

Neural Discrete Controlled Monte Carlo Samplers

Anonymous Authors¹

Abstract

Sampling from an unnormalized probability mass function on a large discrete state space is a core problem in statistical physics and probabilistic inference, yet most learning-based samplers are designed for continuous diffusions. We introduce the *Discrete Controlled Monte Carlo* (DCMC) sampler, a neural sampler based on controlled continuous-time Markov chains (CTMCs). DCMC transports samples from an easy prior to an intractable target by following a prescribed curve of intermediate time marginals and learning a control over CTMC jump rates. DCMC is trained with a physics-informed neural network (PINN) objective that upper bounds the log-variance divergence of the forward–backward path likelihood ratio and yields importance weights for estimating expectations and normalizing constants. We study two complementary control parameterizations: an *additive* control that extends PINN-based generator learning, and a *multiplicative* (exponential-tilt) control that satisfies a discrete time-reversal relation. For the multiplicative case, we show that the optimal control solves an infinitesimal Schrödinger bridge problem for CTMCs with the prescribed marginals. Experiments on Ising and Potts models demonstrate substantial gains of DCMC over prior PINN methods and competitive or better performance compared to recent masked diffusion samplers.

1. Introduction

We study sampling from an unnormalized density γ on a *discrete* state space Ω . The goal is to (i) estimate the normalizing constant $Z = \sum_{x \in \Omega} \gamma(x)$ and (ii) generate samples from the target distribution $\nu(x) = \gamma(x)/Z$. A common

¹Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

instance is the Boltzmann form $\gamma(x) = \exp(-\beta U(x))$, where U is an energy function and β is an inverse temperature. Throughout, we assume oracle access to pointwise evaluations of $\gamma(x)$, while Z and direct sampling from ν are intractable for realistic systems.

Classical Markov chain Monte Carlo (MCMC) methods such as Metropolis–Hastings, Glauber dynamics, and cluster updates (e.g., Swendsen–Wang) are widely used for discrete sampling (Roberts & Tweedie, 1996; Glauber, 1963; Swendsen & Wang, 1987). When combined with annealing or sequential Monte Carlo (SMC), they can also be used to estimate Z (Neal, 1998; Del Moral et al., 2006). Other works also propose Langevin-like dynamics to mimic gradient-based updates of continuous space Langevin Monte Carlo (Zhang et al., 2022; Sun et al., 2023). These methods provide principled sampling procedures with asymptotic correctness. However, local-update chains can mix slowly in high dimensions and near phase transitions, motivating amortized sampling approaches that use learning to construct fast proposal mechanisms and low-variance estimators.

Recent work has begun to import learned transport and control perspectives to discrete controlled continuous-time Markov chain (CTMC) samplers, e.g. via physics-informed neural network (PINN) objectives (Holderrieth et al., 2025) and via masked diffusions (Zhu et al., 2025). However, existing approaches do not simultaneously provide (i) energy-based preconditioning dynamics with analytically tractable local score/ratio terms and (ii) an explicit characterization of the optimal learned dynamics in terms of Schrödinger-bridge optimality along a prescribed marginal flow.

We address this gap by proposing the *Discrete Controlled Monte Carlo* (DCMC) sampler. DCMC defines a controlled CTMC that transports samples from a tractable prior μ to a target ν by following a prescribed curve of intermediate marginals $(\pi_t)_{t \in [0, T]}$. We choose π_t via geometric interpolation, enabling evaluation of the key density ratios without knowing Z . These ratios define a simple, physically motivated preconditioning CTMC, instantiated either by discrete Langevin rates or by a bounded Glauber variant to improve numerical stability. We consider two control parameterizations—*additive* and *multiplicative*—and show that the optimal multiplicative control solves an infinitesi-

055
056
057
058
059
060
061
062
063
064
065
066
067
068
069
070
071
072
073
074
075
076
077
078
079
080
081
082
083
084
085
086
087
088
089
090
091
092
093
094
095
096
097
098
099
100
101
102
103
104
105
106
107
108
109

mal Schrödinger bridge problem for CTMCs with prescribed marginals.

We summarize our contributions as follows:

- We introduce DCMC, a controlled-CTMC neural sampler for discrete unnormalized targets that combines tractable discrete Langevin or bounded Glauber preconditioning dynamics with a PINN-learned control.
- We study two complementary control parameterizations: an additive generator correction and a multiplicative exponential tilt. For the multiplicative form, we establish a formal connection to infinitesimal Schrödinger bridge optimality on discrete path space.
- We empirically validate DCMC on Ising and Potts models across temperatures, improving over prior PINN-based CTMC samplers and achieving competitive or better performance relative to masked diffusion samplers in several temperature regimes.

2. Preliminaries

Let Ω be a finite state space. We write $\rho_t(x) = \mathbb{P}(X_t = x)$ for the marginal pmf of a process $(X_t)_{t \in [0,1]}$ and view ρ_t as a column vector in $\mathbb{R}^{|\Omega|}$. We use the convention that for $x \neq y$, the generator entry $Q_t(y, x)$ is the jump rate $x \rightarrow y$. All proofs are deferred to Appendix B.

2.1. Continuous-time Markov Chains (CTMC)

To describe the stochastic process in discrete state space Ω , a common approach is to adopt a CTMC, which enables theoretical analysis of the dynamics of discrete diffusion models (Austin et al., 2021; Campbell et al., 2022). Every CTMC process is uniquely characterized by an infinitesimal generator Q_t :

$$Q_t = \lim_{\Delta t \rightarrow 0} \frac{\mathbf{P}(t, t + \Delta t) - \mathbf{I}}{\Delta t} \quad (1)$$

where the column-wise sum is zero (i.e. $Q_t(x, x) = -\sum_{y \neq x} Q_t(y, x)$) and

$$\mathbb{P}(X_{t+\Delta t} = y \mid X_t = x) = \mathbf{1}_{y=x} + \Delta t Q_t(y, x) + o(\Delta t). \quad (2)$$

Also, the marginal law evolves according to the Kolmogorov Forward Equation (KFE)

$$\frac{d\rho_t}{dt} = Q_t \rho_t, \quad \rho_0 = \mu. \quad (3)$$

For the path $\{X_t\}_{t \geq 0}$ taken by above CTMC process, we denote the path measure by $\mathbb{P}^{\mu, Q}$ to describe the distribution over the path space. It is possible to define the time-reversed

CTMC process at equation (3) using the concrete score (Meng et al., 2022)

$$Q'(y, x) = \frac{\rho_t(y)}{\rho_t(x)} Q(x, y), \quad (4)$$

which is the discrete counterpart of the time-reversal in (Nelson, 1967; Anderson, 1982; Song et al., 2021; Lou et al., 2024). In practice, concrete score directly models the ratio by a score-matching network (Meng et al., 2022; Lou et al., 2024).

More generally, many objectives and estimators in non-equilibrium sampling can be written using likelihood ratios between path measures. Let $\overleftarrow{\mathbb{P}}^{\nu, Q'}$ denote a reverse-time CTMC initialized with ν at time t , and $\overrightarrow{\mathbb{P}}^{\mu, Q}$ a forward-time CTMC initialized with μ at time 0. When the two path measures are mutually absolutely continuous on $[0, t]$, their Radon–Nikodym derivative (Holderrieth et al., 2025) admits a closed-form expression:

$$\begin{aligned} \log \frac{d\overleftarrow{\mathbb{P}}^{\nu, Q'}}{d\overrightarrow{\mathbb{P}}^{\mu, Q}}(X_{0:t}) &= \log \frac{\nu(X_t)}{\mu(X_0)} \\ &\quad + \int_0^t (\overleftarrow{Q}_s(X_s, X_s) - \overrightarrow{Q}_s(X_s, X_s)) ds \\ &\quad + \sum_{s: X_{s-} \neq X_s} \log \frac{\overleftarrow{Q}_s(X_{s-}, X_s)}{\overrightarrow{Q}_s(X_s, X_{s-})}. \end{aligned} \quad (5)$$

Equation (5) yields importance weights for expectations under ν , and motivates variance-minimizing control of \overleftarrow{Q} ; the choice $Q' = \overleftarrow{Q}$ corresponds to exact time reversal.

2.2. Locally Balanced CTMC Baselines (Discrete Langevin)

To construct a base generator that leaves a target pmf π invariant, we will use a locally balanced rate family. Let $w(x, y) = w(y, x) \geq 0$ be symmetric edge weights, and let $g : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ satisfy the local balance condition $g(r) = r g(1/r)$. Define for $x \neq y$,

$$\begin{aligned} Q^{\text{LB}}(y, x) &= w(x, y) g\left(\frac{\pi(y)}{\pi(x)}\right), \\ Q^{\text{LB}}(x, x) &= -\sum_{y \neq x} Q^{\text{LB}}(y, x). \end{aligned} \quad (6)$$

This defines a valid generator and can be viewed as a discrete analogue of Langevin-type dynamics (Sun et al., 2023).

Here, we consider the Discrete Langevin Dynamics (DLD) following (Sun et al., 2023). DLD follows the perspective of Langevin Dynamics as a Wasserstein Gradient Flow (WGF) over the KL divergence measure to develop a Langevin Dynamic itself in a discrete domain. For inherent variable

110 satisfying $w_{ji} = w_{ij}$ and a locally balanced function $g : \mathbb{R} \rightarrow \mathbb{R}$, the DLMC is given by

$$\frac{d\rho^t}{dt} = \rho^t Q^{\text{LB}} = \left(\sum_i \rho_i^t Q_{ij}^{\text{LB}} \right)_{j=1}^M \quad (7)$$

116 In Section 3 we will parameterize and control the base
117 rates in Eq. (6) to accelerate mixing while preserving the
118 generator constraints in Eq. (1).

120 3. Methods

122 In the context of sampling problem, the time-reversal problem
123 that we aim to solve can be defined to finding the optimal control such that 1) the backward path is the time
124 reversal of the forward path as defined in Eq. (4) and 2)
125 minimize the divergence measure between the forward and
126 backward path likelihood.

128 CTMC that transports samples from an easy prior μ to ν , and
129 uses a path-space likelihood ratio (5) to produce importance
130 weights for estimating normalizing constants.

132 Concretely, DCMC follows the pipeline:

- 134 1. choose a prescribed curve of intermediate marginals
135 $(\pi_t)_{t \in [0,1]}$ from μ to ν ;
- 136 2. build a tractable locally-balanced baseline generator
137 Q_t^{LB} (Eq. (6));
- 138 3. introduce a learnable control over rates, either *additive*
139 (DCMC-Add) or *multiplicative* (DCMC-Mult);
- 140 4. train the control by minimizing a PINN residual that
141 upper-bounds the log-variance divergence of the path-
142 space likelihood ratio;
- 143 5. sample by simulating the learned CTMC

144 The choice we use is the geometric interpolation $\pi_t(\cdot) =$
145 $\pi_0(\cdot)^{\alpha(t)} \pi_T(\cdot)^{1-\alpha(t)}$ which has been used in (Vargas et al.,
146 2024). By the geometric interpolation of the transport, we
147 are able to evaluate the score as the ratio between the data
148 that is necessary to simulate in efficient way. Based on the
149 prior given as an easy-to-sample distribution (e.g. uniform)
150 concrete score term can be naturally evaluated by

$$\begin{aligned} \frac{\pi_t(y)}{\pi_t(x)} &= \frac{\pi_0(y)^{\alpha(t)} \pi_T(y)^{1-\alpha(t)}}{\pi_0(x)^{\alpha(t)} \pi_T(x)^{1-\alpha(t)}} \\ &= \frac{\mu(y)^{\alpha(t)} \nu(y)^{1-\alpha(t)}}{\mu(x)^{\alpha(t)} \nu(x)^{1-\alpha(t)}} \\ &= \frac{\mu(y)^{\alpha(t)} \gamma(y)^{1-\alpha(t)}}{\mu(x)^{\alpha(t)} \gamma(x)^{1-\alpha(t)}} = \frac{\gamma(y)^{1-\alpha(t)}}{\gamma(x)^{1-\alpha(t)}} \end{aligned} \quad (8)$$

159 as the normalizing constants cancel out. We use a linear
160 schedule $\alpha(t) = t$. We may consider this as the concrete
161 score across the predefined marginals. Previous works such
162 as LEAPS or MDNS (Holderrieth et al., 2025; Zhu et al.,
163 2025) do not define the prescribed to constrain the path such

164 that the score term is often unavailable or requires training.
165 In contrary, our method allows simpler computation by the
166 concrete score based on the Discrete WGF in (Sun et al.,
167 2024). Based on above DLMC, we formulate the forward
168 transition to be dependent on Discrete Langevin Dynamics
169 by letting the locally balanced function $g(x) = \sqrt{x}$.

In this section, we seek for the choice of the generator matrix parameterization that is capable of recovering the Controlled Monte Carlo Diffusion Sampler such that it learns a neural network parameterized control that recovers the DLD when control is an identity and the score term is constant in $t \in [0, T]$. Therefore, we consider two potential control methods, namely an additive control and a multiplicative control. The additive control is motivated by the previous work which constructs the Physics-Informed Neural Network (PINN) loss and minimizes a partial differential equation residual with the equivalent decomposition of MCMC generator and neural network parameterized generator matrix, while the multiplicative control is motivated by (Vargas et al., 2024) which guarantees the Schrödinger Bridge optimality. Constructing the forward and backward transitions analogous to Controlled Monte Carlo Diffusion (Vargas et al., 2024) naturally follows in the additive case, as it simply involves adding a parameterized control term to the Langevin rates. In contrast, deriving the corresponding formulation for the multiplicative control is non-trivial. We show how such a control can be parameterized differently and state their mathematical implication by such a construction. All proofs are deferred to Appendix B.

3.1. Additive Control Sampler (DCMC-Add)

We first consider an additive control framework that adds a learnable generator matrix as a control.

$$\begin{aligned} Q^{\theta'}(x, y) &= Q_t^{\text{LB}} + Q_t^\theta \\ &= Q_t^{\text{Uniform}} \sqrt{\frac{\pi_t(y)}{\pi_t(x)}} + Q_t^\theta \quad (\text{DLD}) \end{aligned} \quad (9)$$

This formulation of the Langevin transition rate matrix does not necessarily satisfy the time reversal equation referred in equation 4. However, we can view the DCMC-Add as an extension of the LEAPS (Holderrieth et al., 2025) which considers an additive control based on the Discrete Langevin preconditioning as the Markov-chain Monte Carlo generator (He et al., 2025). More specifically, with the formulation, we are able to achieve the formulation discussed in LEAPS as the forward and backward base transition matrix satisfy the detailed balance condition.

The path-likelihood between the forward and backward path measures and the corresponding PINN loss in additive control can be obtained from equation 5. For the CTMC process with forward additive control in equation 9 the log variance

of path likelihood can be upper bounded by the PINN loss defined by

$$\mathcal{L}(\theta, \phi; t) = \mathbb{E}_{s \sim \text{Unif}_{[0,t]}, x_s \sim Q_s} [|K_s^\theta \rho_s(x_s) - \partial_s F_s^\phi|^2] \quad (10)$$

where the net flux operator K_t^θ is defined by

$$K_t \rho_t(x) = -\partial_t U_t(x) - \sum_{y \in \Omega} Q_t(x, y) \frac{\rho_t(y)}{\rho_t(x)}. \quad (11)$$

The invariance of the PINN loss by the additive control can be shown by the locally balanced condition satisfied by the Langevin generator matrix which is given as the preconditioning method.

By adopting the Langevin preconditioning with the optimal control generator learned by the PINN loss, we design the transition from the sample from prior to the target such that the generated samples tend to converge quickly and precisely to the target distribution with the guidance by the predetermined Langevin path. In continuous case, the Langevin preconditioning is available only when the explicit score term with the log gradient density $\nabla \log \gamma$ is provided. However, the gradient over the target density is undefined in the discrete state space which often is addressed by the RE-INFORCE trick or Gumbel-Softmax relaxation (Williams, 1992; Jang et al., 2016). A previous preconditioning method is to use a softmax normalization based on the conditional distribution of the remaining tokens, which is computationally intensive due to the evaluation on the full conditional distribution (Zhu et al., 2025). As an alternative, it is possible to use the perspective of the log gradient as the score and replace the log gradient term by the ratio between distributions using concrete score (Meng et al., 2022; Lou et al., 2024). This provides a relaxed constraint for the access of the log density due to the availability of the score term. Furthermore, our Langevin preconditioning method that relies on merely an evaluation of the energy function and computing the ratio can be obtained efficiently with single access to the target density based on equation 8.

3.2. Multiplicative Control Sampler (DCMC-Mult)

The multiplicative control adopts the interpretation of generator matrix Q as the velocity field across the graph nodes with scaling factor. This multiplicative control method is motivated by the optimal control parameterization which is defined with respect to the uncontrolled forward measure that is adjusted by an appropriate control that is mapped with an exponential.

$$\begin{aligned} Q^\theta(x, y) &= Q_t^{\text{LB}}(x, y) \exp\left(\frac{1}{2}(\phi^\theta(y) - \phi^\theta(x))\right) \\ Q^{\theta'}(x, y) &= Q_t^{\text{LB}}(x, y) \exp\left(-\frac{1}{2}(\phi^\theta(y) - \phi^\theta(x))\right) \end{aligned} \quad (12)$$

with $Q_t^\theta(x, x) = -\sum_{y \neq x} Q_t^\theta(x, y)$. The potential between nodes are scaled by the multiplicative exponent parameterized by ϕ^θ under such a formulation. The physical intuition behind such a formulation is that we control the base Langevin generator by exponential tilt which encourages a transition to the state y where y is the state which has higher density. Concurrently, the optimal control perspective justifies such a multiplicative control can be achieved, and in fact becomes the unique optimal solution of the Stochastic Optimal Control (SOC) problem under regularity condition. Furthermore, we also benefit from 12 that the ratio between the forward and backward transition can be given by equation

$$\frac{Q^{\theta'}(x, y)}{Q^\theta(y, x)} = \frac{\pi_t(y)}{\pi_t(x)}. \quad (13)$$

as analogous to Nelson's relation referred in many works which uses the time-reversal formulation of the sampling problem in continuous domain (Vargas et al., 2024; Berner et al., 2024). For such parameterization, we will have the optimality which satisfying the time reversal for the multiplicative control.

In the case of multiplicative control, we shall formulate the corresponding PINN loss with multiplicative control which upper bounds the divergence between between the forward measure and backward path measure by

Proposition 3.1. *For the CTMC process with forward multiplicative control in equation 12, the corresponding net flux operator is given by*

$$\begin{aligned} K_t^\theta \pi_t(x) &= \partial_t U_t(x) \\ &- 2 \sum_{y \neq x} Q_t^{\text{LB}}(x, y) \sinh\left(\frac{\phi_t^\theta(y) - \phi_t^\theta(x)}{2}\right) \end{aligned} \quad (14)$$

which the PINN loss in equation 10 upper bounds the log variance divergence of the path likelihood.

One key property and benefit that arises from the usage of the multiplicative control is that when optimized, it achieves the Infinitesimal Schrödinger Bridge optimality over the discrete path space which is the result shown in (Vargas et al., 2024). We only state the informal proposition and leave the formal statement of the proposition and the proof in the appendix B.3.

Proposition 3.2 ((Informal) Infinitesimal Schrödinger Bridge optimality of the multiplicative control). *Fix a finite state space Ω and a prescribed curve of pmfs $(\pi_t)_{t \in [0, T]}$ with $\pi_0 = \mu$ and $\pi_T = \nu$. Let Q_t^{Langevin} denote the base Langevin generator used in the multiplicative parameterization, and consider the controlled generator*

$$Q_t^\phi(x, y) = Q_t^{\text{Langevin}}(x, y) \exp\left(\frac{1}{2}(\phi_t(y) - \phi_t(x))\right) \quad (15)$$

i.e. the forward multiplicative control in Equation 12. Let \mathbb{P}^ϕ denote the path measure of the resulting CTMC.

When there exists a pair (ϕ^*, F^*) such that the PINN residual with net flux operator 14 vanishes pointwise, then the induced path measure \mathbb{P}^{ϕ^*} is the solution of the infinitesimal Schrödinger bridge problem.

As an alternative to the Langevin preconditioning, we further introduce the Glauber dynamic formulation as an alternative to the Langevin dynamic (Glauber, 1963)

$$\begin{aligned} Q^{\text{Glauber}}(x, y) &= Q^{\text{Uniform}}(x, y) \frac{\pi(y)}{\pi(x) + \pi(y)} \\ &= Q^{\text{Uniform}}(x, y) \frac{\frac{\pi(y)}{\pi(x)}}{1 + \frac{\pi(y)}{\pi(x)}}. \end{aligned} \quad (16)$$

We naturally see that this satisfies the Nelson's relation 13 and admits locally balanced function $g(x) = \frac{x}{1+x}$. The Glauber dynamic is intended to simulate the Ising model by discrete Master equation with Jump process (spin flip) which recovers the Langevin dynamics at $\Delta t \rightarrow 0$. While the Langevin dynamics with concrete score is unbounded by its exponential form with respect to the potential differences which may incur training instability, the Glauber dynamic is formulated as the sigmoid which has bounded rate which is expected to become more stable in the high-dimensional case which may include high-energy state and potentially drive the concrete score to explode. Fortunately, we naturally see that the Glauber dynamics achieve Nelson's relation in equation 13 while maintaining the PINN loss in 10. Therefore, we are able to merely modify the transition step of the trained model to apply the Glauber dynamics instead of full retraining. We leave the verification of the properties to the appendix. We may view the Glauber dynamics as an alternative way to precondition the neural network by softmax-like approach which does not require an explicit computation over the conditional distribution. We summarize the sampling algorithm 1 where the training is done by minimizing PINN loss using the output roll-out (algorithm provided in 3).

After training, we sample by simulating the learned inhomogeneous CTMC from $X_0 \sim \mu$ to $t = 1$. Along each trajectory we accumulate log weights from the path-space likelihood ratio (Eq. (5)), implemented via the integral of the K -operator. Final weighted samples can be (i) self-normalized to estimate expectations under ν and (ii) optionally resampled (SMC) to obtain approximately unweighted samples (Del Moral et al., 2006).

4. Experiments

Setup We evaluate DCMC on sample quality and normalizing-constant estimation for classical lattice energy-

Algorithm 1 LEAPS Sampling with Controlled generator

Require: N time steps, M walkers, model G_t^θ , MCMC kernel \mathcal{M}_t , density ρ_t , coeff. $\epsilon_t \geq 0$, resample thres. $0 \leq \delta \leq 1$
Init: $X_0^m \sim \rho_0, A_0^m = 0 \quad (m = 1, \dots, M)$
Set $h = 1/N$
for $n = 0$ to $N - 1$ **do**
 for $m = 1$ to M **do**
 $X_t^m \sim \mathcal{M}_t(\cdot, X_t^m)$ with prob. $h\epsilon_t$ else X_t^m
 Adjust the generator by controlling $Q_t^\theta(y, X_t)$
 (follow equation 9 or 12 for control)
 $X_{t+h}^m \sim (\mathbf{1}_{X_t=y} + hQ_t^\theta(y, X_t))_{y \in S}$
 $A_{t+h}^m = A_t^m + h\mathcal{K}_t^\theta \rho_t(X_t^m)$
 end for
 $t \leftarrow t + h$
end for
output $\{(X_t^m, A_t^m, t)\}_{t,m}$

based models. We consider two families of targets: the binary Ising model and the q -state Potts model on $L \times L$ grids. In all cases, the prior μ is uniform on the discrete configuration space. DCMC uses the geometric bridge $\pi_t \propto \mu^{1-\alpha(t)} \gamma^{\alpha(t)}$ from Section 3 and trains a control using the PINN loss. We use $\alpha(t) = t$ in our experiments. For the locally-balanced baseline, we use a single-site neighborhood graph: $y \in N(x)$ differs from x by a single spin update (spin flip for Ising; single-site recoloring for Potts). This keeps the neighborhood size $\mathcal{O}(L^2)$ (Ising) or $\mathcal{O}(L^2(q-1))$ (Potts), avoiding any dependence on $|\Omega|$.

Baselines. We compare our result against LEAPS (Holdenrieth et al., 2025), a PINN-based CTMC sampler trained via a Kolmogorov-forward residual, and MDNS (Zhu et al., 2025), a masked diffusion sampler for discrete unnormalized targets by adopting the reported numerical values. In the tractable regime (4×4), since $|\Omega| = 2^{16}$ is enumerable, we compute exact distributional divergences between the (reweighted) empirical sampler distribution $\hat{\nu}$ and the ground-truth ν : total variation, $\text{KL}(\hat{\nu} \parallel \nu)$, and $\chi^2(\hat{\nu} \parallel \nu)$, as well as absolute error in $\log \hat{Z}$.

For the evaluation in high dimension distribution (16×16), we have adopted the numeric values and evaluation methods from (Zhu et al., 2025) for fair comparison. However, we reduce the number of samples generated by the model to evaluate on magnaization and correlation coefficient from 2^{20} to 2^{16} which may increase the variance. We denote our samplers DCMC-Mult following Langevin and Glauber dynamics as DCMC-Langevin and DCMC-Glauber respectively. We train each DCMC methods over 150k training iterations which the model seems to plateau. We evaluate the performance using ESS which is a commonly used metric to measure the performance of a neural sampler. We

further evaluate the result by absolute error of magnetization and 2-point correlation. We leave the definition and detailed computation of error metrics at appendix D.

4.1. Ising Model

We first test our trained model on a standard lattice Ising model with varying temperature to validate our performance on a common unnormalized density. Lattice Ising model provides the Hamiltonian of the system given the spins states as the discrete variable $i, j \in \{-1, 1\}$ via the equation

$$H(\sigma) = - \sum_{\langle i, j \rangle} J_{ij} \sigma_i \sigma_j \quad (17)$$

with the interaction hyperparameter J_{ij} . The corresponding density will be given by

$$\nu = \frac{1}{Z} e^{-\beta H(\sigma)} \quad \text{where } Z = \sum_{\sigma \in \Omega} e^{-\beta H(\sigma)}. \quad (18)$$

For the experiment, we fix the hyperparameter $J = 1, h = 0$ and vary temperature (and the respective inverse temperature β) along high $\beta_{\text{high}} = 0.28$, critical $\beta_{\text{critical}} = 0.4407$ and low $\beta_{\text{low}} = 0.6$ where the critical temperature is known to be the most challenging to learn.

We first present our result on 4×4 lattice Ising model as an example target distribution at table 1. For 4×4 model, the total state space is within the tractable range of 2^{16} which allows the direct computation of many related metrics. In such low dimensional case, we are able to generate sufficiently large number of samples (we set 2^{20}) to evaluate the performance of the neural network by obtaining the empirical distribution as an approximation of the ground truth target density. Then, we use the empirical distribution to compute the various divergence measure in function domain to compute the distance between the distribution. We show the result on $\beta_{\text{high}} = 0.28$ and leave the detailed computation method of the metric and result on the other temperatures at appendix D and G.

We show that the Glauber and Langevin dynamics-based DCMC outperformed all MDNS methods in divergence-based metrics, especially with the superior performance of the Glauber dynamics. We see that the KL divergence between the sample distribution and ground truth distribution is extremely low which indicates that our samples have very alike distribution landscape to that of the target. As its performance is superior measured in divergence, this validates our neural sampler has learned the distribution successfully. On the other hand, our absolute error of the log normalizing constant is extremely large compared to that of the MDNS. This phenomenon likely stems from the fact that our PINN loss minimizes an upper bound on the log-variance divergence rather than directly minimizing the divergence.

High-Dimension Scalability We subsequently test our sampling method on high dimensional density for real-life practicality and robustness against the curse of dimension. From table 2, we see that our DCMC outperforms the performance of LEAPS across all temperatures. Furthermore, at high temperature case, the DCMC sampler outperforms all other previous baseline models with extremely high performance both in the ESS and ground truth observable estimation (magnetization and 2-point correlation coefficient). The performance increase is huge especially for magnetization which reduced the error to the factor of 10. One surprising observation is that the performance of DCMC-Add significantly outperforms LEAPS demonstrating the effectiveness of our preconditioning method with minimal computational cost increase.

We have observed that the metric of ESS and absolute error of magnetization and correlation coefficient estimation has varying performance over different temperatures. We have observed that our model has achieved minimal error for low temperature $\beta = 0.6$. For low-temperature case, where the performance measured by the ESS is significantly lower compared to the MDNS, we show that our DCMC sampler outperforms them in other metrics and generates a sample that are more accurately estimating the ground truth observables. On the other hand, we observe that the error becomes much larger in the case of critical temperature $\beta = 0.4407$ which falls behind the baseline especially in the estimating the magnetization coefficient. We consider this as a critical slowing down phenomenon in Ising model which the dynamics become widely known to be challenging at around the critical temperature. While our method is more susceptible and sensitive against critical slowing down, experiment demonstrated the DCMC's superior performance in estimating the magnetization of the ground truth generated samples from DCMC generating the state of the art performance at certain temperature cases in capturing the ground truth observables.

4.2. Potts Model

To justify the performance of DCMC on arbitrary density, we now conduct another experiment on the Potts model which is known to model the interaction between the spin and lattice, extending the Ising model. The Potts model has Hamiltonian defined by

$$H(\sigma) = - \sum_{\langle i, j \rangle} J_{ij} \delta(\sigma_i, \sigma_j) \quad (19)$$

where δ represent the Kronecker delta function computing the interaction. Then, its density will be defined as in 18 which defines the corresponding Boltzmann distribution.

Similar to the Ising model, we test the performance of our DCMC sampler on various temperature to validate its per-

330
 331 *Table 1.* Results for learning 4×4 Ising model with $J = 1$, $h = 0.1$, and $\beta_{\text{high}} = 0.28$, best in **bold**.
 332

| Method | ESS \uparrow | $\text{TV}(\hat{p}_{\text{samp}}, \pi) \downarrow$ | $\text{KL}(\hat{p}_{\text{samp}} \parallel \pi) \downarrow$ | $\chi^2(\hat{p}_{\text{samp}} \parallel \pi) \downarrow$ | Abs. err. of $\log \hat{Z} \downarrow$ |
|-----------------------------|----------------|--|---|--|--|
| DCMC-Langevin | 0.9999 | 0.0727 | 0.0347 | 0.0710 | 0.03537 |
| DCMC-Glauber | 1.0000 | 0.0706 | 0.0344 | 0.0690 | 0.03536 |
| DCMC-Add | 0.9997 | 0.0731 | 0.0354 | 0.0745 | 0.03531 |
| $\mathcal{F}_{\text{RERF}}$ | 0.9621 | 0.0799 | 0.0380 | 0.0845 | 0.00003 |
| \mathcal{F}_{LV} | 0.9713 | 0.0748 | 0.0348 | 0.0714 | 0.00046 |
| \mathcal{F}_{CE} | 0.9513 | 0.0833 | 0.0393 | 0.0903 | 0.00099 |
| $\mathcal{F}_{\text{WDCE}}$ | 0.9644 | 0.0799 | 0.0382 | 0.0868 | 0.00030 |

 341
 342 *Table 2.* Comparison of learning results for the 16×16 Ising model. Mag. and Corr. represent absolute error to ground truth values,
 343 while ESS represents Effective Sample Size. Best results in **bold**.
 344

| Temperature | $\beta_{\text{low}} = 0.6$ | | | $\beta_{\text{critical}} = 0.4407$ | | | $\beta_{\text{high}} = 0.28$ | | |
|----------------------|----------------------------|-------------------|--------------------|------------------------------------|-------------------|--------------------|------------------------------|-------------------|--------------------|
| | Metrics | Mag. \downarrow | Corr. \downarrow | ESS \uparrow | Mag. \downarrow | Corr. \downarrow | ESS \uparrow | Mag. \downarrow | Corr. \downarrow |
| DCMC-Langevin (ours) | 0.9e-3 | 1.6e-3 | 0.468 | 1.6e-2 | 1.4e-2 | 0.741 | 1.5e-3 | 0.9e-3 | 1.000 |
| DCMC-Glauber (ours) | 1.9e-3 | 3.4e-3 | 0.370 | 1.4e-2 | 1.1e-2 | 0.666 | 1.2e-3 | 0.8e-3 | 1.000 |
| DCMC-Add (ours) | 1.7e-3 | 2.9e-3 | 0.390 | 1.3e-2 | 1.1e-2 | 0.670 | 0.8e-3 | 0.7e-3 | 1.000 |
| MDNS | 9.9e-3 | 2.4e-3 | 0.981 | 3.7e-3 | 2.0e-3 | 0.933 | 8.5e-3 | 1.0e-3 | 0.962 |
| LEAPS | 2.4e-2 | 5.8e-1 | 0.261 | 7.4e-3 | 1.6e-1 | 0.384 | 7.4e-3 | 1.6e-3 | 0.987 |

 354 performance. We set the inverse temperature to be along high
 355 $\beta_{\text{high}} = 0.5$, critical $\beta_{\text{critical}} = 1.005$ and low $\beta_{\text{low}} = 1.2$.
 356

 357 Based on table 3, the performance of the DCMC sampler is
 358 extremely high in ESS and well-captures the magnetization
 359 coefficient while MDNS outperforms our approach in cap-
 360 turing the correlation coefficient. We furthermore see that
 361 our Glauber dynamics based sampler outperform in captur-
 362 ing the magnetization of all other methods which justifies
 363 the purpose of Glauber. The performance of LEAPS has
 364 been successful in the case of high temperature where we
 365 push the boundary to perfect ESS of 1.000.

 366 In contrary, we see a significant decrease in the performance
 367 in other temperatures concurrently with LEAPS. Especially,
 368 the performance on low temperature measured in ESS is
 369 extremely low, indicating the network fails to learn the im-
 370 portance weight and properly bound the log variance di-
 371 vergence. However, our work outperforms LEAPS at all
 372 metrics and demonstrates a significant improvement in the
 373 ESS achieved by a relatively simple preconditioning method
 374 in critical temperature showing the increased robustness.
 375

5. Related Works

 376 Sampling from an unnormalized density has been studied
 377 extensively in the field of statistics, often referred to as
 378 a gradient-based MCMC algorithm. The most common
 379 approach is to simulate a SDE that has a fixed prior distribu-
 380 tion which is easy to sample such as Gaussian or Dirac-delta
 381

(Rossky et al., 1978; Welling & Teh, 2011). Despite recent advancements in the gradient-based MCMC methods by annealed importance sampling (AIS) or Sequential Monte Carlo (SMC), these methods still face their own problems such as slow mixing rates in high-dimensional multimodal distributions, mode collapse, and the computational cost of simulating long trajectories to ensure convergence (Neal, 1998; Del Moral et al., 2006).

Recent works develop a neural network learning framework to generate samples through a simulation with a learned neural network. (Berner et al., 2024; Richter & Berner, 2024) formulates the diffusion generative model as an optimal control problem with time reversal. (Shi et al., 2024) constructs an alternative PINN loss that estimates the violation of log-density Fokker-Planck Equation (FPE) across the trajectory and minimizes it to estimate the correct score function. In contrast to the previous works, (Vargas et al., 2024) fixes an geometric interpolation as the base transition of the marginals and learns the corresponding Monte-Carlo control by Girsanov theorem. Furthermore, they show the optimal control achieved by the CMCD solves the infinitesimal Schrodinger Bridge Problem. Our work has the conceptual approach which adopts annealed importance sampling with controlled drift and prescribed marginals, while our work specializes in discrete domain which SDE is unavailable and minimizes the PINN surrogate instead of directly minimizing the path likelihood. Concurrently, the CMCD shows that the optimal MC trajectory solves the infinitesimal Schrodinger Bridge problem (Albergo & Vanden-Eijnden,

385 *Table 3.* Comparison of learning results for the 16×16 Potts model. Mag. and Corr. represent absolute error to ground truth values,
 386 while ESS represents Effective Sample Size. Best results in **bold**.

| Temperature | $\beta_{\text{low}} = 1.2$ | | | $\beta_{\text{critical}} = 1.005$ | | | $\beta_{\text{high}} = 0.5$ | | |
|----------------------|----------------------------|---------------|--------------|-----------------------------------|---------------|--------------|-----------------------------|---------------|--------------|
| | Mag. ↓ | Corr. ↓ | ESS ↑ | Mag. ↓ | Corr. ↓ | ESS ↑ | Mag. ↓ | Corr. ↓ | ESS ↑ |
| DCMC-Langevin (ours) | 5.9e-2 | 6.0e-2 | 0.022 | 1.1e-1 | 9.4e-2 | 0.247 | 1.5e-3 | 1.1e-3 | 1.000 |
| DCMC-Glauber (ours) | 5.4e-2 | 6.7e-2 | 0.019 | 1.4e-1 | 1.2e-1 | 0.174 | 1.3e-3 | 1.0e-3 | 1.000 |
| DCMC-Add (ours) | 1.7e-1 | 2.0e-1 | 0.016 | 1.7e-1 | 1.4e-1 | 0.167 | 1.5e-3 | 1.2e-3 | 1.000 |
| MDNS | 1.3e-3 | 8.8e-5 | 0.933 | 4.3e-3 | 2.9e-3 | 0.875 | 2.2e-3 | 5.8e-4 | 0.983 |
| LEAPS | 2.9e-1 | 2.5e-1 | 0.012 | 2.7e-1 | 2.0e-1 | 0.004 | 2.9e-3 | 1.2e-3 | 0.991 |

2025) uses Jarzynski’s equality on top of AIS and estimates the drift which is used to simulate the coupled FPE and ODE.

In discrete domain, there have been approaches to use gradient MCMC, especially Langevin-based methods, in discrete state space. Unlike continuous domain, the gradient and SDE simulation is unavailable which requires an additional modification to the continuous methods. (Zhang et al., 2022) generalizes the proposal domain of Metropolis Hastings algorithm which enables Langevin proposal in discrete domain. (Sun et al., 2023) adapts the WGF perspective of the Langevin dynamics, and creates its analogue on the probability simplex which they propose the Discrete WGF as a DLD. (Holderrieth et al., 2025) becomes the baseline method to our approach which proposes a PINN-based surrogate loss which models the generator matrix using neural network that minimizes the violation of FPE long the transport of the marginals. While LEAPS parameterizes the generator, our work adopts the control-based method using the generator and more importantly a vector field ϕ that adjusts the generator. We further introduce the preconditioning method with prescribed marginal to improve training. (Zhu et al., 2025) uses a discrete diffusion model and formulates the SOC for CTMC which subsequently uses various losses for efficient training. Our multiplicative control formulation is connected to the optimal control for the CTMC generator while we minimize the surrogate PINN loss that minimizes the log variance divergence of the time reversal process instead of directly minimizing the divergence between the path measures.

6. Discussion and Conclusion

We propose a Discrete Monte Carlo Sampler by investigating different control types with different parameterization to establish a reliable neural sampler. We show the effectiveness of our Langevin preconditioning method by DCMC-Add based on the PINN architecture followed by the demonstrating a strong performance of DCMC-Mult on various target densities such as Ising and Potts model. We propose to have the Glauber dynamics as an alternative

to the Langevin preconditioning due to its stability with saturated flow, derived from its sigmoidal formulation. We further prove the theoretical significance of our approach and connects it to the optimal control and the infinitesimal Schrödinger Bridge problem as discussed in its continuous counterpart work. We find our contribution to be significant in the domain where generation of high-fidelity samples are required such as statistical physics and probabilistic inference.

Limitations Despite the empirical success of our method on real-life distribution such as Ising, our method still faces some challenges. While our method outperforms previous methods in certain experiment configurations, it does not achieve exceeding performance over MDNS on all tasks especially on high temperature in ESS and low temperature for absolute error of the observables. In other words, our work may be an alternative to the MDNS over certain tasks, yet it may not completely replace them, similarly in the critical temperature sampling case. Furthermore, our DCMC requires more training iteration to converge compared to MDNS. Throughout our experiment, we have set the training iteration for 150k which is $3\times$ more compared to that of MDNS except for high temperature case, which demonstrated superior performance in the same training steps.

Future Directions We still believe there are remaining tasks in the theoretical analysis of our sampler and providing optimization that allows high ESS and low estimation error. Especially, we plan to seek for temperature-robust methods to generate reliable samples even at varying temperature based on the lacking performance our method on certain temperatures. Despite our result performed relatively well on low and high temperatures of the lattice ising and, we recognize the lack of quantification of the deviation from the infinitesimal Schrödinger Bridge path that may scale with respect to the PINN loss. We shall further quantify the deviation both empirically and theoretically by establishing the bounds.

References

- 440
441 Albergo, M. S. and Vanden-Eijnden, E. Nets: A non-
442 equilibrium transport sampler, 2025.
443
444 Anderson, B. D. Reverse-time diffusion equation models.
445 *Stochastic Processes and their Applications*, 12(3):313–
446 326, 1982.
447
448 Austin, J., Johnson, D. D., Ho, J., Tarlow, D., and Van
449 Den Berg, R. Structured denoising diffusion models in
450 discrete state-spaces. *Advances in Neural Information
451 Processing Systems*, 34:17981–17993, 2021.
452
453 Berner, J., Richter, L., and Ullrich, K. An optimal con-
454 trol perspective on diffusion-based generative modeling.
455 *Transactions on Machine Learning Research*, 2024.
456
457 Campbell, A., Benton, J., De Bortoli, V., Rainforth, T., Deli-
458 giannidis, G., and Doucet, A. A continuous time frame-
459 work for discrete denoising models. *Advances in Neural
460 Information Processing Systems*, 35:28266–28279, 2022.
461
462 Del Moral, P., Doucet, A., and Jasra, A. Sequential monte
463 carlo samplers. *Journal of the Royal Statistical Society
464 Series B: Statistical Methodology*, 68(3):411–436, 2006.
465
466 Glauber, R. J. Time-dependent statistics of the ising model.
467 *Journal of Mathematical Physics*, 4:294, 1963. doi: 10.
468 1063/1.1703954.
469
470 He, J., Du, Y., Vargas, F., Zhang, D., Padhy, S., OuYang,
471 R., Gomes, C., and Hernández-Lobato, J. M. No trick,
472 no treat: Pursuits and challenges towards simulation-free
473 training of neural samplers, 2025.
474
475 Holderrieth, P., Albergo, M. S., and Jaakkola, T. Leaps: A
476 discrete neural sampler via locally equivariant networks.
477 *arXiv preprint arXiv:2502.10843*, 2025.
478
479 Jang, E., Gu, S., and Poole, B. Categorical repara-
480 rameterization with gumbel-softmax. *arXiv preprint
481 arXiv:1611.01144*, 2016.
482
483 Lou, A., Meng, C., and Ermon, S. Discrete diffusion mod-
484eling by estimating the ratios of the data distribution. *arXiv
485 preprint arXiv:2310.16834*, 2024.
486
487 Meng, C., Choi, K., Song, J., and Ermon, S. Concrete score
488 matching: Generalized score matching for discrete data.
489 *Advances in Neural Information Processing Systems*, 35:
34532–34545, 2022.
490
491 Neal, R. M. Annealed importance sampling, 1998.
492
493 Nelson, E. *Dynamical Theories of Brownian Motion*. Mathe-
494 matical Notes. Princeton University Press, 1967.
495
496 Richter, L. and Berner, J. Improved sampling via learned
497 diffusions. In *International Conference on Learning Rep-
498 resentations*, 2024.
499
500 Roberts, G. O. and Tweedie, R. L. Geometric convergence
501 and central limit theorems for multidimensional hastings
502 and metropolis algorithms. *Biometrika*, 83(1):95–110,
503 1996.
504
505 Rossky, P. J., Doll, J. D., and Friedman, H. L. Brownian
506 dynamics as smart monte carlo simulation. *The Journal
507 of Chemical Physics*, 69(10):4628–4633, 1978.
508
509 Shi, Z., Yu, L., Xie, T., and Zhang, C. Diffusion-pinn
510 sampler, 2024. URL <https://arxiv.org/abs/2410.15336>.
511
512 Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Er-
513 mon, S., and Poole, B. Score-based generative modeling
514 through stochastic differential equations. In *International
515 Conference on Learning Representations*, 2021.
516
517 Sun, H., Dai, H., Dai, B., Zhou, H., and Schuurmans, D.
518 Discrete langevin samplers via wasserstein gradient flow.
519 In *International Conference on Artificial Intelligence and
520 Statistics*, pp. 6290–6313, 2023.
521
522 Sun, J., Berner, J., Richter, L., Zeinhofer, M., Müller, J.,
523 Azizzadenesheli, K., and Anandkumar, A. Dynamical
524 measure transport and neural pde solvers for sampling.
525 *arXiv preprint arXiv:2407.07873*, 2024.
526
527 Swendsen, R. H. and Wang, J.-S. Nonuniversal criti-
528 cal dynamics in monte carlo simulations. *Phys. Rev.
529 Lett.*, 58:86–88, Jan 1987. doi: 10.1103/PhysRevLett.
530 58.86. URL <https://link.aps.org/doi/10.1103/PhysRevLett.58.86>.
531
532 Vargas, F., Padhy, S., Blessing, D., and Nüsken, N. Trans-
533 port meets variational inference: Controlled monte carlo
534 diffusions. In *The Twelfth International Conference on
535 Learning Representations*, 2024.
536
537 Welling, M. and Teh, Y. W. Bayesian learning via stochastic
538 gradient langevin dynamics. In *Proceedings of the 28th
539 international conference on machine learning (ICML-11)*,
540 pp. 681–688, 2011.
541
542 Williams, R. J. Simple statistical gradient-following algo-
543 rithms for connectionist reinforcement learning. *Machine
544 learning*, 8:229–256, 1992.
545
546 Zhang, R., Liu, X., and Liu, Q. A langevin-like sampler
547 for discrete distributions. In *International Conference on
548 Machine Learning*, 2022.
549
550 Zhu, Y., Guo, W., Choi, J., Liu, G.-H., Chen, Y., and Tao, M.
551 MDNS: Masked diffusion neural sampler via stochastic
552 optimal control. In *The Thirty-ninth Annual Conference
553 on Neural Information Processing Systems*, 2025.

A. Notation and Assumptions

Let $\gamma : \Omega \rightarrow \mathbb{R}$ be an unnormalized density (we call it target density in the paper) over an arbitrary measurable space $(\Omega, \mathcal{F}, \lambda)$. The goals of the sampling problem are two fold:

1. Find the normalizing constant Z defined by

$$Z = \sum_{x \in \Omega} \gamma(x)$$

2. Generate samples $\mathbf{x} \sim \pi$ from a target probability distribution (we call it target distribution afterward) defined by

$$\pi = \frac{\gamma}{Z}$$

In the Boltzmann distribution, we consider the target has form $\gamma = \exp(-\beta U(X))$ where β represents the inverse temperature of the system.

B. Proofs

B.1. Proof to Equivalence of the PINN loss for Additive Control

The goal here is that the Langevin term in the forward transition generator satisfy the detailed balance condition

$$Q_t(y, x)\rho_t(x) = Q_t(x, y)\rho_t(y). \quad (20)$$

This can be obtained by expanding the left and right hand side correspondingly such that

$$\begin{aligned} \text{LHS} &= \mathcal{M}_t(y, x) \cdot \pi_t(x) \\ &= \left(Q_t^{\text{Uniform}} \sqrt{\frac{\pi_t(y)}{\pi_t(x)}} \right) \cdot \pi_t(x) \\ &= Q_t^{\text{Uniform}} \cdot \frac{\sqrt{\pi_t(y)}}{\sqrt{\pi_t(x)}} \cdot \sqrt{\pi_t(x)} \cdot \sqrt{\pi_t(x)} \\ &= Q_t^{\text{Uniform}} \sqrt{\pi_t(y)} \sqrt{\pi_t(x)} \end{aligned} \quad (21)$$

and

$$\begin{aligned} \text{RHS} &= \mathcal{M}_t(x, y) \cdot \pi_t(y) \\ &= \left(Q_t^{\text{Uniform}} \sqrt{\frac{\pi_t(x)}{\pi_t(y)}} \right) \cdot \pi_t(y) \\ &= Q_t^{\text{Uniform}} \cdot \frac{\sqrt{\pi_t(x)}}{\sqrt{\pi_t(y)}} \cdot \sqrt{\pi_t(y)} \cdot \sqrt{\pi_t(y)} \\ &= Q_t^{\text{Uniform}} \sqrt{\pi_t(x)} \sqrt{\pi_t(y)}. \end{aligned} \quad (22)$$

LEAPS mentions that the PINN loss in LEAPS also bounds the log variance divergence of the path likelihood under the addition of MCMC generator that satisfies the detailed balance condition.

B.2. Proof to Proposition 3.1

Proposition B.1. For the CTMC process with forward multiplicative control in equation 12, the corresponding Radon-Nikodym Derivative is given by

$$\begin{aligned}
 \log \frac{d\overleftarrow{\mathbb{P}}^{\nu, Q'}}{d\overrightarrow{\mathbb{P}}^{\mu, Q^\theta}} &= \log(\nu(X_t)) - \log(\mu(X_0)) \\
 &\quad + \int_0^T (Q^{\theta'}(X_s, X_s) - Q^\theta(X_s, X_s)) ds + \sum_{s, X_{s-} \neq X_s} \log \left(\frac{Q_s^{\theta'}(X_{s-}, X_s)}{Q_s^\theta(X_s, X_{s-})} \right) \\
 &= \log(\nu(X_t)) - \log(\mu(X_0)) + \int_0^T \left[2 \sum_{y \neq X_s} Q_s^{\text{Uniform}} \sqrt{\frac{\pi_s(y)}{\pi_s(X_s)}} \sinh \left(\frac{\phi^\theta(y) - \phi^\theta(X_s)}{2} \right) \right] ds \\
 &\quad + \sum_{s, X_{s-} \neq X_s} (\log \pi_s(X_s) - \log \pi_s(X_{s-}))
 \end{aligned} \tag{23}$$

Furthermore, the corresponding PINN loss is given by

$$\mathcal{L}(\theta, F; t) = \mathbb{E}_{x \sim \text{Unif}} \left[\partial_t U_t(x) - 2 \underbrace{\sum_{y \neq x} Q_t^{\text{Langevin}}(x, y) \sinh \left(\frac{\phi_t^\theta(y) - \phi_t^\theta(x)}{2} \right)}_{\mathcal{K}_t^\theta \pi_t(x) \text{ (Net Flux)}} - \partial_t F_t \right]^2 \tag{24}$$

which upper bounds the log variance divergence of the path likelihood.

Proof. By definition of Langevin generator, we are able to write the forward and backward generator as

$$\begin{aligned}
 Q^\theta(x, y) &= Q_t^{\text{Uniform}} \sqrt{\frac{\pi_t(y)}{\pi_t(x)}} \exp \left(\frac{1}{2} (\phi^\theta(y) - \phi^\theta(x)) \right) \\
 Q^{\theta'}(x, y) &= Q_t^{\text{Uniform}} \sqrt{\frac{\pi_t(y)}{\pi_t(x)}} \exp \left(-\frac{1}{2} (\phi^\theta(y) - \phi^\theta(x)) \right)
 \end{aligned} \tag{25}$$

which gives the analogue of the Nelson's relation for CTMC

$$\frac{Q_t^{\theta'}(x, y)}{Q_t^\theta(y, x)} = \frac{Q_{\text{uniform}} \sqrt{\frac{\pi_t(y)}{\pi_t(x)}} \exp \left(\frac{1}{2} (\phi_t^\theta(y) - \phi_t^\theta(x)) \right)}{Q_{\text{uniform}} \sqrt{\frac{\pi_t(x)}{\pi_t(y)}} \exp \left(\frac{1}{2} (\phi_t^\theta(y) - \phi_t^\theta(x)) \right)} = \frac{\pi_t(y)}{\pi_t(x)} \tag{26}$$

Subsequently, we shall use the Radon-Nikodym Derivative in equation 5 and simplify the corresponding values.

We first consider the stay term

$$\begin{aligned}
 Q^{\theta'}(x, x) - Q^\theta(x, x) &= \left(- \sum_{y \neq x} Q^{\theta'}(x, y) \right) - \left(- \sum_{y \neq x} Q^\theta(x, y) \right) \\
 &= \sum_{y \neq x} (Q^\theta(x, y) - Q^{\theta'}(x, y)) \\
 &= \sum_{y \neq x} Q_t^{\text{Uniform}} \sqrt{\frac{\pi_t(y)}{\pi_t(x)}} \left[\exp \left(\frac{\phi_t^\theta(y) - \phi_t^\theta(x)}{2} \right) - \exp \left(-\frac{\phi_t^\theta(y) - \phi_t^\theta(x)}{2} \right) \right] \\
 &= 2 \sum_{y \neq x} Q_t^{\text{Uniform}} \sqrt{\frac{\pi_t(y)}{\pi_t(x)}} \sinh \left(\frac{\phi^\theta(y) - \phi^\theta(x)}{2} \right)
 \end{aligned} \tag{27}$$

605 using the property of a generator matrix that $Q_{ii} = \sum_j Q_{ij}$. Then for the jump term,

$$\begin{aligned} \sum_{s, X_s^- \neq X_s} \log \left(\frac{Q'_s(X_s^-, X_s)}{Q_s(X_s, X_s^-)} \right) &= \sum_{s, X_s^- \neq X_s} \log \left(\frac{Q_{\text{uniform}} \sqrt{\frac{\pi_t(y)}{\pi_t(x)}} \exp\left(\frac{1}{2}(\phi_t^\theta(y) - \phi_t^\theta(x))\right)}{Q_{\text{uniform}} \sqrt{\frac{\pi_t(x)}{\pi_t(y)}} \exp\left(\frac{1}{2}(\phi_t^\theta(y) - \phi_t^\theta(x))\right)} \right) \\ &= \sum_{s, X_s^- \neq X_s} \log \frac{\pi_t(y)}{\pi_t(x)} \end{aligned} \quad (28)$$

616 Now, we consider the Kolmogorov Forward Equation as in (Holderrieth et al., 2025) given by

$$\partial_t \pi_t(x) = \sum_{y \neq x} (\pi_t(y) Q_t^\theta(y, x) - \pi_t(x) Q_t^\theta(x, y)). \quad (29)$$

623 We shall divide both sides by $\pi_t(x)$ which gives

$$\begin{aligned} \partial_t \log \pi_t(x) &= \sum_{y \neq x} \left(\frac{\pi_t(y)}{\pi_t(x)} Q_t^\theta(y, x) - Q_t^\theta(x, y) \right) \\ &= \sum_{y \neq x} \left(\underbrace{\frac{\pi_t(y)}{\pi_t(x)} Q^{\text{Langevin}}(y, x) \exp\left(-\frac{\phi_t(y) - \phi_t(x)}{2}\right)}_{=Q^{\text{Langevin}}(x, y)} - Q^{\text{Langevin}}(x, y) \exp\left(\frac{\phi_t(y) - \phi_t(x)}{2}\right) \right) \\ &= \sum_{y \neq x} Q^{\text{Langevin}} \left(\exp\left(-\frac{\phi_t(y) - \phi_t(x)}{2}\right) - \exp\left(\frac{\phi_t(y) - \phi_t(x)}{2}\right) \right) \\ &= -2 \sum_{y \neq x} Q^{\text{Langevin}} \sinh\left(\frac{\phi_t(y) - \phi_t(x)}{2}\right) \end{aligned} \quad (30)$$

640 Now, letting the energy function be $U_t(x) = -\log \pi_t(x) + F_t$ and substituting $\partial_t \log \pi_t = -\partial_t U_t + \partial_t F_t$. We are then left
641 with the equation

$$-\partial_t U_t(x) + \partial_t F_t = -2 \sum_{y \neq x} Q^{\text{Langevin}} \sinh\left(\frac{\phi_t(y) - \phi_t(x)}{2}\right) \quad (31)$$

648 which rearranging this into a squared residual results in the PINN loss that we have desired

$$\mathcal{L}(\theta, F; t) = \mathbb{E}_{x \sim \text{Unif}} \left[\left| \partial_t U_t(x) - \underbrace{2 \sum_{y \neq x} Q_t^{\text{Langevin}}(x, y) \sinh\left(\frac{\phi_t^\theta(y) - \phi_t^\theta(x)}{2}\right)}_{\mathcal{K}_t^\theta \pi_t(x) \text{ (Net Flux)}} - \partial_t F_t \right|^2 \right] \quad (32)$$

659 \square

660 The remainder follows the proof of (Holderrieth et al., 2025). For every $0 \leq t \leq 1$,

$$\begin{aligned}
 661 \quad & \mathcal{L}^{\log\text{-var}}(\theta; t) \\
 662 \quad & = \mathbb{V}_{\mathbf{X} \sim \mathbb{Q}}[A_t] \\
 663 \quad & = \min_{\hat{F}_t \in \mathbb{R}} \mathbb{E}_{\mathbf{X} \sim \mathbb{Q}} [|A_t - \hat{F}_t|^2] \\
 664 \quad & = t^2 \min_{\partial_s \hat{F}_s \in \mathbb{R}, 0 \leq s \leq t} \mathbb{E}_{\mathbf{X} \sim \mathbb{Q}} \left[\left| \frac{1}{t} \int_0^t (\mathcal{K}_s^\theta \rho_s(X_s) - \partial_s \hat{F}_s) ds \right|^2 \right] \\
 665 \quad & \leq t^2 \min_{\partial_s \hat{F}_s \in \mathbb{R}, 0 \leq s \leq t} \mathbb{E}_{\mathbf{X} \sim \mathbb{Q}} \left[\frac{1}{t} \int_0^t |\mathcal{K}_s^\theta \rho_s(X_s) - \partial_s \hat{F}_s|^2 ds \right] \\
 666 \quad & = t^2 \min_{\partial_s \hat{F}_s \in \mathbb{R}, 0 \leq s \leq t} \mathbb{E}_{s \sim \text{Unif}_{[0,1]}, X_s \sim \mathbb{Q}_s} \left[|\mathcal{K}_s^\theta \rho_s(X_s) - \partial_s \hat{F}_s|^2 \right]. \tag{33}
 \end{aligned}$$

675 B.3. Discrete Schrödinger Bridge and Proof to Proposition 3.2

676 **Proposition B.2** (Infinitesimal Schrödinger Bridge optimality of the multiplicative control). *Fix a finite state space Ω and a
677 prescribed curve of pmfs $(\pi_t)_{t \in [0,T]}$ with $\pi_0 = \mu$ and $\pi_T = \nu$. Let Q_t^{Langevin} denote the base Langevin generator used in
678 the multiplicative parameterization, and consider the controlled generator*

$$680 \quad Q_t^\phi(x, y) = Q_t^{\text{Langevin}}(x, y) \exp\left(\frac{1}{2}(\phi_t(y) - \phi_t(x))\right), \quad x \neq y, \tag{34}$$

682 i.e. the forward multiplicative control in Equation 12 with a bounded total jump rate. Let \mathbb{P}^ϕ denote the path measure of the
683 resulting CTMC.
684

685 When there exists a pair (ϕ^*, F^*) such that the PINN residual in Eq. (12) vanishes pointwise, i.e.

$$686 \quad \partial_t U_t(x) - 2 \sum_{y \neq x} Q_t^{\text{Langevin}}(x, y) \sinh\left(\frac{1}{2}(\phi_t^*(y) - \phi_t^*(x))\right) - \partial_t F_t^* = 0 \quad \text{for all } (t, x), \tag{35}$$

689 then the induced path measure \mathbb{P}^{ϕ^*} is the solution of the infinitesimal Schrödinger bridge problem in the following sense:

- 691 1. (Marginal matching) The CTMC with generator $Q_t^{\phi^*}$ realizes the prescribed marginals: $\mathbb{P}_t^{\phi^*} = \pi_t$ for all $t \in [0, T]$
692 (equivalently, π_t satisfies the Kolmogorov forward equation for $Q_t^{\phi^*}$).
- 693 2. (Discrete-time SB approximation) For any partition $0 = t_0 < t_1 < \dots < t_N = T$, let $\mathbb{P}^{(N)}$ be obtained by
694 concatenating, on each interval $[t_i, t_{i+1}]$, the (unique) Schrödinger bridge between π_{t_i} and $\pi_{t_{i+1}}$ with reference
695 dynamics given by Q_t^{Langevin} restricted to that interval. Then, as the interval shrinks to 0, any weak limit point of
696 $(\mathbb{P}^{(N)})_N$ coincides with \mathbb{P}^{ϕ^*} .
- 697 3. (Infinitesimal SB optimality) Consequently, for each $t \in [0, T)$ and each $h \downarrow 0$, the restriction of \mathbb{P}^{ϕ^*} to $[t, t+h]$ is (to
698 first order in h) the Schrödinger bridge between π_t and π_{t+h} relative to the reference Q_t^{Langevin} ; equivalently, \mathbb{P}^{ϕ^*} is
699 locally Schrödinger-bridge optimal on infinitesimal time windows in the sense formalized by CMCD Proposition 3.4

700 *Proof.* We consider $D([0, T], \Omega)$, the càdlàg path space with the corresponding process X_t , and $\mathbb{P}^{\text{Langevin}}$ be the Langevin
701 CTMC law characterized by the generator Q_t^{Langevin} . Furthermore, we define the classes of marginal constraints over
702 continuous-time space,

$$703 \quad \mathcal{M}_\infty(\pi) := \{\mathbb{P} \in D([0, T], \Omega) : \mathbb{P}(X_t = \cdot) = \pi_t, \forall t \in [0, T]\} \tag{36}$$

704 over a time-discretized grid with N partitions $0 = t_0 < \dots < t_N = T$,

$$705 \quad \mathcal{M}_N(\pi) := \{\mathbb{P} : \mathbb{P}(X_{t_i} = \cdot) = \pi_{t_i}, \forall i \in \{0, \dots, N\}\}. \tag{37}$$

711 Furthermore, a discrete Schrödinger Bridge problem on an interval $[a, b]$ and α, β with respect to a CTMC path is defined by

$$712 \quad \arg \min \{D_{KL}(\mathbb{P} \| \mathbb{P}^{\text{Langevin}}) : \mathbb{P}(X_a) = \alpha, \mathbb{P}(X_b) = \beta\} \tag{38}$$

715 We first recognize that $\log \pi_t(x) = -U_t(x) + F_t$, which naturally results in
 716
 717

$$\partial_t \log \pi_t(x) = -\partial_t U_t(x) + \partial_t F_t. \quad (39)$$

719 Suppose equation 35 holds, then we can rearrange to give
 720
 721

$$\partial_t \log \pi_t(x) = -2 \sum_{y \neq x} Q^{\text{Langevin}} \sinh \left(\frac{\phi_t^*(y) - \phi_t^*(x)}{2} \right). \quad (40)$$

724 Now, use the controlled generator defined in equation 34 and the definition of hyperbolic sine to state that for a fixed
 725 space-time tuple (t, x) , we have
 726

$$\sum_{y \neq x} \left(\pi_t(y) Q_t^{\phi^*}(y, x) - \pi_t(x) Q_t^{\phi^*}(x, y) \right) = -2 \sum_{y \neq x} Q^{\text{Langevin}} \sinh \left(\frac{\phi_t^*(y) - \phi_t^*(x)}{2} \right), \quad (41)$$

730 from the detailed-balance symmetry. Combining equation 40 and 41, we have
 731
 732

$$\partial_t \log \pi_t(x) = \sum_{y \neq x} \left(\pi_t(y) Q_t^{\phi^*}(y, x) - \pi_t(x) Q_t^{\phi^*}(x, y) \right) \quad (42)$$

734 which exactly is the KFE for forward equation with generator $Q_t^{\phi^*}$. We then have the marginal law under \mathbb{P}^{ϕ^*} is π_t at all t
 735 which proves the first part.
 736

737 Now, we shall apply the Girsanov formula on CTMC process to obtain the KL divergence in the CTMC path-space given by
 738

$$D_{\text{KL}}(\mathbb{P} \parallel \mathbb{P}^{\text{Langevin}}) = \int_0^T \sum_{x \in \Omega} \mathbb{P}(X_t = x) \sum_{y \neq x} \left[Q_t(x, y) \log \frac{Q_t(x, y)}{Q_t^{\text{Langevin}}(x, y)} - Q_t(x, y) + Q_t^{\text{Langevin}}(x, y) \right] dt. \quad (43)$$

742 From the prescribed curve π_t , any $\mathbb{P} \in \mathcal{M}_\infty(\pi)$ admits predictable jump intensities $Q_t^{\mathbb{P}}(x, y)$ corresponding to the \mathbb{P} such
 743 that $(\pi_t, Q_t^{\mathbb{P}})$ such that it solves the KFE
 744

$$\partial_t \log \pi_t(x) = \sum_{y \neq x} \left(\pi_t(y) Q_t^{\mathbb{P}}(y, x) - \pi_t(x) Q_t^{\mathbb{P}}(x, y) \right) \quad (44)$$

745 and its converse holds. Moreover, we have the convex optimization problem given by
 746
 747

$$\min_{\mathbb{P} \in \mathcal{M}_\infty(\pi)} D_{\text{KL}}(\mathbb{P} \parallel \mathbb{P}^{\text{Langevin}}) \quad (45)$$

752 over generator Q_t under the KFE constrains. We furthermore have the optimal solution of the exponential tilt with respect to
 753 the potential field control $\phi(\cdot)$. Since the minimizer over $\mathcal{M}_\infty(\pi)$ is unique and must be an exponential tilt which satisfies
 754 the KFE constraints, and by defining an optimal exponential tilt
 755

$$Q_t^{\phi^*}(x, y) = Q_t^{\text{Langevin}} \exp \left(\frac{1}{2} (\phi_t^*(y) - \phi_t^*(x)) \right), \quad (46)$$

759 we can identify \mathbb{P}^{ϕ^*} as the $\mathcal{M}_\infty(\pi)$ -minimizer. Subsequently, we fix a partition $0 = t_0 < \dots < t_N = T$ and consider the
 760 time-discretized constrained class $\mathcal{M}_N(\pi)$. For \mathbb{P} that is absolutely continuous with respect to $\mathbb{P}^{\text{Langevin}}$, we shall decompose
 761 the KL-divergence as
 762

$$D_{\text{KL}}(\mathbb{P} \parallel \mathbb{P}^{\text{Langevin}}) = D_{\text{KL}}(\mathbb{P}_{t_0:N} \parallel \mathbb{P}_{t_0:N}^{\text{Langevin}}) + \sum_{i=0}^{N-1} \mathbb{E}_{\mathbb{P}} [D_{\text{KL}}(\mathbb{P}(\cdot | X_{t_i}, X_{t_{i+1}}) \parallel \mathbb{P}^{\text{Langevin}}(\cdot | X_{t_i}, X_{t_{i+1}}))] \quad (47)$$

766 Because $\mathcal{M}_N(\pi)$ fixes the marginal at the grid time only instead of the joint distribution, we see that the minimization of KL
 767 divergence in each segment forces each conditional segment law to be the Schrödinger Bridge optimality with the reference
 768 Langevin path measure. Thus the optimizer is the concatenation of two-marginal SBs on each segment $[t_i, t_{i+1}]$.
 769

Furthermore, we have monotonicity of the optimal value as $\mathcal{M}_\infty \subset \mathcal{M}_N$ and correspondingly

$$\inf_{\mathbb{P} \in \mathcal{M}_N(\pi)} D_{\text{KL}}(\mathbb{P} \parallel \mathbb{P}^{\text{Langevin}}) \leq \inf_{\mathbb{P} \in \mathcal{M}_\infty(\pi)} D_{\text{KL}}(\mathbb{P} \parallel \mathbb{P}^{\text{Langevin}}) = D_{\text{KL}}(\mathbb{P}^{\phi^*} \parallel \mathbb{P}^{\text{Langevin}}), \quad (48)$$

as \mathbb{P}^{ϕ^*} is feasible for all \mathcal{M}_N . Furthermore, we have $D_{\text{KL}}(\mathbb{P}^{(N)} \parallel \mathbb{P}^{\text{Langevin}})$ uniformly bounded by $D_{\text{KL}}(\mathbb{P}^{\phi^*} \parallel \mathbb{P}^{\text{Langevin}})$. Furthermore, by the bounded total jump rate of the system given in the assumption, we have the tight sequence $\{\mathbb{P}^{(N)}\}$ on a finite state space Ω .

Let $\mathbb{P}^{(N_k)} \Rightarrow \bar{\mathbb{P}}$. For each fixed t , we choose grid times $t_i^{(k)} \rightarrow t$. Since CTMCs have probability 0 of a jump at any deterministic time, evaluation of the map $X \mapsto X_t$ is continuous $\bar{\mathbb{P}}$ -almost surely. Hence, $\bar{\mathbb{P}}(X_t) = \lim_k \mathbb{P}^{(N_k)}(X_{t_i^{(k)}}) = \pi_t$.

Therefore $\bar{\mathbb{P}} \in \mathcal{M}_\infty(\pi)$. Furthermore, KL is lower semicontinuous under weak convergence, so $D_{\text{KL}}(\bar{\mathbb{P}} \parallel \mathbb{P}^{\text{Langevin}}) \leq \liminf_k D_{\text{KL}}(\mathbb{P}^{(N_k)} \parallel \mathbb{P}^{\text{Langevin}}) \leq D_{\text{KL}}(\mathbb{P}^{\phi^*} \parallel \mathbb{P}^{\text{Langevin}})$. But \mathbb{P}^{ϕ^*} is the unique minimizer over $\mathcal{M}_\infty(\pi)$ so $\bar{\mathbb{P}} = \mathbb{P}^{\phi^*}$.

To show this in terms of the CTMC generator, $[t, t+h]$, the endpoint constraint implies the net change

$$\pi_{t+h}(x) - \pi_t(x) = \int_t^{t+h} \sum_{y \neq x} (\pi_s(y)Q_s(y, x) - \pi_s(x)Q_s(x, y)) ds. \quad (49)$$

For an infinitesimal step $h \rightarrow 0$, we have

$$\partial_t \pi_t(x) = \sum_{y \neq x} (\pi_t(y)Q_t(y, x) - \pi_t(x)Q_t(x, y)) + \mathcal{O}(1), \quad (50)$$

which indicates that it solves the cost functional at the integrand of equation 43. Therefore, the optimal point of equation 35 becomes the unique solution Q_t^* of the instantaneous cost functional. Above, we have identified $Q_t^{\phi^*}$ as the unique minimizer of the functional which indicates that finally yields the desired result $Q_t^* = Q_t^{\phi^*}$ completing the proof. \square

C. Glauber Dynamics

Glauber dynamics is an algorithm to simulate an Ising model via a time discretization with a random jump of spin proposed by Glauber (Glauber, 1963). Similar to the Metropolis-Hastings algorithm, it is a local MC algorithm which

depending on the MH or Glauber dynamics, the acceptance rate is modified to the line 5 of above expression. In Langevin case, the acceptance ratio will become

$$q(Y|X_t) = \frac{\exp\left(-\frac{1}{2\alpha} \|Y - X_t - \frac{\alpha}{2} \nabla U(X_t)\|_2^2\right)}{Z_\Omega(X_t)}, \quad (51)$$

where Z_Ω is the normalizing constant summed over arbitrary space. Intuitively, Glauber dynamics can be interpreted as the movement with local information from its neighbors compared to the greedy gradient step in the Langevin mechanism. Furthermore, the SDE corresponding to the Glauber dynamics is given by

$$dX_t = -\tanh\left(\frac{\beta}{2} \nabla U(X_t)\right) dt + \sqrt{2\beta^{-1}} dW_t \quad (52)$$

which has modified the standard overdamped Langevin Dynamics using the hyperbolic tangent mapping to the drift term

$$dX_t = -\nabla U(X_t) dt + \sqrt{2\beta^{-1}} dW_t. \quad (53)$$

By this mapping, the drift of the governing SDE is upper and lower bounded by $(-1, 1)$ which allows a stable convergence.

Consequently, instead of the Langevin transition rate matrix $\sqrt{\frac{\pi(y)}{\pi(x)}} Q^{\text{uniform}}$, the transition has now been modified to $\frac{1}{1+e^{\beta \Delta U}} Q^{\text{uniform}} = \frac{e^{-\beta \Delta U}}{1+e^{-\beta \Delta U}} Q^{\text{uniform}}$.

Now, we show that the Glauber Dynamics does not violate Proposition 3.1.

Proposition C.1. *For the CTMC process with the Glauber forward multiplicative control where the concrete score Langevin term of equation 12 has been modified to Glauber dynamic term $\frac{1}{1+e^{\beta \Delta U}} Q^{\text{uniform}}$ the Proposition 3.1 holds.*

825 *Proof.* We obtain the Base Glauber Generator given by

$$826 \quad Q^{\text{Glauber}}(x, y) = Q^{\text{Uniform}}(x, y) \frac{1}{1 + \exp(\beta(U(y) - U(x)))} = Q^{\text{Uniform}}(x, y) \frac{\pi(y)}{\pi(x) + \pi(y)} \quad (54)$$

827 which similarly $Q^{\text{Glauber}}(y, x) = Q^{\text{Uniform}}(y, x) \frac{\pi(x)}{\pi(x) + \pi(y)}$. We naturally see that

$$828 \quad \frac{Q_t^{\theta'}(y, x)}{Q_t^{\theta}(x, y)} = \frac{Q_t^{\theta}(x, y) \frac{\pi_t(y)}{\pi_t(x)}}{Q_t^{\theta}(x, y)} = \frac{\pi_t(y)}{\pi_t(x)}. \quad (55)$$

829 which is the Nelson's relation. Thus, the Jump Term remains unchanged.

830 Now, we consider the flux term can be computed by

$$831 \quad Q_t^{\theta}(x, y) - Q_t^{\theta'}(x, y) = Q^{\text{Glauber}}(x, y) \exp\left(\frac{\phi_t^{\theta}(y) - \phi_t^{\theta}(x)}{2}\right) - Q^{\text{Glauber}}(x, y) \exp\left(-\frac{\phi_t^{\theta}(y) - \phi_t^{\theta}(x)}{2}\right) \\ 832 = Q^{\text{Glauber}}(x, y) \left[\exp\left(\frac{\phi_t^{\theta}(y) - \phi_t^{\theta}(x)}{2}\right) - \exp\left(-\frac{\phi_t^{\theta}(y) - \phi_t^{\theta}(x)}{2}\right) \right] \\ 833 = -2Q^{\text{Glauber}}(x, y) \sinh\left(\frac{\phi_t^{\theta}(y) - \phi_t^{\theta}(x)}{2}\right). \quad (56)$$

834 which by computation similar to above, letting the energy function be $U_t(x) = -\log \pi_t(x) + F_t$ and substituting $\partial_t \log \pi_t = -\partial_t U_t + \partial_t F_t$. We are then left with the equation

$$835 \quad -\partial_t U_t(x) + \partial_t F_t = -2 \sum_{y \neq x} Q^{\text{Glauber}} \sinh\left(\frac{\phi_t(y) - \phi_t(x)}{2}\right) \quad (57)$$

836 which rearranging this into a squared residual results in the PINN loss that we have desired

$$837 \quad \mathcal{L}(\theta, F; t) = \mathbb{E}_{x \sim \text{Unif}} \left[\underbrace{\partial_t U_t(x) - 2 \sum_{y \neq x} Q_t^{\text{Glauber}}(x, y) \sinh\left(\frac{\phi_t^{\theta}(y) - \phi_t^{\theta}(x)}{2}\right)}_{\mathcal{K}_t^{\theta} \pi_t(x) \text{ (Net Flux)}} - \partial_t F_t \right]^2 \quad (58)$$

838 \square

839 We further note than under our circumstance where we set the temperature constant for training, we have U_t constant which
840 removes the $\partial_t U_t(x)$ term from consideration.

D. Experiment Details

841 In this section, we elaborate the experimental settings and detailed computation methods for Ising and Potts model.

842 To increase robustness, we performed 5 independent training runs report the median performance. We have ran 50k
843 for Ising model trained on β_{high} as the performance already saturates to the maximum performance. We then for each
844 trained checkpoint, we make 5 evaluations of 1024 sample batch. Lastly, we average the performance over all checkpoints
845 corresponding to the model and hyperparameter. We run each training on the on the Ising model using 6000ada GPU by
846 PyTorch Lightning module adopted from official LEAPS repository, and a6000 and a5000 for Potts model for architectural
847 equivalence. I adopted the default setting given by $3e-4$ and 125 diffusion steps except for the annealing step for training
848 efficiency without significantly deteriorating the quality.

849 The Effective Sample Size (ESS) is a common evaluation metric for a sampler that has importance weight for its computation.
850 The basic definition of an ESS is given by

$$851 \quad \text{ESS}_r = \frac{(\mathbb{E}_{q^{\theta}} [w])^2}{\mathbb{E}_{q^{\theta}} [w^2]} \quad \text{where} \quad w = \frac{\gamma(\mathbf{x})}{q^{\theta}(\mathbf{x})}. \quad (59)$$

880 We can interpret this by

$$\begin{aligned}
 \text{ESS}_r &= \frac{(\mathbb{E}_{q^\theta}[w])^2}{\mathbb{E}_{q^\theta}[w^2]} \\
 &= \frac{\mathbb{E}_{q^\theta}[w^2] + (\mathbb{E}_{q^\theta}[w])^2 - \mathbb{E}_{q^\theta}[w^2]}{\mathbb{E}_{q^\theta}[w^2]} \\
 &= 1 - \frac{\mathbb{E}_{q^\theta}[w^2] - (\mathbb{E}_{q^\theta}[w])^2}{\mathbb{E}_{q^\theta}[w^2]} \\
 &= 1 - \frac{\mathbb{V}_{q^\theta}[w]}{\mathbb{E}_{q^\theta}[w^2]}.
 \end{aligned} \tag{60}$$

891 In other words, higher ESS can be interpreted as the smaller variance in the ratio between target density value and neural
 892 network predicted value. First, we compute the log importance weight from the accumulated violation of the KFE given by
 893

$$A_t = \int_0^t \mathcal{K}_s \rho_s(X_s) ds. \tag{61}$$

894 We compute this ESS in our experiment from the obtained log weights A_t by
 895

$$\text{ESS}_r = \frac{\left(N^{-1} \sum_{i=1}^N \exp(A_r^i) \right)^2}{N^{-1} \sum_{i=1}^N \exp(2A_r^i)}. \tag{62}$$

903 Subsequently, we shall compute the ground truth observables of Ising and Potts model for evaluation by the absolute error
 904 with the ground truth. We have computed the Hamiltonian and its corresponding density (unnormalized) in equation 18 and
 905 19. We then need to compute the row-wise magnetization and correlation error.
 906

Ising Model Row-wise magnetization $M_\nu^{\text{row}}(k)$, quantifies the total expected magnetization for all sites within the k -th row:

$$M_\nu^{\text{row}}(k) = \sum_{i \in \text{row}(k)} M_\nu(x^i), \tag{63}$$

911 where $M_\nu(x^i) = \mathbb{E}_{\nu(x)}[x^i]$ is the magnetization of a single state i .
 912

913 To capture the spatial dependencies, we define the row-wise correlation $C_\nu^{\text{row}}(k, l)$ as the sum of 2-point correlations between
 914 sites in rows k and l that share the same column index:
 915

$$C_\nu^{\text{row}}(k, l) = \sum_{\substack{i \in \text{row}(k), j \in \text{row}(l) \\ i, j \text{ same col}}} C_\nu(i, j), \tag{64}$$

916 where the pairwise correlation is defined as the covariance $C_\nu(i, j) = \mathbb{E}_{\nu(x)}[x^i x^j] - \mathbb{E}_{\nu(x)}[x^i] \mathbb{E}_{\nu(x)}[x^j]$.
 917

Potts model Similar to the Ising model, we evaluate the structural consistency of the generated states using row-wise
 918 statistics adapted for the q -state Potts model.
 919

920 The row-wise magnetization $M_\nu^{\text{row}}(k)$ aggregates the Potts magnetization for all sites within the k -th row:
 921

$$M_\nu^{\text{row}}(k) = \sum_{i \in \text{row}(k)} M_\nu^{\text{potts}}(x^i), \tag{65}$$

922 where the site-specific magnetization is defined as:
 923

$$M_\nu^{\text{potts}}(i) = \frac{q \cdot \max_{1 \leq c \leq q} (n_c^i/n) - 1}{q - 1}, \tag{66}$$

924 with n_c^i/n representing the frequency of state c at site i (or the marginal probability).
 925

935 The row-wise correlation $C_\nu^{\text{row}}(k, l)$ sums the pairwise Potts correlations between sites in rows k and l that share the same
 936 column index:

$$C_\nu^{\text{row}}(k, l) = \sum_{\substack{i \in \text{row}(k), j \in \text{row}(l) \\ i, j \text{ same col}}} C_\nu^{\text{potts}}(i, j), \quad (67)$$

940 where the pairwise correlation accounts for the q -state symmetry:

$$C_\nu^{\text{potts}}(i, j) = \mathbb{E}_{\nu(x)} \left[\mathbf{1}_{x^i = x^j} - \frac{1}{q} \right]. \quad (68)$$

945 **Computation** The respective observables magnetization and correlation coefficient for Ising and Potts model can be
 946 computed by the following formula:

$$\begin{aligned} \text{Magnetization Error} &= \frac{1}{2L} \sum_k (|M_\nu^{\text{row}}(k) - M_\pi^{\text{row}}(k)| + |M_\nu^{\text{col}}(k) - M_\pi^{\text{col}}(k)|) \\ \text{Correlation Error} &= \frac{1}{L^2} \sum_{(k, l)} (|C_\nu^{\text{row}}(k, l) - C_\pi^{\text{row}}(k, l)| + |C_\nu^{\text{col}}(k, l) - C_\pi^{\text{col}}(k, l)|). \end{aligned} \quad (69)$$

953 For the ground truth value, we use the Swendsen-Wang algorithm which is known to well-simulate the spin systems like
 954 Ising or Potts models. It is a cluster-based Monte Carlo simulation which flips a cluster of spins instead of a single spin-flip.
 955 Its detailed algorithmic implementation is provided in algorithm 2

958 Algorithm 2 Swendsen-Wang Algorithm for the Ising Model

```

959 1: Input: Graph  $G = (V, E)$ , current spin configuration  $\sigma \in \{-1, +1\}^{|V|}$ , inverse temperature  $\beta$ , coupling constant
960    $J > 0$ .
961 2: Output: Updated spin configuration  $\sigma$ .
962 3: Initialize the set of active bonds  $A \leftarrow \emptyset$ .
963 4: for each edge  $(i, j) \in E$  do
964   5:   if  $\sigma_i = \sigma_j$  then
965     6:     Calculate bond probability  $p = 1 - \exp(-2\beta J)$ .
966     7:     Generate  $u \sim \text{Uniform}(0, 1)$ .
967     8:     if  $u < p$  then
968       9:       Add bond  $(i, j)$  to  $A$ .
969     10:    end if
970   11:   end if
971 12: end for
972 13: Identify connected components (clusters)  $\{C_1, \dots, C_k\}$  in the graph  $(V, A)$ .
973 14: for each cluster  $C_m \in \{C_1, \dots, C_k\}$  do
974   15:   Sample a new spin value  $v \sim \{-1, +1\}$  with probability 0.5.
975   16:   for each site  $i \in C_m$  do
976     17:     Update  $\sigma_i \leftarrow v$ .
977   18:   end for
978 19: end for
979 20: return  $\sigma$ 

```

982 **Divergence Measures** In 4×4 , we are able to compute the ground truth $\log Z$ and divergence measure based on the
 983 empirically from the generated samples from the model. We specifically choose to evaluate by using Total Variation distance
 984 (TV), KL divergence (KL), χ^2 -divergence.

985 The total variation distance is defined as

$$\text{TV}(p\|q) = \frac{1}{2} \sum_x |p(x) - q(x)| \quad (70)$$

which denotes the total area between two curves scaled by 0.5.

The KL divergence is defined by

$$\text{KL}(p\|q) = \sum_x p(x) \log \frac{p(x)}{q(x)}. \quad (71)$$

The χ^2 -divergence is defined as a f -divergence with $f(x) = (x - 1)^2$, which results in the formulation given by

$$\chi^2(p\|q) = \sum_x \frac{(p(x) - q(x))^2}{q(x)}. \quad (72)$$

E. Algorithms

Here, we present the pseudocode that describes the training and sampling procedure of our Discrete Controlled Monte Carlo Sampler.

Our algorithm differentiates this work to the previous by:

Table 4. Comparison of different diffusion and control methods.

| Method | Domain | Control Type | Marginal Prescription | Preconditioning | Loss Type | ISB Optimality |
|--------|-----------------|--------------|-----------------------|--------------------------|-------------|----------------|
| CMCD | Continuous | Additive | Geometric | Langevin | RND | ○ |
| LEAPS | Discrete | - | None | None | PINN | ✗ |
| MDNS | Discrete | - | Learned | Limited | RND | ✗ |
| DCMC | Discrete | Both | Geometric | Langevin, Glauber | PINN | ○ |

First, we may obtain batch computation of the PINN loss as which in multiplicative control becomes

$$\begin{aligned} \mathcal{L} &= \mathbb{E}_{t,x} \left[\left| \partial_t U_{t_b}(X_b) - 2 \sum_{y \neq X_b} Q_{t_b}^{\text{Langevin}}(X_b, y) \sinh \left(\frac{\phi_\theta(y, t_b) - \phi_\theta(X_b, t_b)}{2} \right) - \partial_t F_\psi(t_b) \right|^2 \right] \\ &= \frac{1}{B} \sum_m \left| \mathcal{K}_{t_m}^\theta \rho_{t_m}(X_{t_m}^m) - \partial_{t_m} F_{t_m}^\phi \right|^2 \end{aligned} \quad (73)$$

The sample is achieved in a similar method as the LEAPS which becomes the basis of our implementation. The differentiation is that in the sample generation process, we adjust the generator using the appropriate control that was generated in equation 9 or 12 depending on the choice of the control that we make. Furthermore, we only identify the training procedure for the multiplicative control as we have already mentioned and shown that the additive control has the same loss function as the leaps which ultimately results in the same training procedure.

Algorithm 3 Training Procedure (Multiplicative Control)

Require: B batch size, N time steps, model $\phi_\theta(x, t)$, free energy net $F_\psi(t)$, learning rate η , maximum training step

- 1: **while** step < maximum training step **do**
 - 2: Generate sample roll-out $\{(X_t^m, A_t^m, t)\}_{t,m}$ (use algorithm 1)
 - 3: $\mathcal{L}(\theta, \psi) \leftarrow \frac{1}{B} \sum_{b=1}^B \left(\partial_t U_{t_b}(X_b) - 2 \sum_{y \neq X_b} Q_{t_b}^{\text{Langevin}}(X_b, y) \sinh \left(\frac{\phi_\theta(y, t_b) - \phi_\theta(X_b, t_b)}{2} \right) - \partial_t F_\psi(t_b) \right)^2$
 - 4: $\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}(\theta, \phi)$
 - 5: $\phi \leftarrow \phi - \eta \nabla_\phi \mathcal{L}(\theta, \phi)$
 - 6: **end while**
-

F. Network Architectures

Our network architecture relies on (Holderrieth et al., 2025) which adapts the Locally Equivariant Convolutional Neural Network (LEC) for efficient computation of the neighborhood of the current state. As the universal representation theorem

for LEC has already been established, we do not have a theoretical barrier between the representable hypothesis class of the neural network and the space of admissible control that is used to control the CTMC.

We hereby show that we insist on LEC structure to represent the control with our parameterizations. For the additive control, it is trivial to use the LEC architecture as it equivalently constructs the loss function as constructed in the LEAPS paper. For the multiplicative control in equation 12, justification of LEC architecture is less trivial. We look at the PINN loss for the multiplicative control at equation 24, more specifically the Net Flux term involved

$$2 \sum_{y \neq x} Q_t^{\text{Langevin}}(x, y) \sinh \left(\frac{\phi_t^\theta(y) - \phi_t^\theta(x)}{2} \right). \quad (74)$$

Similar to the LEAPS, we need to compute the summation over all possible jumps based on the summation $\sum_{y \neq x}$ which requires infeasible number of neural network evaluation that scales exponentially with respect to the dimension. Therefore, we need a method that efficiently evaluates the summation, which we achieve by incorporating the inductive bias of the local equivariance.

Definition F.1. A neural network $G_t^\theta : S \rightarrow (\mathbb{R}^{N-1})^d$ such that

$$x \mapsto (G_t^\theta(\tau, i | x))_{i=1, \dots, d, \tau \in \mathcal{T} \setminus \{x_i\}} \quad (75)$$

is locally equivariant if

$$G_t^\theta(\tau, i | x) = -G_t^\theta(x^i, i | \text{Swap}(x, i, \tau)) \quad (i = 1, \dots, d) \quad (76)$$

for the swap operator $\text{Swap}(x, i, \tau) = (x_1, \dots, x_{i-1}, \tau, x_{i+1}, \dots, x_d)$.

Intuitively, the local equivariance of a neural network can be understood as the reversibility. Definition F.1 explicitly states that a neural network is locally equivariant if the cost of $A \rightarrow B$ is equal to the negative cost of $B \rightarrow A$. We simply see that for

$$\Delta_\phi(x, y) = \phi(y) - \phi(x) \quad (77)$$

representing the potential difference between the state y and x ,

$$\Delta_\phi(y, x) = -(\phi(y) - \phi(x)) = -\Delta_\phi(x, y) \quad (78)$$

satisfying the condition in equation 76.

G. Additional Results

G.1. Visualization of 16×16 Grid Samples

In this section, we provide uncurated visualizations the generated samples which provides the lattice. The visualization shows the landscape of the generated samples as an image of spins that the pink and blue color corresponding to up and down spins. We see that as the inverse temperature decreases, the spin becomes more arbitrary distributed across the lattice which implies smoother energy landscape as a function of the spin.

This visualization suggests a potential explanation to the reason of that our DCMC faces challenge in learning the Boltzmann distribution with low temperature. By bounding the log variance, our model minimizes the local violation of the KFE along the transport. However, we see that the generated samples are focused significantly on the (approximately) all up-spin or all down-spin state, implying that there is a significant barrier (or valley) in between the modes. In such case, the PINN fails to efficiently learn the generator or field map in the barrier region where the signal almost disappears.

G.2. Results on 4×4 Ising

In the table above and here, we adopt the numerical values for comparison obtained from MDNS. Furthermore, we omit the path KL comparison metric as our method is not an Stochastic Optimal Control formulated sampler, which the comparison to measure the discrepancy with the optimal path measure seems irrelevant. We compare the result with the various loss proposed in the MDNS, which includes Relative Entropy with REINFORCE (RERF) $\mathcal{F}_{\text{RERF}}$, Log-Variance (LV) \mathcal{L}_{LV} , Cross-Entropy (CE) \mathcal{L}_{CE} and Weighted Denoising Cross-Entropy (WDCE) $\mathcal{L}_{\text{WDCE}}$.



Figure 1. Visual comparison of different DCMC models. Top row: DCMC-Add; Middle row: DCMC-Langevin; Bottom row: DCMC-Glauber.

We now present the remaining result of the DCMC tested on 4×4 Ising model tested on different temperatures ($\beta_{\text{critical}} = 0.4407$) and ($\beta_{\text{low}} = 0.6$). In contrast to the high temperature case, at critical temperature, the top performance varies around different methods per evaluation metric. We see that the performance of DCMC outperforms all MDNS in ESS, KL-divergence and χ^2 -divergence metric. On the other hand, we see that the log variance loss of the MDNS achieves the top performance in the TV divergence which implies that the direct minimization of the log variance divergence achieves a small estimation error across all states demonstrating the alleviation of the mode-collapse issue rising from the KL-divergence based loss such as RERF. Furthermore, similar to the high temperature case, we see that the DCMC sampler falls behind MDNS in accurately capture the log normalization constant.

Similarly in low temperature case, we see that despite having a lower ESS compared to the MDNS, we accurately model the distribution in all divergence measures, indicating that our model is efficiently capturing the landscape of the target density while may subject to variation in the output itself. Furthermore, it may imply that despite having larger error in the magnetization and correlation coefficient for high dimensional Ising, our DCMC has learned the distribution sufficiently well which may have lower divergence metric with in the underlying distribution that is practically impossible to evaluate.

G.3. Analysis on the Potts Model

We have witnessed a significant drop in the performance of our model when the inverse temperature of the Potts model has increased to 1.2. We believe this is not the problem of the loss function itself, but rather a loss function not converging sufficiently. We plot the 100 iteration running average plot of the training loss curve trained on Potts model with inverse temperature 1.2 in figure 2 and compare it with the training loss curve on the ising model with inverse temperature 0.6.

1155
 1156 *Table 5.* Results for learning 4×4 Ising model with $J = 1$, $h = 0.1$ and $\beta_{\text{critical}} = 0.4407$, best in **bold**.

| Method | ESS \uparrow | $\text{TV}(\hat{p}_{\text{samp}}, \pi) \downarrow$ | $\text{KL}(\hat{p}_{\text{samp}} \parallel \pi) \downarrow$ | $\chi^2(\hat{p}_{\text{samp}} \parallel \pi) \downarrow$ | Abs. err. of $\log \hat{Z} \downarrow$ |
|-----------------------------|----------------|--|---|--|--|
| DCMC-Langevin | 0.9993 | 0.0312 | 0.0221 | 0.0743 | 0.11867 |
| DCMC-Glauber | 0.9991 | 0.0382 | 0.0234 | 0.0717 | 0.11866 |
| DCMC-Add | 0.9989 | 0.0320 | 0.0218 | 0.0772 | 0.11929 |
| $\mathcal{F}_{\text{RERF}}$ | 0.8480 | 0.0841 | 0.0521 | 0.1691 | 0.00150 |
| \mathcal{F}_{LV} | 0.9809 | 0.0301 | 0.0222 | 0.0830 | 0.00106 |
| \mathcal{F}_{CE} | 0.9545 | 0.0454 | 0.0327 | 0.1824 | 0.00175 |
| $\mathcal{F}_{\text{WDCE}}$ | 0.9644 | 0.0789 | 0.0375 | 0.0839 | 0.00010 |

 1166 *Table 6.* Results for learning 4×4 Ising model with $J = 1$, $h = 0.1$, and $\beta = 0.6$. Best results in **bold**.

| Method | ESS \uparrow | $\text{TV}(\hat{p}_{\text{samp}}, \pi) \downarrow$ | $\text{KL}(\hat{p}_{\text{samp}} \parallel \pi) \downarrow$ | $\chi^2(\hat{p}_{\text{samp}} \parallel \pi) \downarrow$ | Abs. err. of $\log \hat{Z} \downarrow$ |
|-----------------------------|----------------|--|---|--|--|
| DCMC-Langevin | 0.9603 | 0.0133 | 0.0074 | 0.0793 | 0.20192 |
| DCMC-Glauber | 0.9418 | 0.0205 | 0.0073 | 0.0602 | 0.19263 |
| DCMC-Add | 0.8781 | 0.0187 | 0.0058 | 0.0797 | 0.20984 |
| $\mathcal{F}_{\text{RERF}}$ | 0.9196 | 0.0320 | 0.0200 | 0.4071 | 0.00788 |
| \mathcal{F}_{LV} | 0.9722 | 0.0177 | 0.0098 | 0.1864 | 0.00257 |
| \mathcal{F}_{CE} | 0.9855 | 0.0147 | 0.0138 | 2.8388 | 0.00259 |
| $\mathcal{F}_{\text{WDCE}}$ | 0.9465 | 0.0418 | 0.0282 | 1.6582 | 0.00373 |

We see that the two training curves reach plateau, the loss differs significantly, where the loss for Ising has converged to approximately 50 while the Potts has converged to 125. We further see that the training converges better as the temperature increases (or inverse temperature decreases) which explains the reason why the ESS has been very high in the case of high-temperature of the Ising model.

 1183
 1184
 1185
 1186
 1187
 1188
 1189
 1190
 1191
 1192
 1193
 1194
 1195
 1196
 1197
 1198
 1199
 1200
 1201
 1202
 1203
 1204
 1205
 1206
 1207
 1208
 1209

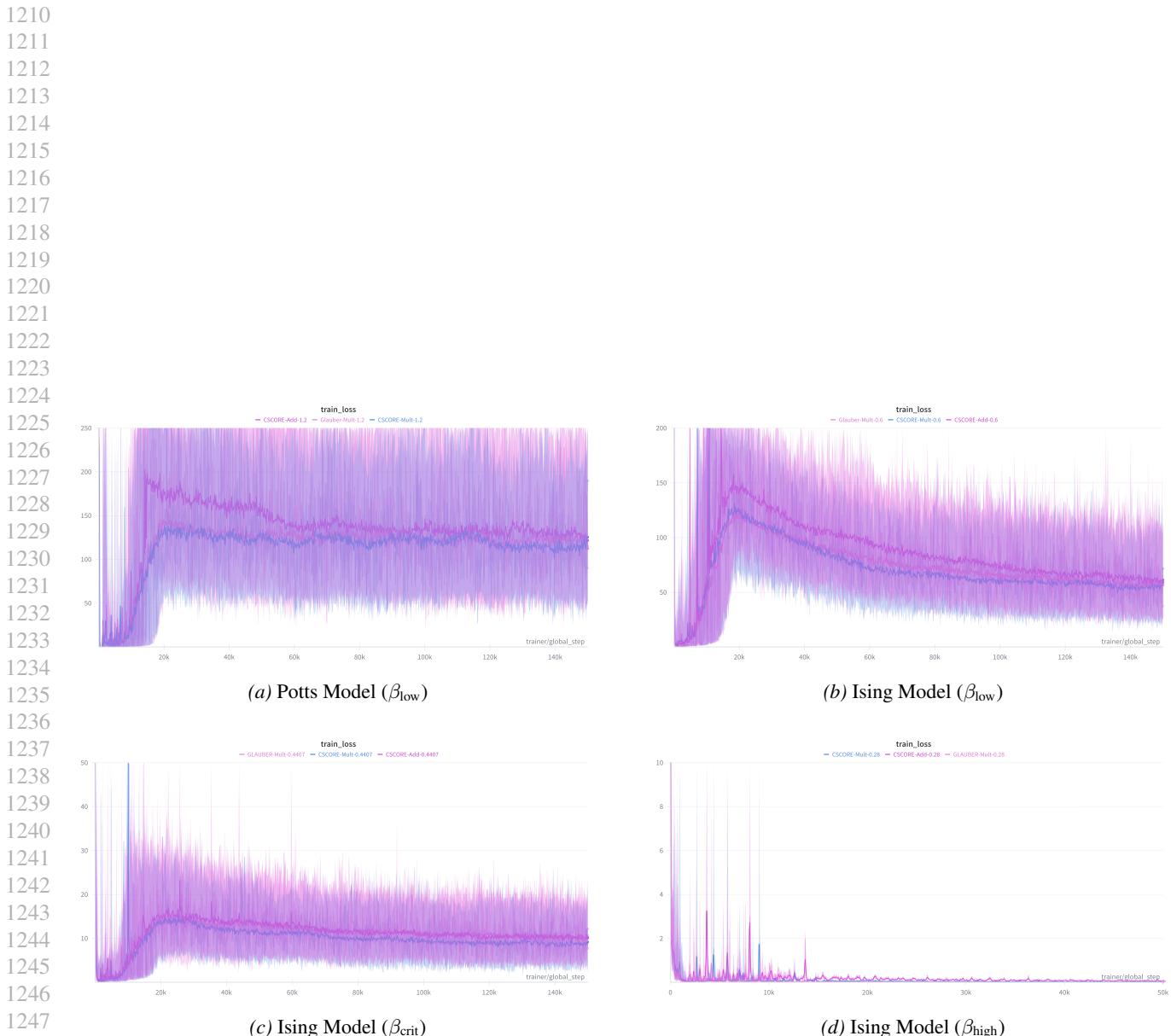


Figure 2. Comparison of loss curves across different models and temperature regimes.