiv \frac{1}{Z_l} e^{-\beta H_l(\boldsymbol{s}^{\leq l})}, \tag{4}$$

where the corresponding Hamiltonian $H_l(\boldsymbol{s}^{\leq l})$ (i.e., a scaled negative log marginal likelihood) is called a renormalized Hamiltonian. An important result in RGT is that, around the criticality, the renormalized Hamiltonian for $l = 0, \ldots, \widetilde{L}$, where $\widetilde{L}$ is a few levels smaller than L, can be approximated as a Hamiltonian with NN interactions, i.e.,

$$H_l(\boldsymbol{s}^{\leq l}) \approx \widetilde{H}_l(\boldsymbol{s}^{\leq l}), \qquad \text{where} \qquad \widetilde{H}_l(\boldsymbol{s}^{\leq l}) = -\boldsymbol{s}^{\leq l} \boldsymbol{J}_l^{\mathrm{NN}} (\boldsymbol{s}^{\leq l})^{\top} \tag{5}$$

with the NN interaction matrix $\boldsymbol{J}_l^{\mathrm{NN}} \in \mathbb{R}^{V_l \times V_l}$.[2] If Eq. (5) holds exactly, the conditional probability $p(\boldsymbol{s}^{l} | \boldsymbol{s}^{\leq l-1})$ can be decomposed as

$$p(\boldsymbol{s}^{l} | \boldsymbol{s}^{\leq l-1}) = \textstyle\prod_{v=1}^{V_l} p(s_v^{l} | \boldsymbol{s}^{\leq l-1}), \tag{6}$$

because the Markov blanket[3] (Bishop, 2006) of s_v^{l} does not contain $s_{v'}^{l}$ for any $v' \neq v$ (see the yellow shadows in Figure 1 top). This makes the sampling from the conditional distribution (6) extremely easy and efficient—one can apply the HB conditional sampling *exactly* for the discrete domain (with a probability table of size $|\mathcal{S}|$), and *approximately* for the continuous domain (with, e.g., a one-dimensional Gaussian mixture). Therefore, starting from samples drawn from $p(\boldsymbol{s}_0)$ (which can be efficiently performed by HB or MCMC if L is sufficiently large and hence V_0 is small), the ancestral sampling of the full lattice according to

$$p(\boldsymbol{s}) = \left(\textstyle\prod_{l=1}^{L} p(\boldsymbol{s}^{l} | \boldsymbol{s}^{\leq l-1}) \right) p(\boldsymbol{s}^{0}) \tag{7}$$

can be efficiently performed. Intuitively, MLMC-HB captures the long-range correlations by the coarse level marginals, i.e., $p(\boldsymbol{s}^{\leq l})$ for small l, which avoids two major difficulties—large lattice size and long-range correlations—arising at the same time.

For the $(D=1)$-dimensional lattice (see Figure 1 upper row), it is known that Eq. (5), and thus Eq. (6), hold exactly, and therefore, MLMC-HB generates accurate samples from the target Boltzmann distribution. For $D \geq 2$ (see Figure 1 lower row), Eq. (5) holds only approximately, and therefore, MLMC-HB should be combined with importance sampling or Metropolis-Hastings rejection. Namely, we draw samples according to

$$q^{\mathrm{NN}}(\boldsymbol{s}) = p(\boldsymbol{s}^{L} | \boldsymbol{s}^{\leq L-1}) \left(\textstyle\prod_{l=1}^{L-1} q^{\mathrm{NN}}(\boldsymbol{s}^{l} | \boldsymbol{s}^{\leq l-1}) \right) q^{\mathrm{NN}}(\boldsymbol{s}^{0}), \tag{8}$$

and compensate the sampling bias by using the sampling probability $q^{\mathrm{NN}}(\boldsymbol{s})$, where $\{q^{\mathrm{NN}}(\boldsymbol{s}^{l} | \boldsymbol{s}^{\leq l-1})\}_{l=1}^{L-1}$ and $q^{\mathrm{NN}}(\boldsymbol{s}_0)$ are approximate distributions with NN interactions to the true

---

[1] The whole discussion in this paper can be applied to a slight generalization with the Hamiltonian in the form of $H(\boldsymbol{s}) = -\sum_{v,v'} (\boldsymbol{J}^{\mathrm{NN}})_{v,v'} \psi(s_v, s_{v'})$, where $\psi(s, s')$ is a similarity function between two states, e.g., $\psi(s, s') = \delta_{s,s'}$ for Potts model (Wu, 1982).

[2] The corresponding NN interaction coefficient—J in Eq. (2) which we refer to as J_l—can be analytically computed in RGT (see Appendix C). When $N, L \to \infty$, J_l converges to a limiting value as l decreases, and thus the *scale invariance at criticality* (SIC) emerges. Since we consider finite lattices, exact SIC never holds. Nevertheless, in one-dimensional finite lattices, the renormalized Hamiltonians consist only of NN interactions, which is sufficient for MLMC-HB to be accurate, as explained in Figure 1.

[3] The Markov blanket of a random variable s_v is a set of other random variables $\mathcal{B}_{s_v} \subseteq \{s_{v'}\}_{v' \neq v}$ that have sufficient information to determine the conditional distribution of s_v given the other random variables, i.e., $p(s_v | \mathcal{B}_{s_v}) = p(s_v | \{s_{v'}\}_{v' \neq v})$.

conditionals $\{p(\boldsymbol{s}^l|\boldsymbol{s}^{\leq l-1})\}_{l=1}^{L-1}$ and the true marginal $p(\boldsymbol{s}_0)$, respectively. Unfortunately, the approximation errors accumulate through ancestral sampling, leading to a significantly low acceptance rate for large lattice sizes. We show in Section 4 that MLMC-HB is not very efficient for $D = 2$. Further details on RGT and MLMC-HB are given in Appendix C and Appendix D, respectively.

2.2 GENERATIVE MODELING FOR MC SIMULATIONS

In recent years, deep generative models have gained significant traction in the field of physics for their efficient modeling of complicated probability distributions. In particular, normalizing flows (Kobyzev et al., 2020) and autoregressive neural networks (van den Oord et al., 2016c) became very popular in the context of computational physics due to their intrinsic capability of providing the exact sampling probability $q_{\boldsymbol{\theta}}(\boldsymbol{s})$, which allows *asymptotically* unbiased MC estimation. Notably, well-trained generative models can provide *independent* samples from an approximate distribution, and asymptotically unbiased estimates of physical observables can be computed by importance sampling (Nicoli et al., 2020): $\langle \mathcal{O} \rangle \approx \frac{1}{M}\sum_{m=1}^{M} \frac{\widetilde{w}_m}{\sum_{m'=1}^{M} \widetilde{w}_{m'}} \mathcal{O}(\boldsymbol{s}_m)$, where $\widetilde{w} = e^{-\beta H(\boldsymbol{s})}/q_{\boldsymbol{\theta}}(\boldsymbol{s})$ is an unnormalized importance weight. However, naive generative modeling can be problematic for sampling large lattices near criticality because large receptive fields are required to capture long-range correlations. Improving the scalability of generative samplers is therefore one of the most crucial challenges for reaching the same level of efficiency as state-of-the-art MCMC samplers for large systems.

3 METHOD

In this section, we describe our proposed method that enhances MLMC-HB (see Section 2.1) with generative modeling (Section 2.2). We focus on $(D = 2)$-dimensional lattice systems with $(V = N \times N)$ grid points.

3.1 RENORMALIZATION-INFORMED GENERATIVE CRITICAL SAMPLER (RIGCS)

Higher-order RGT (Maris & Kadanoff, 1978) shows that, for $D = 2$, the Hamiltonian of the marginal distribution (4) consists not only of the NN interaction terms but also of long-range and higher-order interaction terms:

$$H_l(\boldsymbol{s}^{\leq l}) = -\boldsymbol{s}^{\leq l}\boldsymbol{J}_l^{\mathrm{NN}}(\boldsymbol{s}^{\leq l})^\top - \boldsymbol{s}^{\leq l}\boldsymbol{J}_l^{\mathrm{LR}}(\boldsymbol{s}^{\leq l})^\top - \sum_{v,v',v''} (\mathcal{J}_l^{\mathrm{HO}})_{v,v',v''} s_v^{\leq l} s_{v'}^{\leq l} s_{v''}^{\leq l} + \cdots, \tag{9}$$

where $\boldsymbol{J}_l^{\mathrm{LR}}$ and $\mathcal{J}_l^{\mathrm{HO}}$ denote the matrix and the tensor that express the long-range and high-order interactions, respectively.[4] Instead of approximating the renormalized Hamiltonian (9) with the Hamiltonian (5) with NN interactions, our method, called Renormalization-informed Generative Critical Sampler (RiGCS), approximates the long-range and higher-order interactions with generative models. Specifically, RiGCS performs ancestral sampling according to

$$q_{\boldsymbol{\theta}}(\boldsymbol{s}) = p(\boldsymbol{s}^L|\boldsymbol{s}^{\leq L-1}) \left(\prod_{l=1}^{L-1} q_{\boldsymbol{\theta}_l}(\boldsymbol{s}^l|\boldsymbol{s}^{\leq l-1})\right) q_{\boldsymbol{\theta}_0}(\boldsymbol{s}^0), \tag{10}$$

where $q_{\boldsymbol{\theta}_l}(\boldsymbol{s}^l|\boldsymbol{s}^{\leq l-1})$ for $l = L-1, \ldots, 1$ and $q_{\boldsymbol{\theta}_0}(\boldsymbol{s}_0)$ are conditional and unconditional generative models that approximate $p(\boldsymbol{s}^l|\boldsymbol{s}^{\leq l-1})$ and $p(\boldsymbol{s}_0)$, respectively. Here, $\boldsymbol{\theta} = (\boldsymbol{\theta}_{L-1}, \ldots, \boldsymbol{\theta}_0)$ denotes all model's trainable weight parameters. Similarly to the configuration variable $\boldsymbol{s}$, we use inequalities to express subsets of the parameters, e.g., $\boldsymbol{\theta}_{\leq l} = (\boldsymbol{\theta}_l, \ldots, \boldsymbol{\theta}_0)$. Note that, at the finest level, the conditional distribution $p(\boldsymbol{s}^L|\boldsymbol{s}^{\leq L-1})$ has only NN interactions by assumption, and therefore, the exact HB algorithm can be efficiently applied without generative modeling. Therefore, $\boldsymbol{\theta} = \boldsymbol{\theta}_{\leq L-1}$—as there is no model parameter at the L-th level.

3.2 RECEPTIVE FIELD DESIGN

It is known that the long-range and higher-order interactions decay between lattice sites with longer distances (Maris & Kadanoff, 1978). Therefore, we can tune the accuracy of approximating the

---

[4] Note the difference between the "interactions" and the "correlations." The former mean the direct cross dependent terms in the Hamiltonian, while the latter means statistical dependence in the Boltzmann distribution.

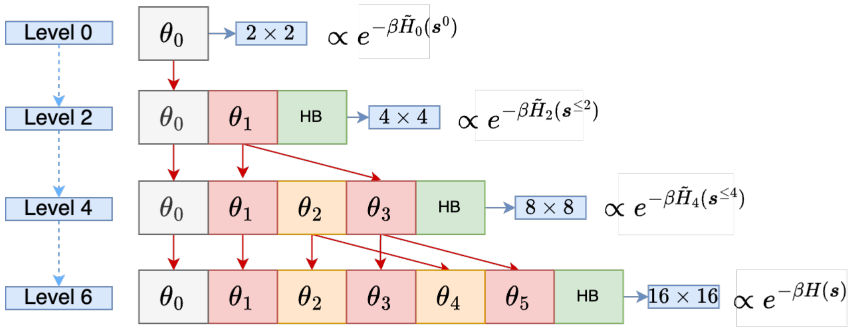

Figure 2: Illustration of sequential training of $(L = 6)$-layered RiCGS for the 16×16 lattice, where the marginal model $q_{\boldsymbol{\theta}_0}(\boldsymbol{s}^0)$ is first trained on the smallest 2×2 target Boltzmann distribution, and then larger RiGCS with conditional models are trained sequentially on the target Boltzmman distributions with larger lattices. The red arrows indicate the model transfer initializations. Note that the parameters without incoming red arrow are randomly initialized. For larger lattices, we continue the last step until the target size is reached.

marginal Hamiltonian by controlling the receptive fields of conditional generative models, i.e., the larger the receptive field is, the more accurate the approximation to the marginal Hamiltonian (9) is. Compared to vanilla generative models (without multilevel sampling), where the receptive field needs to cover the whole correlation range in the original finest-level lattice, our RiGCS approach allows us to keep the receptive field of each conditional model small. In particular, if we set the number L of levels proportional to the lattice size N (per dimension), we can keep the receptive field size constant for different N. This is because our RiGCS captures the long-range interactions in the coarser level models—the coarsest generative model $q_{\theta_0}(\boldsymbol{s}^0)$ with the receptive field size $\alpha \times \alpha$ effectively amounts to the receptive field size $\alpha L/2 \times \alpha L/2$ in the original finest lattice. In principle, an optimal receptive field size for each level l should exist such that the accumulated approximation error is minimized for a given computational cost. However, this work uses the same architecture for the conditional models for $l = L-1, \ldots, 1$, which enables efficient model transfer in training, as explained below. Besides, this choice makes the model complexity of RiGCS linear to L.

3.3 SEQUENTIAL TRAINING WITH MODEL TRANSFER INITIALIZATION

We train our RiGCS by minimizing the reverse Kullback-Leibler (KL) divergence, i.e.,

$$\min_{\boldsymbol{\theta}} \mathrm{KL}(q_{\boldsymbol{\theta}}(\boldsymbol{s}) \| p(\boldsymbol{s})) = \textstyle\sum_{\boldsymbol{s} \in \mathcal{S}^V} q_{\boldsymbol{\theta}}(\boldsymbol{s}) \log \frac{q_{\boldsymbol{\theta}}(\boldsymbol{s})}{p(\boldsymbol{s})} \approx \frac{1}{M_{\mathrm{tr}}} \sum_{m=1}^{M_{\mathrm{tr}}} \log \frac{q_{\boldsymbol{\theta}}(\boldsymbol{s}_m)}{p(\boldsymbol{s}_m)}, \tag{11}$$

which is estimated from the generated samples $\{\boldsymbol{s}_m \sim q_{\boldsymbol{\theta}}(\boldsymbol{s})\}_{m=1}^{M_{\mathrm{tr}}}$—training data drawn from the target distribution are not required. However, training all parameters $\boldsymbol{\theta}$ from scratch, e.g., with random initialization, tends to suffer from long initial random walking steps. This is because the randomly initialized RiGCS, $q_{\boldsymbol{\theta}}(\boldsymbol{s})$, generates random samples, for which the stochastic gradient of the objective (11), rarely provides useful signal to train the model. We tackle this problem with a specific training procedure with *model transfer*, again based on RGT.

We choose L to an even number, and consider a set of *sequential target* Boltzmann distributions $p_{L'}(\boldsymbol{s}^{\leq L'}) \propto e^{-\beta \widetilde{H}_{L'}(\boldsymbol{s}^{\leq L'})}$ for $L' = 0, 2, 4, \ldots, L$, where $\{\widetilde{H}_l(\boldsymbol{s}^{\leq l})\}_{l=0}^{L}$ are the approximate renormalized Hamiltonians with NN interactions, defined in Eq.(5), and $\widetilde{H}_L(\boldsymbol{s}^{\leq L}) = H(\boldsymbol{s})$. For each target $p_{L'}(\boldsymbol{s}^{\leq L'})$ in the increasing order of L', we train a RiGCS $q_{\boldsymbol{\theta}_{\leq L'-1}}(\boldsymbol{s}^{\leq L'})$ that shares the same coarsest lattice size V_0 as the final model $q_{\boldsymbol{\theta}_{\leq L-1}}(\boldsymbol{s}^{\leq L}) = q_{\boldsymbol{\theta}}(\boldsymbol{s})$. This allows us to initialize the RiGCS parameters for learning $p_{L'}(\boldsymbol{s}^{\leq L'})$ to the corresponding parameters already trained on $p_{L'-2}(\boldsymbol{s}^{\leq L'-2})$—an easier (smaller lattice size) system. Figure 2 illustrates this procedure, where the initializations are indicated by the red arrows. Thanks to SIC, the models connected by the red arrows are similar to each other, as detailed in Appendix E.1. Note that all parameters except $\boldsymbol{\theta}_0, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2$—which are trained on the three smallest lattice sizes with random initializations—can be warm started. Pseudocode of sampling and training routines of RiGCS is given in Appendix E.2.

4 NUMERICAL EXPERIMENTS

We evaluate our proposed RiGCS, and compare it with several baselines. Below we describe the experimental setting.

Target Physical Systems We adopt the two-dimensional Ising model, for which the Hamiltonian is given by Eq. (3) with the 2-dimensional binary lattice, i.e., $\boldsymbol{s} \in \mathcal{S}^V = \{-1, 1\}^{N \times N}$, and $J = 1$. This commonly used benchmark system is suitable for our evaluation, because it exhibits a second-order phase transition, and is exactly solvable (Onsager, 1944)—the ground-truth is analytically computed. We set the inverse temperature to $\beta = 0.44$, which corresponds to the critical (inverse) temperature where the phase transition occurs by spontaneous symmetry breaking in the limit of an infinite lattice. The lattice sizes are set to $N = 8, 16, 32, 64, 128$.

Generative models Since we focus on the binary domain, we use autoregressive neural networks (ARNNs)

$$q_\theta(\boldsymbol{s}) = \left(\prod_{v=2}^{V} q_\theta(s_v \mid s_{v-1}, \dots, s_1)\right) q_\theta(s_1) \tag{12}$$

for unconditional and conditional generative models at each level of RiGCS. More specifically, we adopt PixelCNNs (van den Oord et al., 2016c;b), which allow us to control their receptive fields by setting the convolution kernel sizes. We provide details on ARNNs in Appendix F.

Baselines We compare our method against three multilevel/generative model baselines: MLMC-HB, VAN (plain ARNN without multilevel sampling) (Wu et al., 2019), and HAN (Białas et al., 2022). We also evaluate the Wolff cluster method (see App. B), which is widely recognized as a very efficient sampler for the Ising model, and no generative model has yet exceeded its performance. In our experiments, RiGCS achieved comparable (although still worse) performance to the cluster method on large lattices, outperforming the previous state-of-the-art (Białas et al., 2022) as well as other baselines, such as VAN (Wu et al., 2019) and naive MLMC-HB (Schmidt, 1983; Faas & Hilhorst, 1986; Jansen et al., 2020).

Model Architecture: We set the number of levels to $L = N/2 - 2$, such that the coarsest lattice size is always $V_0 = 2 \times 2$. Our RiGCS (10) consists of a PixelCNN[5] with a three masked convolutional layers (12 channels) with the half kernel size of 6 as the coarsest level (unconditional) generative model $q_{\boldsymbol{\theta}_0}(\boldsymbol{s}^0)$, and CNNs with two convolutional layers (12 channels) with the half kernel sizes of 5 and 3 as the intermediate level conditional generative models $\{q_{\boldsymbol{\theta}_l}(\boldsymbol{s}^l|\boldsymbol{s}^{\leq l-1})\}_{l=1}^{L-1}$. The receptive field of $q_{\boldsymbol{\theta}_0}(\boldsymbol{s}^0)$ covers the whole coarsest level lattice, while the receptive field of $q_{\boldsymbol{\theta}_l}(\boldsymbol{s}^l|\boldsymbol{s}^{\leq l-1})$ for $l = 1, \dots, L-1$ is 11×11—it captures the long-range and higher-order interactions up to 5-steps distant sites. Note that RiGCS performs the exact independent HB sampling with $p(\boldsymbol{s}^L|\boldsymbol{s}^{\leq L-1})$ at the finest level, and is trained sequentially from a small lattice size to the larger lattices with $N = 4, 8, ..., 128$, as described in Section 3.3. We again refer readers to Appendix G for more details.

4.1 FREE ENERGY ESTIMATION

We first evaluate the sampling methods in terms of the bias and the variance in estimating the free energy—a thermodynamic observable. We combine the cluster method with annealed importance sampling (AIS) (Neal, 2001) to estimate the free energy (Caselle et al., 2016). For generative samplers, i.e., VAN, HAN, and our RiGCS, we use the asymptotically unbiased estimator (Nicoli et al., 2020) for the free energy:

$$\hat{F} = -\frac{1}{\beta}\log \hat{Z}, \quad \text{where} \quad \hat{Z} = \frac{1}{M}\sum_{m=1}^{M} e^{-\beta H(\boldsymbol{s}_m)}/q_{\boldsymbol{\theta}}(\boldsymbol{s}_m), \quad \boldsymbol{s}_m \sim q_\theta(\boldsymbol{s}), \tag{13}$$

where M is the number of generated samples. We can use this estimator also for MLMC-HB because we can compute all factors in Eq. (8) including the normalization constants. This is because all conditionals $\{q^{\mathrm{NN}}(\boldsymbol{s}^l|\boldsymbol{s}^{\leq l-1})\}_{l=1}^{L}$ are products of independent distributions, and the lowest level marginal $q^{\mathrm{NN}}(\boldsymbol{s}^0)$ is the Boltzmann distribution with only $2^{2\times 2}$ states.

---

[5] We used the PixelCNN implementation by Wu et al. (2019).

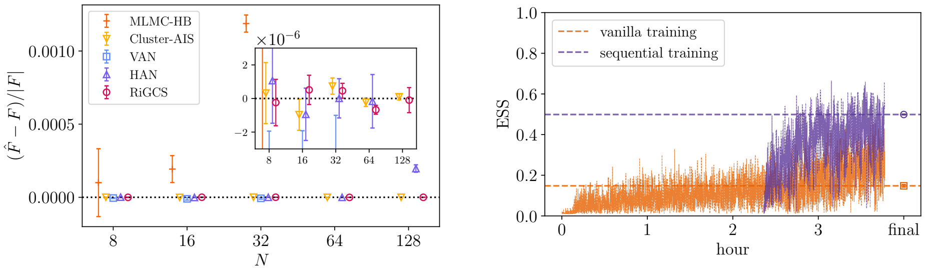

Figure 3: Left: Relative estimation error for the free energy. Only RiGCS provides estimates for $N = 128$ that are comparable to those of Cluster-AIS. Note that the vanilla VAN cannot be trained for $N \geq 64$ in reasonable time. Right: Comparison between vanilla training and the proposed sequential training for RiGCS for $N = 64$ in terms of the achieved ESS. The plot for the sequential training starts at the time ≈ 2.3 hours when the pre-training for smaller lattice systems is finished. ESS at each training epoch is computed with $M = 16$ samples, and the markers at "final" show the ESS computed with $M = 10^6$ samples after training finished.

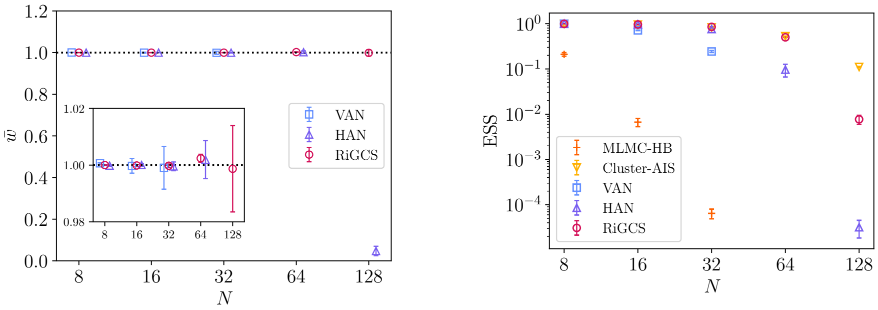

Figure 4: EMDM (left) and ESS (right). The vanilla VAN cannot be trained for $N \geq 64$ in reasonable time. The inset in the left hand side plot zooms around the region of $\bar{w} = 1.0$, showing that RiGCS does not suffer from mode dropping even for large $N = 128$. Similarly, the ESS displayed in the plot on the right indicates that RiGCS achieves the closest performance to Cluster-AIS among the generative neural samplers for large systems.

Figure 3 left shows the relative estimation error $(\hat{F} - F)/|F|$ by our RiGCS and the baselines, where F is the analytically computed free energy (Onsager, 1944), and each error bar show one standard deviation of the statistical error. We observe the following. MLMC-HB performs poorly for $N \geq 32$, exhibiting strong biases (results are out of the range for $N \geq 64$). The other four methods provide compatible (unbiased) estimation up to $N \leq 32$, but Cluster-AIS and our RiGCS outperform VAN and HAN in terms of the variance. For $N = 128$, only Cluster-AIS and our RiGCS can perform compatible estimation to the true free energy value with reasonable computation time. More specifically, the vanilla VAN cannot be trained for $N \geq 64$ because its wall-clock training time exceeds several weeks. HAN gives highly biased estimate that is incompatible with the ground-truth. These results proved the superiority of our RiGCS to the existing generative models for thermodynamic observables estimation.

4.2 QUALITY MEASURES FOR GENERAL OBSERVABLE ESTIMATION

Next, we evaluate the samplers in terms of quality measures related to the bias and variance for general (non-thermodynamic) observables. We use a recently proposed *Effective Mode-Dropping* Measure (EMDM) (Nicoli et al., 2023) and the commonly used Effective Sample Size (ESS) as indicators of bias and variance, respectively. EMDM is defined as EMDM $= \bar{w} = \mathbb{E}_{\widetilde{q}_\theta}[w(\boldsymbol{s})]$, where $w(\boldsymbol{s}) = \frac{p(\boldsymbol{s})}{q_\theta(\boldsymbol{s})}$, and $\widetilde{q}_\theta$ is the renormalized density of $q_\theta(\boldsymbol{s})$ with the very low density areas—in which no sample appears with high probability—eliminated from its support.[6] Nicoli et al. (2023) showed that the bias of the importance-weighted estimators for general observables can be bounded

---

[6] The threshold for the "very low density area" depends on the number M of samples, and $\widetilde{q}_\theta(\boldsymbol{s}) = q_\theta(\boldsymbol{s})$ for $M \to \infty$.

with EMDM. Note that EMDM $\in [0,1]$, and EMDM $= 1$ indicates no effective mode-dropping. ESS (per sample) is defined as ESS $= \frac{1}{\mathbb{E}_{q_\theta}[w(\boldsymbol{s})^2]}$, and is known to be inversely proportional to the variance of general unbiased estimators. Note that ESS $\in [0,1]$ and ESS $= 1$ implies that $q_\theta(\boldsymbol{s}) = p(\boldsymbol{s})$.

Figure 4 left shows the EMDM of our RiGCS in comparison with the baseline generative models, i.e., VAN and HAN. The vanilla VAN can be trained only up to $N = 32$, as explained above. We observe that, for $N \leq 64$, the EMDMs of HAN and RiGCS are compatible with EMDM ≈ 1, indicating no effective mode-dropping. However, for $N = 128$, the EMDM of HAN drops significantly, implying that it is affected by effective mode-dropping. This result is consistent with the biased free energy estimation by HAN in Figure 3 left. Our RiGCS does not show a sign of mode-dropping for $N = 128$, demonstrating its robustness in accurately modeling the target distribution without suffering from mode collapse in high dimensions.

Figure 4 right compares the ESS of RiGCS, and the baseline methods. Our RIGCS outperforms all baselines except the cluster method, which is known to be a powerful state-of-the-art method for general observable estimation. Notably, for $N = 128$, RiGCS improves the ESS of HAN, a state-of-the-art generative model, by a few orders of magnitute, becoming the only generative model with *non-vanishing* ESS.

4.3 COMPUTATION COSTS

Training Costs Generative modeling approaches, i.e., VAN, HAN, and our RiGCS require training. For the $N = 64$ lattice, wall-clock training time for VAN, HAN, and RiGCS are approximately 60 days, 2.8 hours, and 3.8 hours respectively. We also evaluated the advantage of the sequential training with model transfer for RiGCS, introduced in Section 3.3. Figure 3 right compares the achieved ESS by the sequential training (warm start) and random initialization (cold start), where significant advantage is observed. Note that the sequential training (purple curve in Figure 3) starts at the time ≈ 2.3 hours to account for the pre-training time for smaller lattice systems.

Sampling Costs For $N = 64$, sampling costs for MLMC-HB, VAN, HAN, and RiGCS are approximately 14, 27, 0.2 and 0.4 seconds, respectively, for generating a batch of 100 samples.

Empirical sampling and training costs are shown in Appendix H.

5 CONCLUSIONS

Critical behavior such as phase transitions are important phenomena of high relevance in many fields of physics, where Renormalization Group Theory (RGT) plays a central role for theoretical analysis. Insights from RGT were also used for improving tools for numerical analysis, leading to the MultiLevel Monte Carlo (MLMC) methods based on the emerging scale invariance at criticality (SIC). In this paper, we further enhanced such tools by leveraging machine learning techniques. Specifically, we adopted conditional generative models with appropriate size of receptive fields—instead of the nearest-neighbor heat bath conditional samplers—such that they capture long-range and higher-order interactions that exist under slight violation of SIC. With this modification, our Renormalization-informed Generative Critical Sampler (RiGCS) outperforms state-of-the-art generative samplers. We, furthermore, specialized its training procedure with model transfer, again inspired by SIC, and significantly reduced the training cost. Although many machine learning applications for sciences have been proposed, where general domain knowledge, e.g., invariances, equivariances and preservation laws, is incorporated for model design, this work is one of the few applications where the knowledge of critical phenomena, i.e., SIC, is incorporated for the architecture design of machine learning models, as well as its training procedure. We envision this work as a first step towards specialized machine learning methods for critical regimes, which facilitates further developments of efficient algorithms. Our future work includes the application of RiGCS to other physical models, e.g., Potts models and lattice gauge theories, and develop methods combining RiGCS and related methods, e.g., HAN.

REFERENCES

Ryan Abbott, Michael S. Albergo, Denis Boyda, Daniel C. Hackett, Gurtej Kanwar, Fernando Romero-López, Phiala E. Shanahan, and Julian M. Urban. Multiscale Normalizing Flows for Gauge Theories. *PoS*, LATTICE2023:035, 2024. doi: 10.22323/1.453.0035.

A. Aharony. A renormalization group analysis of the phase transitions in disordered systems. *Phys. Rev. B*, 8:4270–4278, 1973. doi: 10.1103/PhysRevB.8.4270.

Sungsoo Ahn, Michael Chertkov, Adrian Weller, and Jinwoo Shin. Bucket renormalization for approximate inference. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 109–118. PMLR, 10–15 Jul 2018. URL `https://proceedings.mlr.press/v80/ahn18a.html`.

M. S. Albergo, G. Kanwar, and P. E. Shanahan. Flow-based generative models for markov chain monte carlo in lattice field theory. *Phys. Rev. D*, 100:034515, Aug 2019. doi: 10.1103/PhysRevD.100.034515. URL `https://link.aps.org/doi/10.1103/PhysRevD.100.034515`.

Constantia Alexandrou, Andreas Athenodorou, Charalambos Chrysostomou, and Srijit Paul. The critical temperature of the 2d-ising model through deep learning autoencoders. *The European Physical Journal B*, 93(12), December 2020. ISSN 1434-6036. doi: 10.1140/epjb/e2020-100506-5. URL `http://dx.doi.org/10.1140/epjb/e2020-100506-5`.

Dimitrios Bachtis, Gert Aarts, Francesco Di Renzo, and Biagio Lucini. Inverse renormalization group in quantum field theory. *Phys. Rev. Lett.*, 128:081603, Feb 2022. doi: 10.1103/PhysRevLett.128.081603. URL `https://link.aps.org/doi/10.1103/PhysRevLett.128.081603`.

Piotr Białas, Piotr Korcyl, and Tomasz Stebel. Analysis of autocorrelation times in neural markov chain monte carlo simulations. *Phys. Rev. E*, 107:015303, Jan 2023. doi: 10.1103/PhysRevE.107.015303. URL `https://link.aps.org/doi/10.1103/PhysRevE.107.015303`.

Indaco Biazzo. The autoregressive neural network architecture of the boltzmann distribution of pairwise interacting spins systems. *Communications Physics*, 6(1), October 2023. ISSN 2399-3650. doi: 10.1038/s42005-023-01416-5. URL `http://dx.doi.org/10.1038/s42005-023-01416-5`.

Indaco Biazzo, Dian Wu, and Giuseppe Carleo. Sparse autoregressive neural networks for classical spin systems. *Machine Learning: Science and Technology*, 5(2):025074, jun 2024. doi: 10.1088/2632-2153/ad5783. URL `https://dx.doi.org/10.1088/2632-2153/ad5783`.

Piotr Białas, Piotr Korcyl, and Tomasz Stebel. Hierarchical autoregressive neural networks for statistical systems. *Computer Physics Communications*, 281:108502, 2022. ISSN 0010-4655. doi: https://doi.org/10.1016/j.cpc.2022.108502. URL `https://www.sciencedirect.com/science/article/pii/S0010465522002211`.

C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, New York, NY, USA, 2006.

Jacob C Bridgeman and Christopher T Chubb. Hand-waving and interpretive dance: an introductory course on tensor networks. *J. Phys. A: Math. Theor.*, 50(22):223001, 2017. doi: 10.1088/1751-8121/aa6dc3. URL `http://stacks.iop.org/1751-8121/50/i=22/a=223001`.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901. Curran Associates, Inc.,

2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

P. Butera and M. Comi. Renormalization group approach to the ising model: Precise evaluation of the wilson functions. *Phys. Rev. B*, 65:144431, 2002. doi: 10.1103/PhysRevB.65.144431.

R. T. Cahill and John B. Kogut. Multigrid monte carlo method for lattice gauge theory. *Physics Letters B*, 114(4):387–392, 1982. doi: 10.1016/0370-2693(82)90062-4.

John Cardy. *Scaling and renormalization in statistical physics*, volume 5. Cambridge university press, 1996.

Juan Carrasquilla and Roger G Melko. Machine learning phases of matter. *Nature Physics*, 13(5):431–434, 2017.

Michele Caselle, Gianluca Costagliola, Alessandro Nada, Marco Panero, and Arianna Toniato. Jarzynski's theorem for lattice gauge theory. *Phys. Rev. D*, 94:034503, Aug 2016. doi: 10.1103/PhysRevD.94.034503. URL https://link.aps.org/doi/10.1103/PhysRevD.94.034503.

Michele Caselle, Elia Cellini, Alessandro Nada, and Marco Panero. Stochastic normalizing flows as non-equilibrium transformations. *Journal of High Energy Physics*, 2022(7):1–31, 2022.

Michele Caselle, Elia Cellini, and Alessandro Nada. Sampling the lattice nambu-goto string using continuous normalizing flows. *Journal of High Energy Physics*, 2024(2):1–28, 2024.

Marco Cè, Leonardo Giusti, and Stefan Schaefer. Domain decomposition, multilevel integration, and exponential noise reduction in lattice qcd. *Phys. Rev. D*, 93:094507, May 2016. doi: 10.1103/PhysRevD.93.094507. URL https://link.aps.org/doi/10.1103/PhysRevD.93.094507.

F. Cérou, P. Del Moral, T. Furon, and A. Guyader. Sequential monte carlo for rare event estimation. *Statistics and Computing*, 22(3):795–808, 2012. doi: 10.1007/s11222-011-9231-6. URL https://doi.org/10.1007/s11222-011-9231-6.

Zhuo Chen, Laker Newhouse, Eddie Chen, Di Luo, and Marin Soljacic. Antn: Bridging autoregressive neural networks and tensor networks for quantum many-body simulation. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 450–476. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/01772a8b0420baec00c4d59fe2fbace6-Paper-Conference.pdf.

Jui-Hua Chung and Ying-Jer Kao. Quantum renormalization group flow with neural networks. *Physical Review Research*, 3(2):023230, 2021. doi: 10.1103/PhysRevResearch.3.023230. URL https://doi.org/10.1103/PhysRevResearch.3.023230.

Jordan Cotler and Semon Rezchikov. Renormalization group flow as optimal transport. *Phys. Rev. D*, 108:025003, Jul 2023a. doi: 10.1103/PhysRevD.108.025003. URL https://link.aps.org/doi/10.1103/PhysRevD.108.025003.

Jordan Cotler and Semon Rezchikov. Renormalizing diffusion models, 2023b. URL https://arxiv.org/abs/2308.12355.

Mary Kathryn Cowles and Bradley P. Carlin. Markov chain monte carlo convergence diagnostics: A comparative review. *Journal of the American Statistical Association*, 91(434):883–904, 1996. doi: 10.1080/01621459.1996.10476956. URL https://www.tandfonline.com/doi/abs/10.1080/01621459.1996.10476956.

Kyle Cranmer, Gurtej Kanwar, Sébastien Racanière, Danilo J Rezende, and Phiala E Shanahan. Advances in machine-learning-based sampling motivated by lattice quantum chromodynamics. *Nature Reviews Physics*, 5(9):526–535, 2023.

Michael Creutz, Laurence Jacobs, and Claudio Rebbi. Monte carlo study of abelian lattice gauge theories. *Phys. Rev. D*, 20:1915–1922, Oct 1979. doi: 10.1103/PhysRevD.20.1915. URL `https://link.aps.org/doi/10.1103/PhysRevD.20.1915`.

Michael Creutz, Laurence Jacobs, and Claudio Rebbi. Monte carlo computations in lattice gauge theories. *Physics Reports*, 95(4):201–282, 1983. ISSN 0370-1573. doi: https://doi.org/10.1016/0370-1573(83)90016-9. URL `https://www.sciencedirect.com/science/article/pii/0370157383900169`.

Francesco D'Angelo and Lucas Böttcher. Learning the ising model with generative neural networks. *Phys. Rev. Res.*, 2:023266, Jun 2020. doi: 10.1103/PhysRevResearch.2.023266. URL `https://link.aps.org/doi/10.1103/PhysRevResearch.2.023266`.

B. Derrida. Renormalization group study of a random ferromagnetic chain. *J. Phys. C: Solid State Phys.*, 13:2997–3014, 1980. doi: 10.1088/0022-3719/13/15/013.

Domenico Di Sante, Matija Medvidović, Alessandro Toschi, Giorgio Sangiovanni, Cesare Franchini, Anirvan M. Sengupta, and Andrew J. Millis. Deep learning the functional renormalization group. *Physical Review Letters*, 129(13):136402, September 2022. doi: 10.1103/PhysRevLett.129.136402. URL `https://link.aps.org/doi/10.1103/PhysRevLett.129.136402`.

Arnaud Doucet, Nando De Freitas, Neil James Gordon, et al. *Sequential Monte Carlo methods in practice*, volume 1. Springer, 2001.

Stavros Efthymiou, Matthew J. S. Beach, and Roger G. Melko. Super-resolving the ising model with convolutional neural networks. *Phys. Rev. B*, 99:075113, Feb 2019. doi: 10.1103/PhysRevB.99.075113. URL `https://link.aps.org/doi/10.1103/PhysRevB.99.075113`.

Michael G. Endres, Richard C. Brower, William Detmold, Kostas Orginos, and Andrew V. Pochinsky. Multiscale monte carlo equilibration: Pure yang-mills theory. *Phys. Rev. D*, 92:114516, Dec 2015. doi: 10.1103/PhysRevD.92.114516. URL `https://link.aps.org/doi/10.1103/PhysRevD.92.114516`.

G. Evenbly and G. Vidal. Class of strongly correlated quantum lattice systems with efficient tensor network representations. *Phys. Rev. Lett.*, 112:240502, 2014. doi: 10.1103/PhysRevLett.112.240502.

G. Evenbly and G. Vidal. Tensor network renormalization. *Phys. Rev. Lett.*, 115:180405, Oct 2015. doi: 10.1103/PhysRevLett.115.180405. URL `https://link.aps.org/doi/10.1103/PhysRevLett.115.180405`.

M. Faas and H.J. Hilhorst. Hierarchical monte carlo simulation of the ising model. *Physica A: Statistical Mechanics and its Applications*, 135(2):571–590, 1986. ISSN 0378-4371. doi: https://doi.org/10.1016/0378-4371(86)90161-5. URL `https://www.sciencedirect.com/science/article/pii/0378437186901615`.

Jacob Finkenrath. Fine grinding localized updates via gauge equivariant flows in the 2D Schwinger model. *PoS*, LATTICE2023:022, 2024. doi: 10.22323/1.453.0022.

M. E. Fisher. Renormalization of critical exponents by hidden variables. *Phys. Rev. Lett.*, 30:1541–1544, 1973. doi: 10.1103/PhysRevLett.30.1541.

C.M. Fortuin and P.W. Kasteleyn. On the random-cluster model: I. introduction and relation to other models. *Physica*, 57(4):536–564, 1972. ISSN 0031-8914. doi: https://doi.org/10.1016/0031-8914(72)90045-6. URL `https://www.sciencedirect.com/science/article/pii/0031891472900456`.

Niklas Gebauer, Michael Gastegger, and Kristof Schütt. Symmetry-adapted generation of 3d point sets for the targeted discovery of molecules. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL `https://proceedings.neurips.cc/paper_files/paper/2019/file/a4d8e2a7e0d0c102339f97716d2fdfb6-Paper.pdf`.

Niklas W. A. Gebauer, Michael Gastegger, Stefaan S. P. Hessmann, Klaus-Robert Müller, and Kristof T. Schütt. Inverse design of 3d molecular structures with conditional generative neural networks. *Nature Communications*, 13(1):973, 2022. doi: 10.1038/s41467-022-28526-y. URL `https://doi.org/10.1038/s41467-022-28526-y`.

Mathis Gerdes, Pim de Haan, Corrado Rainone, Roberto Bondesan, and Miranda CN Cheng. Learning lattice quantum field theories with equivariant continuous flows. *SciPost Physics*, 15:238, 2023.

Mathieu Germain, Karol Gregor, Iain Murray, and Hugo Larochelle. Made: Masked autoencoder for distribution estimation, 2015. URL `https://arxiv.org/abs/1502.03509`.

J. Goodman and A. D. Sokal. Multigrid monte carlo method. conceptual foundations. *Physical Review D*, 33(7):2074–2088, 1986. doi: 10.1103/PhysRevD.33.2074.

W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 04 1970. ISSN 0006-3444. doi: 10.1093/biomet/57.1.97. URL `https://doi.org/10.1093/biomet/57.1.97`.

Markus Hauru, Clement Delcamp, and Sebastian Mizera. Renormalization of tensor networks using graph-independent local truncations. *Phys. Rev. B*, 97:045111, Jan 2018. doi: 10.1103/PhysRevB.97.045111. URL `https://link.aps.org/doi/10.1103/PhysRevB.97.045111`.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

Mohamed Hibat-Allah, Martin Ganahl, Lauren E Hayward, Roger G Melko, and Juan Carrasquilla. Recurrent neural network wave functions. *Physical Review Research*, 2(2):023358, 2020.

Wanda Hou and Yi-Zhuang You. Machine learning renormalization group for statistical physics. *Machine Learning: Science and Technology*, 4(4):045010, oct 2023. doi: 10.1088/2632-2153/ad0101. URL `https://dx.doi.org/10.1088/2632-2153/ad0101`.

Hong-Ye Hu, Dian Wu, Yi-Zhuang You, Bruno Olshausen, and Yubei Chen. Rg-flow: a hierarchical and explainable flow model based on renormalization group and sparse prior. *Machine Learning: Science and Technology*, 3(3):035009, aug 2022. doi: 10.1088/2632-2153/ac8393. URL `https://dx.doi.org/10.1088/2632-2153/ac8393`.

Hui-Yuan Hu, Shuo-Hui Li, Lei Wang, and Yi-Zhuang You. Quantum state compression with machine learning. *Physical Review Research*, 2(2):023369, 2020. doi: 10.1103/PhysRevResearch.2.023369. URL `https://doi.org/10.1103/PhysRevResearch.2.023369`.

John P. Huelsenbeck, Fredrik Ronquist, Rasmus Nielsen, and Jonathan P. Bollback. Bayesian inference of phylogeny and its impact on evolutionary biology. *Science*, 294(5550):2310–2314, 2001. doi: 10.1126/science.1065889. URL `https://www.science.org/doi/abs/10.1126/science.1065889`.

Anette Hulsebos and Roger W. Hockney. Multigrid methods in lattice qcd. *Computers in Physics*, 3 (4):52–60, 1989. doi: 10.1063/1.4822942.

Karl Jansen, Eike H. Müller, and Robert Scheichl. Multilevel monte carlo algorithm for quantum mechanics on a lattice. *Phys. Rev. D*, 102:114512, Dec 2020. doi: 10.1103/PhysRevD.102.114512. URL `https://link.aps.org/doi/10.1103/PhysRevD.102.114512`.

Rajendra P. Joshi, Niklas W. A. Gebauer, Mridula Bontha, Mercedeh Khazaieli, Rhema M. James, James B. Brown, and Neeraj Kumar. 3d-scaffold: A deep learning framework to generate 3d coordinates of drug-like molecules with desired scaffolds. *The Journal of Physical Chemistry B*, 125(44):12166–12176, 11 2021. doi: 10.1021/acs.jpcb.1c06437. URL `https://doi.org/10.1021/acs.jpcb.1c06437`.

Fabian Joswig, Simon Kuberski, Justus T. Kuhlmann, and Jan Neuendorf. pyerrors: A python framework for error analysis of monte carlo data. *Computer Physics Communications*, 288:108750, July 2023. ISSN 0010-4655. doi: 10.1016/j.cpc.2023.108750. URL `http://dx.doi.org/10.1016/j.cpc.2023.108750`.

Leo P. Kadanoff. Scaling laws for ising models near T_c. *Physics Physique Fizika*, 2:263–272, Jun 1966. doi: 10.1103/PhysicsPhysiqueFizika.2.263. URL `https://link.aps.org/doi/10.1103/PhysicsPhysiqueFizika.2.263`.

PW Kasteleyn and CM Fortuin. Phase transitions in lattice systems with random local properties. *Journal of the Physical Society of Japan Supplement*, 26:11, 1969.

Jaron Kent-Dobias and James P. Sethna. Cluster representations and the wolff algorithm in arbitrary external fields. *Phys. Rev. E*, 98:063306, Dec 2018. doi: 10.1103/PhysRevE.98.063306. URL `https://link.aps.org/doi/10.1103/PhysRevE.98.063306`.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014.

Ivan Kobyzev, Simon JD Prince, and Marcus A Brubaker. Normalizing flows: An introduction and review of current methods. *IEEE transactions on pattern analysis and machine intelligence*, 43 (11):3964–3979, 2020. doi: 10.1109/TPAMI.2020.2992934.

Maciej Koch-Janusz and Zohar Ringel. Mutual information, neural networks and the renormalization group. *Nature Physics*, 14(6):578–582, 2018. ISSN 1745-2481. doi: 10.1038/s41567-018-0081-4. URL `https://doi.org/10.1038/s41567-018-0081-4`.

John Kogut and Leonard Susskind. Hamiltonian formulation of wilson's lattice gauge theories. *Phys. Rev. D*, 11:395–408, Jan 1975. doi: 10.1103/PhysRevD.11.395. URL `https://link.aps.org/doi/10.1103/PhysRevD.11.395`.

J M Kosterlitz. The critical properties of the two-dimensional xy model. *Journal of Physics C: Solid State Physics*, 7(6):1046, mar 1974. doi: 10.1088/0022-3719/7/6/005. URL `https://dx.doi.org/10.1088/0022-3719/7/6/005`.

C. H. Lee and Xiao-Liang Qi. Exact holographic mapping in free fermion systems. *Phys. Rev. B*, 93:035112, 2016. doi: 10.1103/PhysRevB.93.035112.

Philipp M. Lenggenhager, Deniz E. Gökmen, Zohar Ringel, Sebastian D. Huber, and Maciej Koch-Janusz. Optimal renormalization group transformation from information theory. *Physical Review X*, 10(1):011037, 2020. doi: 10.1103/PhysRevX.10.011037. URL `https://doi.org/10.1103/PhysRevX.10.011037`.

Shuo-Hui Li and Lei Wang. Neural network renormalization group. *Phys. Rev. Lett.*, 121:260601, Dec 2018. doi: 10.1103/PhysRevLett.121.260601. URL `https://link.aps.org/doi/10.1103/PhysRevLett.121.260601`.

Henry W. Lin, Max Tegmark, and David Rolnick. Why does deep and cheap learning work so well? *Journal of Statistical Physics*, 168(6):1223–1247, July 2017. ISSN 1572-9613. doi: 10.1007/s10955-017-1836-5. URL `http://dx.doi.org/10.1007/s10955-017-1836-5`.

Jin-Guo Liu, Liang Mao, Pan Zhang, and Lei Wang. Solving quantum statistical mechanics with variational autoregressive networks and quantum circuits. *Machine Learning: Science and Technology*, 2(2):025011, feb 2021. doi: 10.1088/2632-2153/aba19d. URL `https://dx.doi.org/10.1088/2632-2153/aba19d`.

Tanguy Marchand, Misaki Ozawa, Giulio Biroli, and Stéphane Mallat. Multiscale data-driven energy estimation and generation. *Phys. Rev. X*, 13:041038, Nov 2023. doi: 10.1103/PhysRevX.13.041038. URL `https://link.aps.org/doi/10.1103/PhysRevX.13.041038`.

Humphrey J. Maris and Leo P. Kadanoff. Teaching the renormalization group. *American Journal of Physics*, 46(6):652–657, June 1978. doi: 10.1119/1.11224.

Humphrey J. Maris and Leo P. Kadanoff. Teaching the renormalization group. *American Journal of Physics*, 46:652–657, 1978. URL `https://api.semanticscholar.org/CorpusID:123119591`.

Nobuyuki Matsumoto, Richard C. Brower, and Taku Izubuchi. Decimation map in 2D for accelerating HMC. *PoS*, LATTICE2023:033, 2023. doi: 10.22323/1.453.0033.

Nicholas Metropolis and S. Ulam. The monte carlo method. *Journal of the American Statistical Association*, 44(247):335–341, 1949. doi: 10.1080/01621459.1949.10483310.

Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953.

Andriy Mnih and Karol Gregor. Neural variational inference and learning in belief networks, 2014. URL `https://arxiv.org/abs/1402.0030`.

Radford M Neal. Annealed importance sampling. *Statistics and computing*, 11:125–139, 2001.

Kim A. Nicoli, Shinichi Nakajima, Nils Strodthoff, Wojciech Samek, Klaus-Robert Müller, and Pan Kessel. Asymptotically unbiased estimation of physical observables with neural samplers. *Phys. Rev. E*, 101:023304, Feb 2020. doi: 10.1103/PhysRevE.101.023304. URL `https://link.aps.org/doi/10.1103/PhysRevE.101.023304`.

Kim A. Nicoli, Christopher J. Anders, Lena Funcke, Tobias Hartung, Karl Jansen, Pan Kessel, Shinichi Nakajima, and Paolo Stornati. Estimation of thermodynamic observables in lattice field theories with deep generative models. *Phys. Rev. Lett.*, 126:032001, Jan 2021. doi: 10.1103/PhysRevLett.126.032001. URL `https://link.aps.org/doi/10.1103/PhysRevLett.126.032001`.

Kim A. Nicoli, Christopher J. Anders, Tobias Hartung, Karl Jansen, Pan Kessel, and Shinichi Nakajima. Detecting and mitigating mode-collapse for flow-based sampling of lattice field theories. *Phys. Rev. D*, 108:114501, Dec 2023. doi: 10.1103/PhysRevD.108.114501. URL `https://link.aps.org/doi/10.1103/PhysRevD.108.114501`.

Hidetoshi Nishimori and Gerardo Ortiz. *Elements of phase transitions and critical phenomena*. Oxford university press, 2011.

Frank Noé, Simon Olsson, Jonas Köhler, and Hao Wu. Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning. *Science*, 365(6457):eaaw1147, 2019. doi: 10.1126/science.aaw1147. URL `https://www.science.org/doi/abs/10.1126/science.aaw1147`.

Lars Onsager. Crystal statistics. i. a two-dimensional model with an order-disorder transition. *Phys. Rev.*, 65:117–149, Feb 1944. doi: 10.1103/PhysRev.65.117. URL `https://link.aps.org/doi/10.1103/PhysRev.65.117`.

George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *J. Mach. Learn. Res.*, 22(1), 2021. URL `https://jmlr.org/papers/volume22/19-1028/19-1028.pdf`.

Jan M Pawlowski and Julian M Urban. Reducing autocorrelation times in lattice simulations with generative adversarial networks. *Mach. Learn.: Sci. Tech.*, 1(4):045011, 2020. doi: 10.1088/2632-2153/abae73.

Xiao-Liang Qi. Exact holographic mapping and emergent space-time geometry. *arXiv preprint arXiv:1309.6282*, 2013.

Alberto Ramos. Automatic differentiation for error analysis of Monte Carlo data. *Comput. Phys. Commun.*, 238:19–35, 2019. doi: 10.1016/j.cpc.2018.12.020.

William T Redman, Tianlong Chen, Zhangyang Wang, and Akshunna S. Dogra. Universality of winning tickets: A renormalization group perspective. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 18483–18498. PMLR, 17–23 Jul 2022. URL `https://proceedings.mlr.press/v162/redman22a.html`.

Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37, pp. 1530–1538. PMLR, 2015. URL `https://proceedings.mlr.press/v37/rezende15.html`.

C.P. Robert and G. Casella. *Monte Carlo statistical methods*. Springer Verlag, 2004.

Dor Ron, Achi Brandt, and Robert H. Swendsen. Adaptive multiscale renormalization group method for two-dimensional ising model. *Physical Review E*, 104(2):025311, 2021. doi: 10.1103/PhysRevE.104.025311. URL `https://doi.org/10.1103/PhysRevE.104.025311`.

Tim Salimans, Andrej Karpathy, Xi Chen, and Diederik P. Kingma. Pixelcnn++: Improving the pixelcnn with discretized logistic mixture likelihood and other modifications. In *5th International Conference on Learning Representations*, 2017. URL `https://openreview.net/forum?id=BJrFC6ceg`.

Stefan Schaefer, Rainer Sommer, and Francesco Virotta. Critical slowing down and error analysis in lattice QCD simulations. *Nucl. Phys. B*, 845:93–119, 2011. doi: 10.1016/j.nuclphysb.2010.11.020.

K. E. Schmidt. Using renormalization-group ideas in monte carlo sampling. *Phys. Rev. Lett.*, 51: 2175–2178, Dec 1983. doi: 10.1103/PhysRevLett.51.2175. URL `https://link.aps.org/doi/10.1103/PhysRevLett.51.2175`.

R. Shankar. Renormalization-group approach to interacting fermions. *Rev. Mod. Phys.*, 66:129–192, Jan 1994. doi: 10.1103/RevModPhys.66.129. URL `https://link.aps.org/doi/10.1103/RevModPhys.66.129`.

Ankur Singha, Dipankar Chakrabarti, and Vipul Arora. Generative learning for the problem of critical slowing down in lattice Gross-Neveu model. *SciPost Phys. Core*, 5:052, 2022. doi: 10.21468/SciPostPhysCore.5.4.052.

Ankur Singha, Dipankar Chakrabarti, and Vipul Arora. Conditional normalizing flow for Markov chain Monte Carlo sampling in the critical region of lattice field theory. *Phys. Rev. D*, 107(1): 014512, 2023a. doi: 10.1103/PhysRevD.107.014512.

Ankur Singha, Dipankar Chakrabarti, and Vipul Arora. Sampling U(1) gauge theory using a retrainable conditional flow-based model. *Phys. Rev. D*, 108(7):074518, 2023b. doi: 10.1103/PhysRevD.108.074518.

Robert H. Swendsen. Monte carlo renormalization group. *Phys. Rev. Lett.*, 42:859–861, Apr 1979. doi: 10.1103/PhysRevLett.42.859. URL `https://link.aps.org/doi/10.1103/PhysRevLett.42.859`.

Robert H. Swendsen and Jian-Sheng Wang. Nonuniversal critical dynamics in monte carlo simulations. *Phys. Rev. Lett.*, 58:86–88, Jan 1987. doi: 10.1103/PhysRevLett.58.86. URL `https://link.aps.org/doi/10.1103/PhysRevLett.58.86`.

Giacomo Torlai and Roger G. Melko. Learning thermodynamics with boltzmann machines. *Phys. Rev. B*, 94:165134, Oct 2016. doi: 10.1103/PhysRevB.94.165134. URL `https://link.aps.org/doi/10.1103/PhysRevB.94.165134`.

Oskar Triebe, Nikolay Laptev, and Ram Rajagopal. Ar-net: A simple auto-regressive neural network for time-series. *arXiv:1911.12436*, 2019.

Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv:1609.03499*, 2016a.

Aaron van den Oord, Nal Kalchbrenner, Lasse Espeholt, koray kavukcuoglu, Oriol Vinyals, and Alex Graves. Conditional image generation with pixelcnn decoders. In *Advances in Neural Information Processing Systems*, volume 29, pp. 4797–4805, 2016b. URL `https://proceedings.neurips.cc/paper/2016/file/b1301141feffabac455e1f90a7de2054-Paper.pdf`.

Aäron van den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 1747–1756. PMLR, 20–22 Jun 2016c. URL `https://proceedings.mlr.press/v48/oord16.html`.

G. Vidal. Class of quantum many-body states that can be efficiently simulated. *Phys. Rev. Lett.*, 101:110501, 2008. doi: 10.1103/PhysRevLett.101.110501.

Jian Wang, Xin Lan, Yuxin Tian, and Jiancheng Lv. MS$ˆ3$d: A RG flow-based regularization for GAN training with limited data. In *Forty-first International Conference on Machine Learning*, 2024. URL `https://openreview.net/forum?id=TuALw8xVum`.

Lingxiao Wang, Yin Jiang, Lianyi He, and Kai Zhou. Continuous-mixture autoregressive networks learning the kosterlitz-thouless transition. *Chinese Physics Letters*, 39(12):120502, dec 2022. doi: 10.1088/0256-307X/39/12/120502. URL `https://dx.doi.org/10.1088/0256-307X/39/12/120502`.

K. G. Wilson. Renormalization group and critical phenomena. i. renormalization group and the kadanoff scaling picture. *Phys. Rev. B*, 4:3174–3183, 1971. doi: 10.1103/PhysRevB.4.3174.

K. G. Wilson and J. Kogut. The renormalization group and the ϵ-expansion. *Phys. Rep.*, 12(2): 75–199, 1974. doi: 10.1016/0370-1573(74)90023-4.

Kenneth G. Wilson. Confinement of quarks. *Phys. Rev. D*, 10:2445–2459, Oct 1974. doi: 10.1103/PhysRevD.10.2445. URL `https://link.aps.org/doi/10.1103/PhysRevD.10.2445`.

Kenneth G Wilson. Monte-carlo calculations for the lattice gauge theory. *Recent developments in gauge theories*, pp. 363–402, 1980.

Ulli Wolff. Collective monte carlo updating for spin systems. *Phys. Rev. Lett.*, 62(4):361, 1989a. doi: https://doi.org/10.1103/PhysRevLett.62.361.

Ulli Wolff. Comparison between cluster monte carlo algorithms in the ising model. *Phys. Lett. B*, 228(3):379–382, 1989b. doi: https://doi.org/10.1103/PhysRevE.105.015313.

Ulli Wolff. Critical slowing down. *Nuclear Physics B*, 17:93–102, 1990. doi: https://doi.org/10.1016/0920-5632(90)90224-I.

Ulli Wolff. Monte carlo errors with less errors. *Comp. Phys. Comm.*, 156(2):143–153, 2004. ISSN 0010-4655. doi: https://doi.org/10.1016/S0010-4655(03)00467-3. URL `https://www.sciencedirect.com/science/article/pii/S0010465503004673`.

Dian Wu, Lei Wang, and Pan Zhang. Solving statistical mechanics using variational autoregressive networks. *Phys. Rev. Lett.*, 122:080602, Feb 2019. doi: 10.1103/PhysRevLett.122.080602. URL `https://link.aps.org/doi/10.1103/PhysRevLett.122.080602`.

F. Y. Wu. The potts model. *Rev. Mod. Phys.*, 54:235–268, Jan 1982. doi: 10.1103/RevModPhys.54.235. URL `https://link.aps.org/doi/10.1103/RevModPhys.54.235`.

Shuo Yang, Zheng-Cheng Gu, and Xiao-Gang Wen. Loop optimization for tensor network renormalization. *Phys. Rev. Lett.*, 118:110504, Mar 2017. doi: 10.1103/PhysRevLett.118.110504. URL `https://link.aps.org/doi/10.1103/PhysRevLett.118.110504`.

A EXTENDED RELATED WORK

Renormalization Group Theory (RGT) has significantly impacted the study of statistical systems, especially in analyzing critical phenomena and phase transitions. The pioneering works by Wilson, Kogut, and Kadanoff laid the foundational principles of RG theory (Wilson, 1971; Wilson & Kogut, 1974; Kadanoff, 1966). Subsequent advancements have expanded the application of RGT-inspired methods to disordered systems, random ferromagnetic chains, and Monte Carlo simulations (Aharony, 1973; Fisher, 1973; Swendsen, 1979; Derrida, 1980; Butera & Comi, 2002). These developments have enriched the understanding of critical phenomena and further established the applications of RGT techniques in both theoretical and computational contexts.

In computational physics, substantial research has focused on RGT-inspired sampling methods for lattice simulations. For example, early studies on the Ising model (Schmidt, 1983; Faas & Hilhorst, 1986) achieved notable success in small systems in one dimension yet faced limitations as the lattice sizes increased. More recently, Jansen et al. (2020) introduced a new theoretical framework of low variance estimation, showing promising results for one-dimensional quantum systems. In lattice gauge theory, the application of RGT concepts led to the development of several algorithms, including multigrid (Cahill & Kogut, 1982; Goodman & Sokal, 1986; Hulsebos & Hockney, 1989), multiscale thermalization techniques (Endres et al., 2015), and decimation maps (Matsumoto et al., 2023).

In recent years, RGT-inspired approaches have been combined with machine learning to develop more scalable samplers. Notable examples include applications in $U(1)$ (Finkenrath, 2024) and $SU(3)$ gauge theories (Abbott et al., 2024) where a renormalization group (RG) scheme has been combined with normalizing flows. Similarly to our approach, Białas et al. (2022) proposed a Hierarchical Autoregressive Network (HAN) for sampling configurations of the 2D Ising model. This latter leverages a recursive domain decomposition (Cè et al., 2016) technique in which different regions of the configurations are sampled in parallel using the same autoregressive network, thus replacing the traditional scaling with the system's linear extent L. Białas et al. (2022) demonstrated the effectiveness of HAN on the two-dimensional Ising model, with simulations on lattices up to 128×128 spins. However, this method has several shortcomings, particularly in terms of performance on larger lattices. We refer to Appendix H and Section 4 for a thorough discussion. On a side note, we emphasize that our multilevel approach and the domain decomposition proposed in Białas et al. (2022) are not mutually exclusive. In fact, those methods could in principle be combined leading to more powerful sampling protocols. We defer this investigation to future work.

Besides the development of enhancing sampling methods other recent works have leveraged the idea of RG in different ways. In their work, Li & Wang (2018) focused on *neural network renormalization group*, investigating the capability of neural networks to perform hierarchical feature extraction and hierarchical transformations. Koch-Janusz & Ringel (2018) propose a machine learning approach to identify the relevant degrees of freedom and extract Ising critical exponents in one and two-dimensional systems. Efthymiou et al. (2019) leverage the idea of image super-resolution and train convolutional neural networks that invert real-space renormalization decimations, and show that it is possible to predict thermodynamic quantities for lattice sizes larger than those used in training. Lenggenhager et al. (2020) draw a connection between real-space RG and real-space mutual information. From an information-theoretic standpoint, they investigate the information loss at arbitrary coarse graining of the lattices through the lenses of RG. Another important work (Li & Wang, 2018) propose to use techniques allowing relaxation to continuous variable to enhance HMC sampling for the Ising model. To this end, they use bijective transformation to learn hierarchical maps to automatically identify mutually independent collective variables. While inspired by RGT, their work does not focus on multilevel sampling, e.g., the approach does not scale and in the paper only results on a 16^2 lattice are shown. A more recent study (Marchand et al., 2023) introduce the *wavelet-conditional renormalization group* (WCRG) where fast wavelet transforms are used to build an RG transformation across scales. While similar in spirit, their approach is substantially different compared to ours, as it trains the model by using a contrastive divergence loss and requires a lot of training samples drawn from the target distributions. Hu et al. (2020) use the neural network renormalization group (Li & Wang, 2018) as a universal approach to design generic exact holographic maps (EHM) for interacting field theories. Chung & Kao (2021) use Restricted Boltzmann Machines (RBM) to learn a valid real-space RG transformation without prior knowledge of the physical system, establishing a solid connection between the RG transformation in physics and statistical learning theory. Ron et al. (2021) and Bachtis et al. (2022), instead, use modified block-spin transformations—to improve convergence in the Monte Carlo (MC) renormalization group trajectory—and inverse RG transformations, respectively, to extract critical exponents of a given physical theory.

Recently, a so-called machine-learning renormalization group (MLRG) algorithm has been developed to explore and analyze many-body lattice systems in statistical physics (Hou & You, 2023). In a recent work by Di Sante et al. (2022), the authors propose a data-driven dimensionality reduction and use a Neural ODE solver in a low-dimensional latent space to efficiently learn the functional RG dynamics. The authors showed promising results in the context of the Hubbard model.

Lin et al. (2017) pointed out that convolutional neural networks, in supervised learning tasks, can act as a "coarse-graining" procedure, isolating relevant macroscopic features from irrelevant microscopic noise. In recent years, RG-inspired machine learning applications have emerged in the context of variational inference (Ahn et al., 2018), regularization techniques (Wang et al., 2024), transfer learning (Redman et al., 2022), and multi-scale semantic manipulation of images (Hu et al., 2022). While all these related works leverage the concept of renormalization group (RG) in different ways—such as for extracting critical exponents, or interpreting RG as coarse-graining procedures in machine learning—they suffer from a few shortcomings when it comes to efficient sampling. First, they often lack the access to a tractable probability density and, second, do not allow for data-free training of a neural sampler as well as rapid and effective sampling at multiple scales—as RiGCS does.

RG ideas have also been used in the context of Tensor Networks (TN) to construct an emergent scale invariant description for critical systems. TN describe the wave function or the partition function of a system as a contraction of a network of smaller tensors. This approach can be shown to be efficient as long as the entanglement in the system is moderate (Bridgeman & Chubb, 2017). Blocking tensors together and coarse gaining the system allow for (numerically) obtaining a description of the system at a larger length scale. Prominent algorithms for coarse graining the partition function of a critical system are the Tensor Network Renormalization group (TNR) (Evenbly & Vidal, 2015) and loop TNR (Yang et al., 2017) for square lattices, as well as Graph-Independent Local Truncations (GILT) for arbitrary graphs (Hauru et al., 2018). The Multi-scale Entanglement Renormalization Ansatz (MERA) (Vidal, 2008; Evenbly & Vidal, 2014) leverages the hierarchical structure of RG to efficiently represent quantum states for critical systems in 1+1 dimensions that are described by an underlying conformal field theory. Generalizing on this idea, and understanding holographic duality as a generalization of the RG flow, Qi (2013) and Lee & Qi (2016) propose an exact holographic mapping which is a one-to-one unitary mapping between boundary and bulk degrees of freedom. In comparison to MC methods, TN approaches enable the direct computation of expected values of observables, as they provide an approximation for the wave function or the partition function of a system. However, although the numerical algorithms for TN methods, which allow for recovering exact scale invariance at the critical point, scale polynomially with respect to both system size and tensor size, the computational cost remains challenging due to large degree in the tensor size χ (the leading order cost of TNR is $\mathcal{O}(\chi^6)$, and of MERA up to $\mathcal{O}(\chi^9)$).

Furthermore, Cotler and Rezchikov have also uncovered intriguing connections between RG theory and optimal transport (Cotler & Rezchikov, 2023a), and diffusion models (Cotler & Rezchikov, 2023b), highlighting promising new directions for further investigation.

B MARKOV CHAIN MONTE CARLO AND CLUSTER METHODS

B.1 MARKOV CHAIN MONTE CARLO

MCMC methods produce a sequence of configurations $\{\boldsymbol{s}_1, \boldsymbol{s}_2, \dots\}$ following a distribution p through a Markov chain. To this end, starting from a configuration $\boldsymbol{s}_i$ a new configuration $\boldsymbol{s}'$ is proposed that is either accepted or rejected. In case it is accepted, it becomes the next member of the Markov chain $\boldsymbol{s}_{i+1}$. If it is rejected, one continues to propose trail configurations starting from $\boldsymbol{s}_i$ until one is eventually accepted. In order to ensure that the configurations produced follow the traget distribution, the transition probability $T(\boldsymbol{s} \to \boldsymbol{s}')$ from configuration $\boldsymbol{s}$ to $\boldsymbol{s}'$ has to fulfill

$$\sum_{\boldsymbol{s}} p(\boldsymbol{s}) T(\boldsymbol{s} \to \boldsymbol{s}') = p(\boldsymbol{s}') \tag{14}$$

One way of guaranteeing the condition above fulfilled is to ensure that the transition probabilities of the Monte Carlo scheme fulfill the detailed balance condition

$$p(\boldsymbol{s}) T(\boldsymbol{s} \to \boldsymbol{s}') = p(\boldsymbol{s}') T(\boldsymbol{s}' \to \boldsymbol{s}). \tag{15}$$

Together with ergodicity, i.e. the possibility of reaching any configuration from another one with a succession of trail moves, this assures that after equilibrating, the configurations produced by the Markov process follow the target distribution $p(\boldsymbol{s})$.

Algorithms such as the Metropolis-Hastings and Heatbath algorithms are widely used to efficiently generate samples in these simulations. Once enough configurations are sampled, physical observables such as energy, magnetization, and correlation functions can be computed by averaging over the configurations:

$$\langle \mathcal{O} \rangle \approx \frac{1}{M} \sum_{m=1}^{M} \mathcal{O}(\boldsymbol{s}_m),$$

where M is the number of sampled configurations.

MCMC samples are inherently correlated, because each new configuration depends on the previous one. This correlation is measured by the autocorrelation time, which indicates the extent to which samples remain correlated. As one approaches the critical point, the relaxation time τ of thermodynamic properties diverges as a power law of the correlation length ξ, i.e.,

$$\tau \propto \xi^z, \tag{16}$$

where z is the dynamical critical exponent. As a result, also the autocorrelation time of the MCMC method diverges close to the critical point, because $\xi \to \infty$ as one approaches SIC, which is known as critical slowing down (Wolff, 1990). For the finite hypercubic lattices we consider, the correlation length in lattice units is bounded by the extent N of the lattice in each dimension, hence one finds

$$\tau \propto N^z, \tag{17}$$

as one goes close to the critical point. Eq. (17) shows that depending on the value of z, creating independent configurations becomes increasingly challenging close to criticality for growing lattice sizes. For local update schemes, as for example Metropolis-Hastings, one typically obtains values of $z \approx 2$. This increase in autocorrelation necessitates a larger number of samples to achieve accurate statistical estimates, thereby raising the computational cost (Schaefer et al., 2011). In contrast, cluster algorithms can yield dynamical critical exponents close to zero, thus avoiding critical slowing down.

B.2 CLUSTER ALGORITHMS

In the following, we briefly review two cluster algorithms for the Ising model, which allow for efficient simulations close to the critical point. In particular, the Wolff algorithm, combined with AIS, was used as the Cluster-AIS baseline in the main text.

B.2.1 SWENDSEN-WANG ALGORITHM

The basic principle of the Swendsen-Wang algorithm is to flip entire clusters of spins instead of a single one (Swendsen & Wang, 1987). To this end, a given spin configuration is divided into clusters by assigning bonds between the spins. A cluster then consists of all spins connected directly or indirectly via a bond. Subsequently, all the spins belonging to a cluster are flipped collectively. More specifically, the algorithm consists of the following steps starting from a given configuration $\boldsymbol{s}$:

1. Inspect all nearest neighbors s_i, s_j in $\boldsymbol{s}$. If s_i and s_j are aligned in the same direction, a bond is formed in between them with probability $p_{ij} = 1 - \exp(-2\beta J)$. If they are antiparallel, no bond is formed.
2. Identify all clusters, i.e., all sets of spins connected either directly ore indirectly by a bond.
3. Flip all spins within each cluster collectively with a certain probability p^{flip}, resulting in a spin configuration $\boldsymbol{s}'$.
4. Delete all bonds and and repeat the steps for the new spin configuration $\boldsymbol{s}'$.

Step 3 is based on the fact, that the Ising model partition function can be written as a sum over all possible clusters, where the individual clusters are uncorrelated (Kasteleyn & Fortuin, 1969; Fortuin & Kasteleyn, 1972). As a result, the spins within each cluster can be flipped independently. If p^{flip} is chosen to be close to zero, the new configuration $\boldsymbol{s}'$ will, in general, not differ a lot from $\boldsymbol{s}$. In contrast, choosing $p^{\text{flip}} = 1$ will result in a full inversion of the configuration $\boldsymbol{s}$, which does not change the energy at all. As both extremal cases do not produce sensible new configurations, p^{flip} is typically set to $1/2$.

Intuitively, the effectiveness of the Swendsen-Wang algorithm can be understood by the fact that flipping large clusters allows for efficiently destroying the long-range correlations emerging close to the critical point. For $D > 2$, the Swendsen-Wang algorithm becomes less capable, as the majority of clusters formed tend to be small, with only a few large ones being generated.

B.2.2 WOLFF ALGORITHM

The Wolff algorithm (Wolff, 1989a) is a single-cluster variant of the Swendsen-Wang algorithm. Instead of dividing the entire configuration in clusters and flipping each of them, the Wolff algorithm only forms a single cluster and collectively flips the spins inside this cluster. Starting from a given configuration $\boldsymbol{s}$, the Wolff algorithm proceeds with the following steps:

1. Choose a random spin s_i within the configuration.
2. Starting from s_i, form bonds analogous to the Swendsen-Wang algorithm with probability $p_{ij} = 1 - \exp(-2\beta J)$ with all nearest neighbors s_j that are aligned parallel to s_i.
3. For each neighboring spin s_j added to the cluster form bonds with its respective neighbors that are not in the cluster, according to step 2.
4. Repeat steps 2 and 3 iteratively until no more spins can be added to the cluster.
5. Flip all spins within the cluster and obtain a spin configuration $\boldsymbol{s}'$.
6. Delete all bonds and and repeat the steps for the new spin configuration $\boldsymbol{s}'$.

Note that compared to the Swendsen-Wang algorithm, the cluster is flipped with certainty. If the cluster formed by the Wolff algorithm is large, the long-range correlations are broken up essentially as effectively as in the Swendsen-Wang algorithm, but without the extra effort of having to form smaller clusters in the remainder of the system. If the cluster formed by the Wolff algorithm is small, the configuration does not change significantly, however, at the same time, the computational effort is also small. Thus, the Wolff algorithm turns out to be even more efficient in decreasing the dynamical critical exponent z as the Swendsen-Wang approach. Therefore, we used the Wolff algorithm as the *Cluster* algorithm in our experiments.

C RENORMALIZATION GROUP

The Renormalization Group (RG) (Wilson, 1971; Cardy, 1996) is a powerful framework in theoretical physics for studying the behavior of systems as they are progressively coarse-grained to larger length scales. During this process, microscopic degrees of freedom are systematically marginalized, generating a flow in parameter space known as the RG flow. More formally, given a Hamiltonian H describing the system at a given length-scale, one can define a RG transformation $\mathcal{R}_l$

$$H' = \mathcal{R}_l [H] , \tag{18}$$

which changes the scale of the system and yields a Hamiltonian H' describing the system at larger length scale l with less degrees of freedom. For $\mathcal{R}_l$ to be a proper RG transformation, it has to fulfill the semi-group property, i.e. there is a neutral element $\mathcal{R}_{\text{id}}$ that does not change the scale and the composition of two transformations to different length scales l and l' has to fulfill $\mathcal{R}_{l'} \circ \mathcal{R}_{l'} = \mathcal{R}_{l'+l}$. This transformation generates a flow on the space of Hamiltonians that can yield crucial insights into the macroscopic properties of physical systems. In particular, critical points correspond to fixed points H^* in the RG flow,

$$H^* = \mathcal{R}_l [H^*] , \tag{19}$$

as the system exhibits scale invariance at criticality. Close to the critical point, one can expres the Hamiltonian of the system as $H = H^* + \delta H$, where δH is a small perturbation. Expanding the RG transformation around the fixed point, one finds

$$\mathcal{R}_l [H^* + \delta H] = H^* + \mathcal{L} [H^*] \, \delta H + \mathcal{O}\left(\delta H^2\right) \approx H^* + \delta H' , \tag{20}$$

where $\delta H' = \mathcal{L} [H^*] \, \delta H$. Applying the transformation $\mathcal{R}_l$ n-times, we find in leading order

$$\mathcal{R}_{n \times l} [H^* + \delta H] \approx H^* + \mathcal{L} [H^*]^n \, \delta H . \tag{21}$$

Expanding δH in the eigenoperators M_m of $\mathcal{L}\left[H^*\right]$, one finds that the leading order correction can be expressed as

$$\mathcal{L}\left[H^*\right]^n \delta H = \sum_m c_m \lambda_m^n M_m, \tag{22}$$

where c_m are the expansion coefficients and λ_m the eigenvalues corresponding to the eigenoperator M_m. For large n, or equivalently at large length scales, one observes that the λ_m determine the behavior of the system: for $\lambda_m < 1$ ($\lambda_m > 1$) the corresponding eigenoperator is called *irrelevant* (*relevant*), for $\lambda_m = 1$ the operator is called *marginally relevant*. Equation (22) shows that the relevant and marginal operators determine the macroscopic behavior of the system. Thus, close to the critical point systems having the same (marginally) relevant operators will show the same behavior at macroscopic scales, regardless of the microscopic degrees of freedom. This gives rise to the notion of universality classes, i.e. physical systems showing the same scaling behavior at criticality described by typically a few critical exponents, despite being microscopically different. Thus, information about a system's behavior close to criticality can be obtained by studying another model within the same universality class. Namely, the Ising model, originally developed to describe phase transitions in ferromagnetic systems, can also be used to study the liquid-gas transition, superfluids, and the Higgs mechanism (Wilson, 1971; Wilson & Kogut, 1974).

A simple example of an RG transformation is the Kadanoff block spin transformation, which will be illustrated below. The partition function of the Ising Hamiltonian is given by

$$Z = \sum_{\boldsymbol{s}} \exp\left(-\beta H(s)\right) = \sum_{\boldsymbol{s}} \exp\left(\beta\, \boldsymbol{s} \boldsymbol{J}^{\mathrm{NN}} \boldsymbol{s}^\top\right). \tag{23}$$

For $D = 1$ this can be rewritten as (Maris & Kadanoff, 1978)

$$\begin{aligned} Z &= \sum_{s_1, s_3, s_5, \ldots} \left(\sum_{s_2, s_4, s_6, \ldots} e^{K(s_1 s_2 + s_2 s_3)} e^{K(s_3 s_4 + s_4 s_5)} \ldots \right) & (24) \\ &= \sum_{s_1, s_3, s_5, \ldots} \left[e^{K(s_1 + s_3)} + e^{-K(s_1 + s_3)} \right] \left[e^{K(s_3 + s_5)} + e^{-K(s_3 + s_5)} \right] \ldots, & (25) \end{aligned}$$

where $K = \beta J$, we have separated the sum over the even and odd spins in the first line, and explicitly performed the sum over the even spins from the first to the second line. Using that the spins can take values ±1 the following identity holds

$$e^{K(s_1 + s_3)} + e^{-\beta J(s_1 + s_3)} = f(K) e^{K'\, s_1 s_3}, \tag{26}$$

where

$$\begin{aligned} f(K) &= 2\sqrt{\cosh(2K)}, \\ K' &= \ln(\cosh(2K))/2. \end{aligned} \tag{27}$$

Inserting this into Eq. (25), one finds

$$Z = f(K)^{N/2} \sum_{s_1, s_3, s_5, \ldots} \exp(K' H_{L-1}) = f(K)^{N/2} \sum_{\boldsymbol{s}^{L-1}} \exp\left(K' H_{L-1}(\boldsymbol{s}^{L-1})\right). \tag{28}$$

This demonstrates that the partition function of the system on the fine lattice is related to the one on a coarser lattice described by the same type of Hamiltonian at a different value of βJ. Moreover, looking at Eq. (27), the only fixed points K^* in the recursion relation for the renormalized couplings are the trivial ones, i.e. $K^* = 0$, corresponding to $T \to \infty$, where system is in the paramagnetic phase, and $K^* = \infty$, corresponding to $T = 0$, the system is in the ferromagnetic phase.

For $D = 2$ one can follow a similar approach by marginalizing at each iteration over the even (or odd) degrees of freedom in a "checker board" pattern (see Fig. 5). After a single step of the procedure, one obtains for the partition function (Maris & Kadanoff, 1978)

$$Z = f(K)^{N/2} \sum_{\boldsymbol{s}^{L-1}} \exp\left(K_1 \sum_{\langle ij \rangle} s_i s_j + K_2 \sum_{\langle\langle ij \rangle\rangle} s_i s_j + K_3 \sum_{\square} s_i s_j s_r s_t \right)$$

where $\langle ij \rangle$ corresponds to the nearest neighbors on the lattice after summing over one sublattice, $\langle\langle ij \rangle\rangle$ to spins on next-nearest neighbor sites and $\square$ indicates the spin on a plaquette of the coarser lattice and

$$K_1 = \frac{1}{4} \ln\left(\cosh\left(4K\right)\right), \; K_2 = \frac{1}{8} \ln\left(\cosh\left(4K\right)\right), \; K_3 = \frac{1}{8} \ln\left(\cosh\left(4K\right)\right) - \frac{1}{2} \ln\left(\cosh\left(2K\right)\right).$$

Note that in this case, the partition function is not just given bye the exponential of the same type of Hamiltonian as the original model just with different parameters on a coarser lattice (see Fig. 5). Continuing this procedure one would generate various long-range and multi-body interactions in the renormalized Hamiltonian, which is captured in the Hamiltonian of Eq. (9).

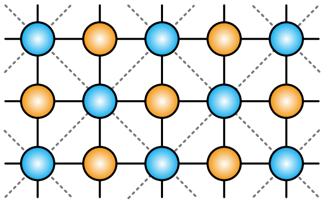

Figure 5: Illustration of the Kadanoff block spin method on a square lattice, the spheres indicate the spins and the solid black lines the original lattice. After summing over the configurations the orange spins, one obtains an renormalized Hamiltonian for the blue spins on a square lattice that is tilted by 45°compared to the original lattice (indicated by the grey dashed lines). The resulting Hamiltonian on the dashed lattice contains nearest-neighbor interactions, interactions of all four spins along a square as well as next nearest-neighbor interactions along the diagonal of the squares.

A practical example of RG flow in machine learning is the application of CNNs to a classification supervised learning problem (Lin et al., 2017). CNNs perform a form of coarse-graining, where successive convolutional layers progressively filter out microscopic noise (irrelevant operators) and isolate high-level features (relevant operators) essential for distinguishing the target classes. The latter example can be tested by training a CNN to classify the phase (ferromagnetic or paramagnetic) of the Ising model. As shown in (Carrasquilla & Melko, 2017) the output of such CNN is strongly correlated with the magnetization, indicating that both neural networks and the RG flow capture the same key parameter—magnetization—as a relevant feature to characterize the phase transition in the Ising model. A similar behavior can be observed by studying the latent representation of an autoencoder trained on Ising configurations Alexandrou et al. (2020).

D MULTILEVEL MONTE CARLO WITH HEAT BATH (MLMC-HB) ALGORITHM

With the site partitioning (see Figure 1) based on the block-spin transformations (Kadanoff, 1966), MLMC-HB performs ancestral sampling, according to Eq. (8), i.e.,

$$q^{\mathrm{NN}}(\boldsymbol{s}) = p(\boldsymbol{s}^L|\boldsymbol{s}^{\leq L-1}) \left(\prod_{l=1}^{L-1} q^{\mathrm{NN}}(\boldsymbol{s}^l|\boldsymbol{s}^{\leq l-1}) \right) q^{\mathrm{NN}}(\boldsymbol{s}^0),$$

which approximates the target distribution (7):

$$p(\boldsymbol{s}) = \left(\prod_{l=1}^{L} p(\boldsymbol{s}^l|\boldsymbol{s}^{\leq l-1}) \right) p(\boldsymbol{s}^0).$$

Here,

$$q^{\mathrm{NN}}(\boldsymbol{s}^{\leq l}) = \left(\prod_{l'=1}^{l-1} q^{\mathrm{NN}}(\boldsymbol{s}^{l'}|\boldsymbol{s}^{\leq l'-1}) \right) q^{\mathrm{NN}}(\boldsymbol{s}^0)$$

for $l = 0, \ldots, L-1$ approximates the true marginal distribution (4):

$$p(\boldsymbol{s}^{\leq l}) = \int p(\boldsymbol{s})\mathcal{D}[\boldsymbol{s}^{>l}] \equiv \tfrac{1}{Z_l} e^{-\beta H_l(\boldsymbol{s}^{\leq l})}$$

with NN interation Hamiltonians, i.e.,

$$q^{\mathrm{NN}}(\boldsymbol{s}^{\leq l}) = \tfrac{1}{\widetilde{Z}_l} e^{-\beta \widetilde{H}_l(\boldsymbol{s}^{\leq l})} \quad \text{with} \quad \widetilde{H}_l(\boldsymbol{s}^{\leq l}) = -\boldsymbol{s}^{\leq l} \boldsymbol{J}_l^{\mathrm{NN}} (\boldsymbol{s}^{\leq l})^\top.$$

Based on RGT (Appendix C), the interaction coefficients $\{J_l\}$ (J in Eq. (2) for each $\boldsymbol{J}_l^{\mathrm{NN}}$) are computed by the following recursive equation in the two-dimensional Ising model (Maris & Kadanoff, 1978): for lattice spacing $a_l = \frac{a_{l-1}}{\sqrt{2}}$, i.e., 45 degree rotated lattice,

$$K_{l-1} = \frac{3}{8} \log\{\cosh(4K_l)\}, \tag{29}$$

where $K_l = \beta J_l$ for $l = L, \dots, 1$ with $J_L = 1$.

Thanks to the site partitioning and the approximation with NN interactions, the conditional sampling probability can be fully decomposed into independent distributions as

$$q^{\mathrm{NN}}(\boldsymbol{s}^l|\boldsymbol{s}^{\le l-1}) = \textstyle\prod_{v=1}^{V_l} q^{\mathrm{NN}}(s_v^l|\boldsymbol{s}^{\le l-1}) \tag{30}$$

(see the Markov blankets shown Figure 1). Sampling from the independent distribution $q^{\mathrm{NN}}(s_v^l|\boldsymbol{s}^{\le l-1})$ can be easily performed by the heat bath (HB) algorithm. For the Ising models, where $\boldsymbol{s} \in \{-1,1\}^V$, s_v^l can be drawn as

$$q^{\mathrm{NN}}(s_v^l|\boldsymbol{s}^{\le l-1}) = \frac{\exp\left(-s_v^l \sum_{v'} (\boldsymbol{J}_l^{\mathrm{NN}})_{v,v'} s_{v'}^{<l-1}\right)}{\exp\left(\sum_{v'} (\boldsymbol{J}_l^{\mathrm{NN}})_{v,v'} s_{v'}^{<l-1}\right) + \exp\left(-\sum_{v'} (\boldsymbol{J}_l^{\mathrm{NN}})_{v,v'} s_{v'}^{<l-1}\right)}.$$

In the 2D Ising model, each site has 4 nearest neighbors, allowing us to easily sample each site using a probability versus state space table. In the case of MLMC-HB, which involves nearest-neighbor interactions governed by the recursion relation (29) for J_l, the sampling density can be tuned by adjusting J at different levels to improve performance.

E ALGORITHMIC DETAILS OF RIGCS

In the following, we provide a detailed description of the sequential training with model transfer specialized for RiGCS. Furthermore, we also provide pseudocode for both training and sampling in Algorithm 1 and Algorithm 2, respectively.

E.1 DETAILS OF SEQUENTIAL TRAINING WITH MODEL TRANSFER INITIALIZATION

As mentioned in Section 3.3, we train our RiGCS by minimizing the reverse Kullback-Leibler (KL) divergence (11), which can suffer from long initial random walking steps if the training parameters $\boldsymbol{\theta}$ are not well initialized, e.g., by randomly initialization. This is because a randomly initialized RiGCS, $q_{\boldsymbol{\theta}}(\boldsymbol{s})$, generates random samples, for which the stochastic gradient of the objective (11) rarely provides useful signal to train the model for a large lattice system. We tackle this problem with a specialized training procedure for RiGCS with *model transfer*, again based on RGT.

We choose L to an even number, and consider a set of *sequential target* Boltzmann distributions $\{p_{L'}(\boldsymbol{s}^{\le L'}) \propto e^{-\beta \widetilde{H}_{L'}(\boldsymbol{s}^{\le L'})}; L' = 0, 2, 4, \dots, L\}$, where $\{\widetilde{H}_l(\boldsymbol{s}^{\le l})\}_{l=0}^{L}$ are the approximate Hamiltonians with NN interactions, defined in Eq.(5), and $\widetilde{H}_L(\boldsymbol{s}^{\le L}) = H(\boldsymbol{s})$. Let us consider the corresponding set of RiGCSs $\{q_{\boldsymbol{\theta}_0}(\boldsymbol{s}^0), \{q_{\boldsymbol{\theta}_{\le L'-1}}(\boldsymbol{s}^{\le L'}); L' = 2, 4, \dots, L\}\}$ that share the same coarsest lattice size V_0. We train the RiGCSs to the sequential targets in the increasing order of L', namely,

At level 0, the 0-layered RiGCS (plain VAN) $q_{\boldsymbol{\theta}_0}(\boldsymbol{s}_0)$ is trained on $p_0(\boldsymbol{s}^0) \propto e^{-\beta \widetilde{H}_0(\boldsymbol{s}^0)}$,

At level 2, the 2-layered RiGCS $q_{\boldsymbol{\theta}_{\le 1}}(\boldsymbol{s}_{\le 2}) = p(\boldsymbol{s}^2|\boldsymbol{s}^{\le 1}) q_{\boldsymbol{\theta}_1}(\boldsymbol{s}^1|\boldsymbol{s}^0) q_{\boldsymbol{\theta}_0}(\boldsymbol{s}_0)$

is trained on $p_2(\boldsymbol{s}^{\le 2}) \propto e^{-\beta \widetilde{H}_2(\boldsymbol{s}^{\le 2})}$,

At level 4, the 4-layered RiGCS $q_{\boldsymbol{\theta}_{\le 3}}(\boldsymbol{s}_{\le 4}) = p(\boldsymbol{s}^4|\boldsymbol{s}^{\le 3}) \left(\prod_{l=1}^{3} q_{\boldsymbol{\theta}_l}(\boldsymbol{s}^l|\boldsymbol{s}^{\le l-1}) \right) q_{\boldsymbol{\theta}_0}(\boldsymbol{s}^0)$

is trained on $p_4(\boldsymbol{s}^{\le 4}) \propto e^{-\beta \widetilde{H}_4(\boldsymbol{s}^{\le 4})}$,

$\vdots$

At level L', the L'-layered RiGCS $q_{\boldsymbol{\theta}_{\le L'-1}}(\boldsymbol{s}_{\le L'}) = p(\boldsymbol{s}^{L'}|\boldsymbol{s}^{\le L'-1}) \left(\prod_{l=1}^{L'-1} q_{\boldsymbol{\theta}_l}(\boldsymbol{s}^l|\boldsymbol{s}^{\le l-1}) \right) q_{\boldsymbol{\theta}_0}(\boldsymbol{s}^0)$

is trained on $p_{L'}(\boldsymbol{s}^{\le L'}) \propto e^{-\beta \widetilde{H}_{L'}(\boldsymbol{s}^{\le L'})}$,

$\vdots$

At level L, the L-layered RiGCS $q_{\boldsymbol{\theta}_{\le L-1}}(\boldsymbol{s}_{\le L}) = p(\boldsymbol{s}^{L}|\boldsymbol{s}^{\le L-1}) \left(\prod_{l=1}^{L-1} q_{\boldsymbol{\theta}_l}(\boldsymbol{s}^l|\boldsymbol{s}^{\le l-1}) \right) q_{\boldsymbol{\theta}_0}(\boldsymbol{s}^0)$

is trained on $p_{L}(\boldsymbol{s}^{\le L}) \propto e^{-\beta \widetilde{H}_{L}(\boldsymbol{s}^{\le L})}$.

Figure 2 illustrates this procedure in the case of $L = 6$ for the 16×16 lattice, where the parameters $\{\boldsymbol{\theta}_l\}$ to be trained are explicitly shown.

Now, assume that SIC approximately holds. Then, the renormalized Hamiltonian should be well approximated with NN interactions, i.e., $H_l(\boldsymbol{s}^{\le l}) \approx \widetilde{H}_l(\boldsymbol{s}^{\le l})$ (see Eq.(5)). This means that the sequential target distributions $\{p_{L'}(\boldsymbol{s}^{\le L'}); L' = 0, 2, 4, \ldots, L\}$ have similar marginal distributions on the sites $\boldsymbol{s}^{\le l}$ (for $l \le L'$). Therefore, once the parameters $\{\boldsymbol{\theta}_l\}$ of the $(L'-2)$-layered RiGCS are trained on the corresponding target $p_{L'-2}(\boldsymbol{s}^{\le L'})$, they are good initializations for the corresponding components of the L'-layered RiGCS to be trained on the next target $p_{L'}(\boldsymbol{s}^{\le L'})$. This justifies the model transfer initializations depicted as the vertical red arrows in Figure 2. Furthermore, SIC—stating that the interaction terms in the renormalized Hamiltonians $\{H_l(\boldsymbol{s}^{\le l})\}$ quickly converge to a fixed point for $l < \widetilde{L}$ with some $\widetilde{L} < L$—implies that the renormalized Hamiltonians for different scales, e.g., $H_{l-2}(\boldsymbol{s}^{\le l-2})$ and $H_l(\boldsymbol{s}^{\le l})$, have similar sets of iteraction terms. Therefore, thanks to our choice of using the same architecture for all conditional models over different levels, we can also apply model transfer initializations from $\boldsymbol{\theta}_{l-2}$ to $\boldsymbol{\theta}_l$, as depicted as the slanting red arrows in Figure 2.

Summarizing, our sequential training with model transfer performs the following procedure:

1. Train the (unconditional) generative model $q_{\boldsymbol{\theta}_0}(\boldsymbol{s}_0)$ to approximate $\widetilde{p}(\boldsymbol{s}^0) \propto e^{-\beta \widetilde{H}_0(\boldsymbol{s}^0)}$ with $\boldsymbol{J}_0^{\mathrm{NN}}$. Set $\widetilde{\boldsymbol{\theta}}_0 \leftarrow \boldsymbol{\theta}_0$.
2. Refine $\boldsymbol{\theta}_{\le 1}$ from its initial value $\widetilde{\boldsymbol{\theta}}_{\le 1} = (\widetilde{\boldsymbol{\theta}}_1, \widetilde{\boldsymbol{\theta}}_0)$, where $\widetilde{\boldsymbol{\theta}}_1$ is set randomly, by training $q_{\boldsymbol{\theta}_{\le 1}}(\boldsymbol{s}_{\le 2}) = p(\boldsymbol{s}^2|\boldsymbol{s}^{\le 1}) q_{\boldsymbol{\theta}_1}(\boldsymbol{s}^1|\boldsymbol{s}^0) q_{\boldsymbol{\theta}_0}(\boldsymbol{s}^0)$ to approximate $\widetilde{p}(\boldsymbol{s}^{\le 2}) \propto e^{-\beta \widetilde{H}_2(\boldsymbol{s}^{\le 2})}$. Set $\widetilde{\boldsymbol{\theta}}_{\le 1} \leftarrow \boldsymbol{\theta}_{\le 1}$.
3. Refine $\boldsymbol{\theta}_{\le 3}$ from its initial value $\widetilde{\boldsymbol{\theta}}_{\le 3} = (\widetilde{\boldsymbol{\theta}}_1, \widetilde{\boldsymbol{\theta}}_2, \widetilde{\boldsymbol{\theta}}_1, \widetilde{\boldsymbol{\theta}}_0)$, where $\widetilde{\boldsymbol{\theta}}_2$ is set randomly, by training $q_{\boldsymbol{\theta}_{\le 3}}(\boldsymbol{s}_{\le 4}) = p(\boldsymbol{s}^4|\boldsymbol{s}^{\le 3}) \left(\prod_{l'=1}^{3} q_{\boldsymbol{\theta}_{l'}}(\boldsymbol{s}^{l'}|\boldsymbol{s}^{l'-1}) \right) q_{\boldsymbol{\theta}_0}(\boldsymbol{s}^0)$ to approximate $\widetilde{p}(\boldsymbol{s}^{\le 4}) \propto e^{-\beta \widetilde{H}_4(\boldsymbol{s}^{\le 4})}$. Set $\widetilde{\boldsymbol{\theta}}_{\le 3} \leftarrow \boldsymbol{\theta}_{\le 3}$.
4. For $L' = 6, 8, \ldots, L$, refine $\boldsymbol{\theta}_{\le L'-1}$ from its initial value $\widetilde{\boldsymbol{\theta}}_{\le L'-1} = (\widetilde{\boldsymbol{\theta}}_{L'-3}, \widetilde{\boldsymbol{\theta}}_{L'-4}, \widetilde{\boldsymbol{\theta}}_{\le L'-3})$ by training $q_{\boldsymbol{\theta}_{\le L'-1}}(\boldsymbol{s}_{\le L'}) = p(\boldsymbol{s}^{L'}|\boldsymbol{s}^{\le L'-1}) \left(\prod_{l=1}^{L'-1} q_{\boldsymbol{\theta}_l}(\boldsymbol{s}^l|\boldsymbol{s}^{\le l-1}) \right) q_{\boldsymbol{\theta}_0}(\boldsymbol{s}^0)$ to approximate $\widetilde{p}(\boldsymbol{s}^{\le L'}) \propto e^{-\beta \widetilde{H}_{L'}(\boldsymbol{s}^{\le L'})}$. Set $\widetilde{\boldsymbol{\theta}}_{\le L'-1} \leftarrow \boldsymbol{\theta}_{\le L'-1}$.

Note that all parameters except $\boldsymbol{\theta}_0, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2$—which are trained on the three smallest lattice sizes with random initializations—can be initialized to the parameters trained on easier (smaller lattice) problems, which significantly accelerates the training process, as shown in Figure 3 (right). For large L and $l \ll L$, the approximate renormalized Hamiltonian $\widetilde{H}_l(\boldsymbol{s}^{\le l})$ with NN interactions might be significantly different from the true renormalized Hamiltonian $H_l(\boldsymbol{s}^{\le l})$ that may have long-range and higher-order interaction terms. With our training procedure, this gap is reduced step by step by fine-tuning the generative models with receptive fields beyond the nearest neighbors.

Algorithm 1 RiGCS training

```
Input:
    - Coarsest lattice size $N_0$
    - PixelCNN $q_{\theta_0}$
    - numbers of levels $L$
    - Conditional networks $\{q_{\boldsymbol{\theta}_l}(\boldsymbol{s}^l|\boldsymbol{s}^{\leq l-1})\}$
    - HB algorithm $p(\boldsymbol{s}^l|\boldsymbol{s}^{l-1})$
Output:
    - Trained RiGCS (PixelCNN-based generative model) for sampling $\boldsymbol{s}^L \in \mathcal{S}^D$ with $D = 2^L N_0 \times 2^L N_0$

Train the PixelCNN $q_{\theta_0}$ on $N_0 \times N_0$ target lattices.
Add to the RiGCS's list of models the conditional VAN $q_{\boldsymbol{\theta}_1}(\boldsymbol{s}^1|\boldsymbol{s}^{\leq 0})$ (randomly initialized) and the HB $p(\boldsymbol{s}^2|\boldsymbol{s}^{\leq 1})$.
Train the RiGCS on $2N_0 \times 2N_0$ lattices.
Replace the HB $p(\boldsymbol{s}^2|\boldsymbol{s}^{\leq 1})$ with the conditional VAN $q_{\boldsymbol{\theta}_2}(\boldsymbol{s}^2|\boldsymbol{s}^{\leq 1})$ randomly initialized.
Add to the RiGCS's list of models the conditional VAN $q_{\boldsymbol{\theta}_3}(\boldsymbol{s}^3|\boldsymbol{s}^{\leq 2})$ initialized with the weights of the trained model $q_{\tilde{\boldsymbol{\theta}}_1}$, and the HB $p(\boldsymbol{s}^4|\boldsymbol{s}^{\leq 3})$.
Train the RiGCS on $4N_0 \times 4N_0$ lattices.
for $l = 5, l < L, l = l + 2$ do
    Replace the HB $p(\boldsymbol{s}^{l-1}|\boldsymbol{s}^{\leq l-2})$ with the conditional VAN $q_{\boldsymbol{\theta}_{l-1}}(\boldsymbol{s}^{l-1}|\boldsymbol{s}^{\leq l-2})$ initialized with the trained model $q_{\tilde{\boldsymbol{\theta}}_{l-3}}$ weights.
    Add to the RiGCS's list of models the conditional VAN $q_{\boldsymbol{\theta}_l}(\boldsymbol{s}^l|s^{\leq l-1})$, initialized with the trained model $q_{\tilde{\boldsymbol{\theta}}_{l-2}}$ weights, and HB $p(\boldsymbol{s}^{l+1}|\boldsymbol{s}^{\leq l})$.
    Train the RiGCS on $2^{l+1}N_0 \times 2^{l+1}N_0$ lattices.
end for
```

Algorithm 2 RiGCS sampling

```
Input:
    - Coarsest lattice size $N_0$
    - PixelCNN $q_{\theta_0}$
    - List of conditional models $\{q_{\boldsymbol{\theta}_l}(\boldsymbol{s}^l|\boldsymbol{s}^{\leq l-1})\}_{l=1}^{L-1}$
    - Heatbath algorithm for sampling the finest level $L$: $p(\boldsymbol{s}^L|\boldsymbol{s}^{L-1})$.
Output:
    - Samples $\boldsymbol{s}^L \in \mathcal{S}^D$ with $D = 2^L N_0 \times 2^L N_0$
    - Exact sampling probability $\ln q_{\boldsymbol{\theta}}(s^L)$.

Sample $\boldsymbol{s}^0 \sim q_{\boldsymbol{\theta}_0}$ and compute $\ln q_{\boldsymbol{\theta}_0}(\boldsymbol{s}_0)$.
for $l = 1, l < L, l = l + 2$ do
    Embed the sample $\boldsymbol{s}^{l-1}$ into a $2N_{l-1} \times 2N_{l-1}$ with zeros in the lattice sites of the levels $l, l+1$.
    Sample $\boldsymbol{s}_l \sim q_{\boldsymbol{\theta}_l}(\boldsymbol{s}^l|\boldsymbol{s}^{\leq l-1})$ and compute $\ln q_{\boldsymbol{\theta}_l}(\boldsymbol{s}_l)$.
    if $l + 1 \neq L$ then
        Sample $\boldsymbol{s}_{l+1} \sim q_{\boldsymbol{\theta}_{l+1}}(\boldsymbol{s}^l|\boldsymbol{s}^{\leq l})$ and compute $\ln q_{\boldsymbol{\theta}_{l+1}}(\boldsymbol{s}^{l+1})$.
    else
        Sample $\boldsymbol{s}^L \sim p(\boldsymbol{s}^L|\boldsymbol{s}^{L-1})$ with HB and compute $\ln q_{\boldsymbol{\theta}}(\boldsymbol{s}^L)$.
    end if
end for
```

E.2 PSEUDOCODE FOR RIGCS

The pseudocode provided in Algorithm 1 and Algorithm 2 describes the practical steps for training RiGCS and for sampling from a trained RiGCS.

F AUTOREGRESSIVE NEURAL NETWORKS

Autoregressive neural networks are a class of generative models used to model conditional probability distributions. For high-dimensional configurations, each element in the configuration is predicted based on the previous elements. These models are widely used in time series forecasting (Triebe et al., 2019), natural language processing (van den Oord et al., 2016a), large language models (Brown et al., 2020), and generative modeling (van den Oord et al., 2016b) as they explicitly capture the dependencies between elements in a sequence.

In the last decades, autoregressive neural networks have been extensively deployed in different scientific domains including statistical physics (Wu et al., 2019; Nicoli et al., 2020; Wang et al., 2022; Biazzo, 2023; Biazzo et al., 2024), quantum chemistry (Gebauer et al., 2019; Joshi et al., 2021; Gebauer et al., 2022), learning wave functions of many body systems (Hibat-Allah et al., 2020), tensor newtorks (Chen et al., 2023) and quantum computing (Liu et al., 2021).

Relying on the factorizability of arbitrary distributions as

$$p(\boldsymbol{s}) = \left(\prod_{v=2}^{V} p(s_v \mid s_{v-1}, \ldots, s_1) \right) p(s_1), \tag{31}$$

autoregressive models approximate each factor in the right-hand side with neural network models $q_{\boldsymbol{\theta}}$ with the parameters $\boldsymbol{\theta}$ to be optimized so that $q_{\boldsymbol{\theta}}(\boldsymbol{s}) \approx p(\boldsymbol{s})$. The ancestral sampling in the order of $s_1, \ldots, s_V$ allows sampling and density evaluation at the same time. State-of-the-art architectures often use convolutional neural networks, leveraging masked filters to ensure that the conditional dependencies are restricted to previous elements in the sequence (van den Oord et al., 2016b; Salimans et al., 2017).

G IMPLEMENTATION DETAILS FOR VAN, HAN AND RIGCS

G.1 VAN

The two main architectures used to implement VANs are Masked Autoencoder for Distribution Estimation (MADE) (Germain et al., 2015) and the PixelCNN (van den Oord et al., 2016c;b), which rely respectively on fully-connected and convolutional layers. In order to ensure the autoregressive properties required by the models, the weights of these architectures are masked such that the i-th component of the output $\hat{\boldsymbol{s}}_i = g_\theta(\boldsymbol{s})$ of the network g_θ depends only on the the previous values $\boldsymbol{s}_{<i}$, i.e.,

$$\hat{\boldsymbol{s}}_i = g_\theta(\boldsymbol{s}_{<i}).$$

In the last layer of the network, a sigmoid function is used such that the $\hat{\boldsymbol{s}}_i$ represents the normalized probability of being 1. The conditional density of the new entry is then computed according to a Bernoulli distribution with factor $\hat{\boldsymbol{s}}_i$:

$$\sigma_\theta(\boldsymbol{s}_i|\boldsymbol{s}_{<i}) = \hat{s}_i \delta_{s_i,+1} + (1 - \hat{s}_i)\delta_{s_i,-1}.$$

In our experiments, we used the PixelCNN implemented by Wu et al. (2019) which leverages masked convolutional kernels $k = K \times K$, with K odd and entries $k_{i,j}$ where $i, j = 0, 1, \cdots, K-1$ such that the element $i = (K-1)/2$ and $j = (K-1)/2$ represents the center of the kernel. The mask of the PixelCNN layers fixes to 0 the elements $k_{(K-1)/2, j>(K-1)/2}$ and $k_{i>(K-1)/2, j}$ in order to ensure the autoregressive properties of the network. In the case that the PixelCNN has more than two layers, it makes use of residual connection (He et al., 2015) (i.e. the input to the layer is summed with the output) for each layer, excluding the first and the last. Before each masked convolutional layer of the residual connections, and at the end of the network (before the sigmoid function), is added a standard convolutional layer with a kernel size of 1.

The PixelCNN used for pure VAN simulations has 6 masked convolutional layers with 32 channels and half kernel size $(K-1)/2 = 6$

G.2 HAN

The HAN (Białas et al., 2022) model leverages recursive domain decomposition (Cè et al., 2016) in order to sample in parallel different regions of the lattice configurations. The crucial aspect of the domain decomposition is that the domains must be connected through a common boundary, and, once it is given, each domain can be sampled independently using the same model. In the HAN implementation, a boundary $\mathcal{B}_0$ that divides the lattice in four domains is first sampled using a standard MADE architecture. Then, each domain is split into four by sampling in parallel four boundaries $\mathcal{B}_i$ using another MADE model conditioned on the boundary $\mathcal{B}_0$. This procedure is repeated until the remaining entries have all the neighbours fixed and therefore can be generated using HB.

In our experiments, we used the code and hyperparameters provided by the authors of Białas et al. (2022).

G.3 RIGCS

In our implementation of the RiGCS, at the coarsest level $l = 0$, corresponding to the (unconditional) generative model $q_{\boldsymbol{\theta}_0}(\boldsymbol{s}_0)$, we use a PixelCNN made of 3 masked convolutional layer of 12 kernels with half kernel size equal to 6.

For the implementation of the conditional models $\{q_{\boldsymbol{\theta}_l}(\boldsymbol{s}^l|\boldsymbol{s}^{\leq l-1})\}_{l=1}^{L-1}$ we defined one "block" as two consecutive levels, that correspond to upsampling from lattice size $N_l \times N_l$ to $N_{l+2} \times N_{l+2} = 2N_l \times 2N_l$ with $l \in 0, 2, 4, 6, ...$ Each block takes as input a coarse configuration $\boldsymbol{s}^l$ of size $N_l \times N_l$ and embeds it into a $2N_l \times 2N_l$ configuration where all the entries for the levels $l+1$ and $l+2$ are fixed to 0. Afterwards, the spins of levels $l+1$ and $l+2$ are sampled sequentially according to the output of a standard CNN that takes as input the embedded configuration. Each conditional CNN (conditional VAN) of the RiGCS has one hidden layer with 12 kernels and kernel sizes of 5 and 3 for the hidden and output layers, respectively. We use the same CNN architecture for both levels as well as for all blocks except the last. With this kind of conditional network, the autoregressive properties of the model are ensured and the receptive field is set as explained in 3.2. In the last block of the model, the level $l = L$ is sampled using the HB algorithms due to the local nature of the target Hamiltonian.

As for the PixelCNN, the conditional VAN uses a sigmoid activation in the final layer; thus, the conditional density of the new entry is computed according to a Bernoulli distribution with factor $\hat{\boldsymbol{s}}_i^l$:

$$\sigma_\theta(s_i^l|\boldsymbol{s}_{<i}^l, \boldsymbol{s}^{<l}) = \hat{s}_i^l \delta_{s_i^l,+1} + (1 - \hat{s}_i^l)\delta_{s_i^l,-1}.$$

During the training procedure described in 3.3, each block is initialized with the weights of the conditional VAN of the previous block.

Observe that the architectures used for the conditional VAN are reminiscent of the Multi-Scale PixelCNN introduced in van den Oord et al. (2016c).

G.4 TRAINING

All the generative neural samplers (RiGCS, VAN, HAN) used in our experiments are trained by minimizing the reverse Kullback-Leibler (KL) divergence:

$$\min_{\boldsymbol{\theta}} \mathrm{KL}(q_{\boldsymbol{\theta}}(\boldsymbol{s}) \| p(\boldsymbol{s})) \tag{32}$$

with the gradient estimator:

$$\nabla_\theta \mathrm{KL}(q_{\boldsymbol{\theta}}(\boldsymbol{s}) \| p(\boldsymbol{s})) = \mathbb{E}_{\boldsymbol{s} \sim q_\theta}\left[\left(\beta H(s) + \log q_\theta(\boldsymbol{s})\right)\nabla_\theta \log q_\theta(\boldsymbol{s})\right].$$

In order to make the variance of the estimator more stable, we leverage a control variates method (Mnih & Gregor, 2014) as suggested in Wu et al. (2019). We use the ADAM optimizer (Kingma & Ba, 2014) with learning rate 0.001 and standard βs for training all models.

We trained VANs for 50000 gradient updates (epochs) with batch size 100, and HANs for 100000 gradient updates with batch size 1000, respectively. For RiGCS, training is performed for a total of 3000 epochs for each sequential target lattice volume. When training on a target lattice $N_L = N$, the pretraining phase involves coarser levels, with the following number of epochs: 2000 epochs for level $L-2$, 1500 epochs for level $L-4$, and 1000 epochs for all subsequent levels, except for the coarsest level, which is always trained for 500 epochs.

All models are trained and evaluated on one NVIDIA A100 with 80 GB.

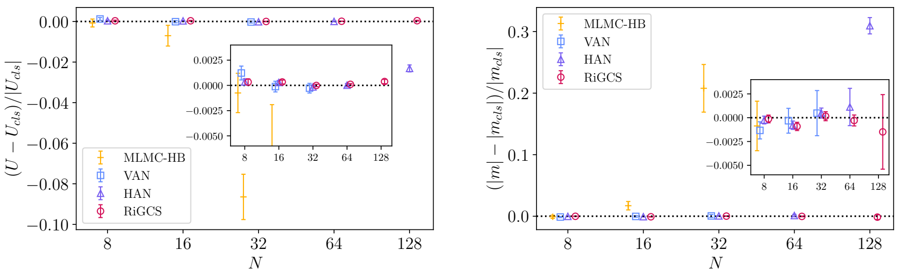

Figure 6: Relative estimation error for the internal energy (left) and absolute magnetization (right). The vanilla VAN cannot be trained for $N \geq 64$ in reasonable time. Note that, unlike in Figure 3 right, we used the MC estimators U_{cls}, m_{cls} by the cluster method as the reference values.

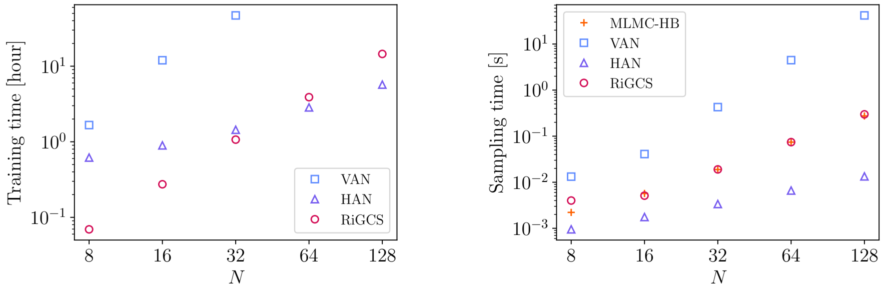

Figure 7: Total training time (left) and sampling time (right) for different lattice sizes.

G.5 SAMPLING

We evaluate the MC esimtates by Cluster(-AIS), HAN and RiGCS with one million samples, and those by VAN and MLMC-HB with $100k$ samples. In the case of Cluster-AIS, for target volume 128×128, we sampled only $500k$ due to the computational costs. Errors are computed using an automatic differentiation method introduced in Ramos (2019) and implemented by Joswig et al. (2023).

H ADDITIONAL NUMERICAL RESULTS

Figure 6 shows additional numerical results of estimating the internal energy and the magnetization, where the estimators by the cluster method are used as the reference values. Consistently with the EMDM and ESS shown in Figure 4, our RiGCS provides unbiased estimates with lowest variances compared to other generative neural samplers. Figure 7 shows empirical training (left) time and sampling (right) time.