USHE: VARIANCE-CONTROLLED PROBABILISTIC SPIKING ENCODING FOR LOW-LATENCY SNNS

Anonymous authors
Paper under double-blind review

ABSTRACT

Input encoding is a central design choice in spiking neural networks (SNNs), shaping how information and stochasticity enter both learning and inference. Widely used Bernoulli rate encoding relies on independent sampling, introducing input variance that can limit representational precision at low latency. We introduce unbiased spiking hypergeometric encoding (USHE), a tunable family of probabilistic encoders that preserves input intensities in expectation while controlling spike-rate variance and temporal dependence. By decoupling the latent population size from the simulation horizon, USHE controls encoding stochasticity without increasing network latency. Theoretically, we characterize USHE members through constrained variance-minimization problems and establish independent Bernoulli encoding as the infinite-population limit at fixed latency. Experiments on CIFAR-10 and CIFAR-100 show consistent improvements in clean accuracy over Bernoulli encoding and reveal distinct finite-population operating points under adversarial attacks and Gaussian corruption. Controlled encoder substitutions further separate training and inference effects, showing that the inference population governs a systematic clean–robustness trade-off whose magnitude depends on the trained model and architecture. Together, these results establish population-controlled temporal dependence as a principled design dimension for stochastic spike encoding, rather than treating input encoding as a fixed preprocessing step.

1 INTRODUCTION

Spiking neural networks (SNNs) process information through discrete events and stateful neuronal dynamics, offering a route to event-driven computation. The combination of surrogate-gradient learning and residual architectures has transformed deep SNNs into practical candidates across various machine learning domains, achieving notable success in visual recognition (Neftci et al., 2019; Fang et al., 2021). For static images, however, the input encoder must first construct a temporal representation. This choice is particularly consequential at short simulation horizons: it determines how accurately a finite spike train represents the input and which fluctuations reach the downstream dynamics. Efficient low-latency inference therefore requires attention not only to the network, but also to the probability law of its input sequence.

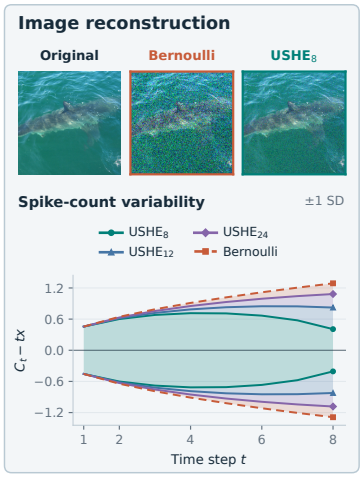


Figure 1: **Top:** Image reconstruction comparison under one trial for latency $T = 8$; **Bottom:** Variance of the number of emitted spikes per fixed random input USHE family for various N.

Two common choices illustrate the resulting tension. Direct encoding repeatedly supplies real-valued inputs, avoiding sampling error but not producing a binary input code. Bernoulli rate encoding instead maps each intensity $p \in [0, 1]$ to T independent binary draws. Its empirical rate is unbiased, but has variance $p(1-p)/T$, which can be substantial when only a few timesteps are available. Rate encoding has also exhibited stronger adversarial robustness than direct encoding under studied protocols, and its connection to randomized smoothing

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

has motivated certified robustness guarantees (Mukhoty et al., 2024). Thus, the randomness that reduces clean reconstruction precision can also alter sensitivity to perturbations. Neither eliminating stochasticity nor maximizing it is necessarily the appropriate choice for the complete classifier.

Existing approaches modify different aspects of this interface. Signed rate encoding enables mean-centering through a ternary spike alphabet (Mukhoty et al., 2025). FEEL-SNN combines frequency-selective encoding with adaptive leak factors (Xu et al., 2024), while Structured Temporal Orthogonal Decorrelation (STOD) targets temporal gradient alignment through constrained input-layer kernels (Zhang et al., 2026). We ask a complementary question: *can temporal sampling variability be controlled while preserving the binary alphabet, every timestep's marginal distribution, and the network's simulation horizon?* Such a construction would isolate temporal dependence from changes in expected input activity, learned preprocessing, and neuron dynamics.

We introduce *Unbiased Spiking Hypergeometric Encoding* (USHE), a family indexed by a nonnegative integer n. At visible horizon T, USHE_N uses a latent population of size $N \geq T$. Given an intensity p, it stochastically rounds Np to an adjacent integer spike budget and samples T positions without replacement from the resulting binary population. The unused positions require no additional network timesteps. Every emitted spike remains marginally Bernoulli(p), and the expected total count remains Tp, independently of N. Finite populations instead introduce nonpositive temporal covariance, reducing empirical-rate variance without lowering the expected input firing rate. Population size therefore controls the dependence between spikes, rather than their individual distributions.

Our main contributions and findings are:

- **A variance-optimal encoding family.** We introduce USHE, which controls temporal dependence at fixed latency while preserving binary spikes and Bernoulli marginals. We derive its exact statistics, prove rate-reconstruction variance optimality under finite exchangeability (with global optimality at $N = T$) and establish convergence in total variation to independent Bernoulli encoding. As a limiting case where $N \to \infty$ and any fixed latency T, we recover Bernoulli encoding.
- **Improved low-latency accuracy and attack-dependent robustness.** Matched experiments on CIFAR-10 and CIFAR-100 show consistent clean-accuracy improvements over Bernoulli encoding. Intermediate populations also improve attacked accuracy in selected settings, including simultaneous clean and adversarial gains for low-latency ResNet-19 models. The comparisons across architectures, horizons, and attacks reveal that the preferred population depends on the evaluation setting, rather than following a universal ordering. These empirical findings distinguish input-rate optimality from the robustness of the trained classifier.
- **Separation of training and inference encoding effects.** Frozen-checkpoint substitutions reveal that changing the encoder at inference is not equivalent to training with that encoder. In the evaluated CIFAR-10 binary substitutions, reducing the inference population improves mean clean accuracy but lowers PGD-7 accuracy, while increasing it has the reverse effect. Crossed training–inference comparisons further show that the observed ResNet-19 robustness advantage cannot be explained by inserting USHE into a Bernoulli-trained model at deployment. Encoding therefore has distinct training and inference roles, which our experiments evaluate separately.

2 Related Work

Input encoding and stochastic rate codes. Input encoding determines both the representation of static data and its temporal variability. Bernoulli rate encoding preserves pixel intensities in expectation but introduces finite-horizon sampling noise, affecting reconstruction and adversarial sensitivity (Sharmin et al., 2020; Mukhoty et al., 2024). Temporal codes represent information through spike times (Mostafa, 2018), while phase codes assign time-dependent weights to spikes (Kim et al., 2018). Direct encoding is common in short-horizon image classification: repeatedly supplying real-valued inputs avoids sampling error and integrates readily with surrogate-gradient training (Kundu et al., 2021; Kim et al., 2022). In a discretized single-spike latency code, short horizons instead restrict the available firing times; temporal coding can nevertheless offer sparser communication (Mostafa, 2018). This makes direct encoding attractive for low-latency benchmarks, at the cost of a nonbinary input interface. Within stochastic encoding, Mukhoty et al. (2024) connect Bernoulli

rate codes to randomized smoothing and derive certificates against ℓ_1-bounded input perturbations. RSC-SNN studies the accuracy–robustness trade-off through Gaussian input randomization (Wu et al., 2024). Signed rate encoding enables mean-centering through the alphabet $\{-1, 0, 1\}$; its Sparse Encoding Attack perturbs the encoded spikes directly (Mukhoty et al., 2025). These approaches distinguish changes to input statistics from changes to the spike alphabet, and empirical robustness from certification.

Robustness of spiking neural networks. Representative defenses modify training, neuronal dynamics, or input processing. HIRE-SNN trains low-latency direct-input networks with crafted input noise (Kundu et al., 2021), while SNN-RAT combines mixed adversarial training with regularization motivated by a spiking Lipschitz bound (Ding et al., 2022). HoSNN introduces homeostatic threshold adaptation to suppress perturbations of membrane dynamics (Geng & Li, 2023). Stochastic gating randomizes interlayer spike communication (Ding et al., 2024b), whereas FEEL-SNN combines frequency-selective input processing with learned, time-varying leak factors (Xu et al., 2024). STOD links temporal collinearity of parameter gradients under direct encoding to vulnerability and introduces orthogonal input kernels and structured constraints to diversify temporal features (Zhang et al., 2026). It is particularly relevant because it targets temporal structure, but through learned transformations rather than the sampling law of an unbiased binary code. For stochastic or nondifferentiable models, robustness evaluation must account for the active randomization and backward computation; BPDA and expectation over transformation provide established tools for addressing these difficulties (Athalye et al., 2018a;b).

Relation to USHE. USHE studies the joint temporal distribution of an unbiased binary input code. It preserves the $\{0, 1\}$ alphabet, each timestep's Bernoulli marginal, and expected input activity, while a latent population size controls temporal dependence and reconstructed-rate variance at fixed latency. Unlike signed encoding, Gaussian input randomization, or learned input transformations, this isolates temporal sampling dependence without changing the one-step input distribution. Each family member minimizes unbiased rate-reconstruction variance under a prescribed finite-exchangeability constraint, and independent Bernoulli encoding is recovered in the infinite-population limit. No learned encoder parameters, modified neuron dynamics, or specialized adversarial-training objective are required. USHE therefore complements training- and architecture-based defenses through a statistically characterized encoding family. Its guarantees concern input-rate reconstruction; classification accuracy and robustness remain properties of the trained network and evaluation protocol.

3 METHOD

Unbiased Spiking Hypergeometric Encoding (USHE) separates the number of SNN timesteps from the variability of the encoded rate. For an input $x \in [0, 1]^d$, the encoder emits $S \in \{0, 1\}^{T \times d}$ over a visible horizon T. Its additional parameter is a latent population size $N \geq T$ and we denote the resulting family member by USHE_N. Varying N changes the dependence among temporal spikes while preserving their binary alphabet and Bernoulli marginals. No additional network timesteps or learned encoder parameters are introduced.

3.1 AN UNBIASED FINITE-POPULATION CODE

Consider one intensity (say normalized input pixel value) $p \in [0, 1]$. An integer budget is needed to sample a binary population, but deterministic rounding of Np would bias the represented intensity. We instead use stochastic rounding (Croci et al., 2022):

$$k = \lfloor Np \rfloor, \qquad r_N = Np - k, \qquad K = k + B, \quad B \sim \mathrm{Bernoulli}(r_N). \tag{1}$$

Thus K equals one of the two adjacent integers and $\mathbb{E}[K] = Np$. For such chosen K, the idea is to uniformly permute K ones and $N - K$ zeros, and emit the first T entries as the spike train of length T. Equivalently, if we denote by s_t the spike emitted at time step t, and $S_t := \sum_{i=1}^{t} s_i$, with $S_0 = 0$, our spiking process is the following

$$s_t \mid K, s_{1:t-1} \sim \mathrm{Bernoulli}\left(\frac{K - S_{t-1}}{N - t + 1}\right), \qquad S_t = S_{t-1} + s_t, \quad 1 \leq t \leq T. \tag{2}$$

Input coordinates use independent budgets and sampling; timesteps within a coordinate are dependent.

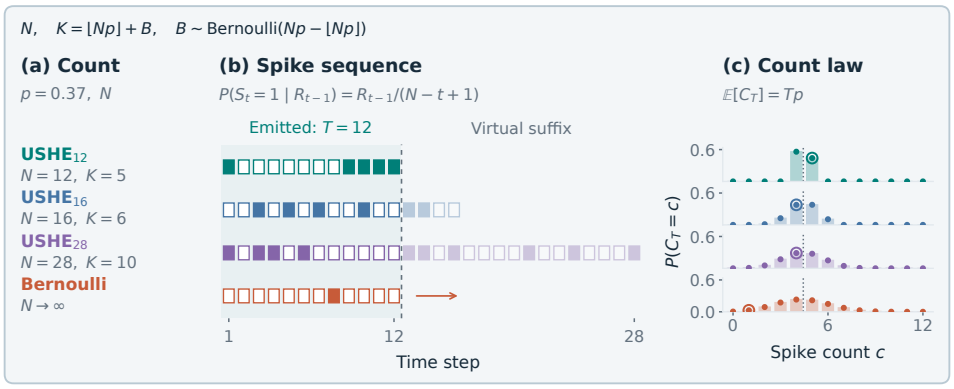


Figure 2: Mechanism of USHE: **(a)** USHE forms an unbiased integer spike budget in a latent population N. **(b)** Regardless of the population N USHE reveals only T samples without replacement. The visible horizon stays fixed while N controls temporal dependence, connecting minimum-variance rate estimation to independent Bernoulli encoding. **(c)** Varying N the spike count of produced spikes in latency T expresses variability controlled by the population size N.

For example, at $p = 1/2$ and $T = 4$, choosing $N = 4$ emits exactly two spikes in a random order. Choosing $N = 8$ instead samples four entries from a population with four ones and four zeros, so the visible count fluctuates while still averaging two. Both encoders run the SNN for four steps. The population parameter changes the uncertainty in the represented rate, not its mean or the visible computation horizon. We refer to Figure 2 for a visual explanation of the encoder mechanism.

3.2 SNN INTEGRATION AND COMPUTATION

Streaming implementation. Training uses sampled spike sequences and the network's usual surrogate-gradient backpropagation; the fixed input encoder introduces no parameters to optimize. Algorithm 1 tracks only the remaining integer budget per coordinate. It requires $O(Td)$ sampling work and $O(d)$ auxiliary state, excluding storage of the emitted sequence and SNN activations. A length-N tensor is never materialized. Increasing N therefore leaves the number of sampling transitions unchanged. The expected input spike count is $T\sum_i x_i$ for every member of the family; hidden-layer activity and hardware cost need not be identical.

Algorithm 1. Streaming USHE

Input: $p \in [0,1]^d$, $N \geq T \geq 1$
$a \leftarrow \lfloor Np \rfloor$, $r \leftarrow Np - a$
$B \sim \text{Bernoulli}(r)$ independently
$R \leftarrow a + B$
for $t = 1, \ldots, T$ **do**
  $\pi \leftarrow R/(N - t + 1)$
  $s_t \sim \text{Bernoulli}(\pi)$ independently
  $R \leftarrow R - s_t$
  emit s_t
end for

3.3 ANALYSIS

We fix an input $p \in [0,1]$ and integers $1 \leq T \leq N$. For any positive integer m, write $r_m = \{mp\}$ and $v_m = r_m(1 - r_m)$, where $\{u\} = u - \lfloor u \rfloor$. We denote the rate reconstructed by USHE_N at horizon T by

$$\widehat{p}_{T,N} = \frac{S_T}{T} = \frac{1}{T}\sum_{t=1}^{T} s_t.$$

Write $V_{T,N}(p) = \text{Var}(\widehat{p}_{T,N})$. Throughout this analysis, T is the visible horizon (latency) and N is the latent population size.

Proposition 3.1 (Unbiased marginals and exact variance)**.** For every $p \in [0,1]$, the USHE sequence is exchangeable, $\mathbb{E}[s_t] = p$, and $\mathbb{E}[\widehat{p}_{T,N}] = p$. For $N > 1$ we have

$$V_{T,N}(p) = \frac{(N-T)p(1-p)}{T(N-1)} + \frac{(T-1)v_N}{TN(N-1)}. \tag{3}$$

Hence, distinct timesteps have nonpositive covariance. For $T = N = 1$, the output is a single $\mathrm{Bernoulli}(p)$ draw and $V_{1,1}(p) = p(1-p)$.

The first term in equation 3 comes from the hypergometric nature of the USHE, while the second term accounts for the variability of the rounded budget; it vanishes when Np is an integer. Every individual spike still has variance $p(1-p)$. USHE therefore controls the variability of the reconstructed rate through temporal dependence, rather than by changing the marginal randomness of a spike. Proposition 3.4 also shows motivation behind the name USHE.

A variance-minimization problem. To explain the role of the population size, consider all laws Q on binary vectors $X_{1:N}$ that are exchangeable and satisfy $\mathbb{E}_Q[X_i] = p$. We denote this class by $\mathcal{P}_N(p)$. Exchangeability means that permuting temporal positions leaves the joint law unchanged. In particular, it does not require independence or a fixed total count. For a visible prefix of length T, and $Q \in \mathcal{P}_N(p)$, we define

$$\widehat{p}_T(X) = \frac{1}{T}\sum_{t=1}^{T} X_t, \qquad R_T(Q;p) = \mathbb{E}_Q\left[(\widehat{p}_T(X) - p)^2\right]. \tag{4}$$

Our optimization problem is $\min_{Q\in\mathcal{P}_N(p)} R_T(Q;p)$. Thus the encoder may choose any unbiased distribution of the latent total count, not only the stochastic-rounding construction in equation 1.

Theorem 3.2 (Optimality at a prescribed population size)**.** For $p \in [0,1]$ and $2 \le T \le N$,

$$\min_{Q\in\mathcal{P}_N(p)} R_T(Q;p) = V_{T,N}(p). \tag{5}$$

The unique minimizing latent law, denoted by $Q_N^\star$, samples $K = \lfloor Np \rfloor + B$ with $B \sim \mathrm{Bernoulli}(r_N)$ and, conditional on K, places the K spikes uniformly among the N positions. Its first T coordinates have exactly the law of USHE_N at horizon T. The same $Q_N^\star$ minimizes the risk simultaneously for every $1 \le T \le N$. At $T = 1$, all admissible visible laws are $\mathrm{Bernoulli}(p)$, so this objective does not distinguish their latent extensions.

The reason is an exact decomposition. If $C = \sum_{i=1}^{N} X_i$, then for $Q \in \mathcal{P}_N(p)$ and $N > 1$,

$$R_T(Q;p) - V_{T,N}(p) = \frac{T-1}{TN(N-1)}\big(\mathrm{Var}_Q(C) - v_N\big). \tag{6}$$

The integer constraint and $\mathbb{E}_Q[C] = Np$ imply $\mathrm{Var}_Q(C) \ge v_N$, with equality precisely for the rounded budget in equation 1. Consequently, stochastic rounding and uniform temporal allocation arise as the solution to the optimization problem, rather than as additional assumptions on its count distribution.

Corollary 3.3 (Global optimality at the endpoint)**.** Let $X_{1:T}$ be any binary sequence satisfying $\mathbb{E}[\widehat{p}_T(X)] = p$, without assuming exchangeability or unbiased individual timesteps. Then

$$\mathrm{Var}(\widehat{p}_T(X)) \ge \frac{v_T}{T^2} = V_{T,T}(p). \tag{7}$$

Equality holds if and only if $\sum_{t=1}^{T} X_t \in \{\lfloor Tp \rfloor, \lceil Tp \rceil\}$ almost surely. Thus USHE_T is globally variance-optimal for unbiased rate reconstruction for a fixed latency T.

The optimal count distribution is unique, but its temporal ordering need not be. USHE chooses its exchangeable realization. For $N > T$, optimality is instead constrained by compatibility with an exchangeable population of length N; without this requirement, the global minimum remains v_T/T^2.

Corollary 3.4 (Ordered optimal risks and the Bernoulli limit)**.** For fixed T, p and integers $T \le N \le M$,

$$\frac{v_T}{T^2} = V_{T,T}(p) \le V_{T,N}(p) \le V_{T,M}(p) \le \frac{p(1-p)}{T}. \tag{8}$$

Moreover, at fixed T, the joint USHE law satisfies

$$d_{\mathrm{TV}}\big(\mathcal{L}(s_{1:T}), \mathrm{Bernoulli}(p)^{\otimes T}\big) \longrightarrow 0 \qquad \text{as } N \longrightarrow \infty. \tag{9}$$

Indeed, an exchangeable M-step population restricts to an admissible N-step population whenever $M \geq N$. Requiring a longer extension therefore imposes a stronger constraint, so its optimal risk cannot decrease. The ordering need not be strict. Independent Bernoulli encoding is feasible for every population size and is recovered in the infinite-population limit.

All proofs are in Appendix A.

Beyond squared error. The same USHE law minimizes every convex function of the reconstructed rate within $\mathcal{P}_N(p)$, including absolute error and $|\widehat{p}_T - p|^q$ for $q \geq 1$. Appendix A.6 gives the full convex-order statement and its proof; this strengthening does not follow from variance ordering alone. Appendix A.7 also gives a minimax interpretation without assuming exchangeability: USHE minimizes the worst-case rate variance over all choices of T positions in a length-N binary population with unbiased marginals.

4 Experiments

4.1 Experimental setting

We evaluate USHE on CIFAR-10 and CIFAR-100 Krizhevsky & Hinton (2009) using the VGG-SNN and ResNet-19 architectures adopted in TET Deng et al. (2022). We consider low-latency inference with $T \in \{4, 8\}$ time steps and obtain predictions by averaging the output logits over time. USHE is compared primarily with Bernoulli rate encoding, with direct encoding included where relevant. We denote by USHE(N) a configuration with encoder population size $N \geq T$.

For each dataset, architecture, and latency, all encoder variants use the same preprocessing, network configuration, and training protocol. Unless stated otherwise, we report top-1 test accuracy using a single stochastic encoding sample per input. Results are averaged over three evaluation seeds obtained from one trained checkpoint per configuration; consequently, the reported variation reflects stochastic evaluation rather than variation across independently trained models. Experiment-specific evaluation settings are introduced in the corresponding subsections, while complete training configurations, random seeds, attack definitions, and implementation details are provided in Appendix B.

4.2 Clean accuracy

We first isolate the effect of replacing independent Bernoulli encoding with USHE. Table 1 reports the minimal-population configuration, $N = T$, which is fixed across all datasets, architectures, and temporal horizons. The complete population-size sweep is provided in Appendix C.1.

At matched temporal horizons, USHE improves clean accuracy in all eight comparisons. The gains range from 3.40 to 6.72 percentage points and are observed for both architectures and datasets. The improvement is particularly pronounced at $T = 4$: averaged across the four dataset–architecture pairs, USHE gains 6.32 percentage points at $T = 4$, compared with 4.73 points at $T = 8$.

Notably, USHE with $T = 4$ also exceeds Bernoulli encoding with $T = 8$ in every dataset–architecture pair, by 0.85–2.06 percentage points. Thus, within stochastic rate encoding, USHE provides higher clean accuracy while using half as many SNN simulation steps. Deterministic direct encoding is considered separately together with its robustness trade-off in Section 4.3.

Table 1: Clean top-1 test accuracy (%) under Bernoulli encoding and USHE with $N = T$. The best result in each matched setting is shown in bold. Δ denotes USHE minus Bernoulli in percentage points. Results represent one training seed per configuration.

T	Bernoulli	USHE ($N = T$)	Δ (pp)
CIFAR-10, VGG-SNN			
4	79.25	**84.96**	+5.71
8	82.90	**86.30**	+3.40
CIFAR-10, ResNet-19			
4	79.18	**85.56**	+6.38
8	83.73	**88.31**	+4.58
CIFAR-100, VGG-SNN			
4	51.53	**58.01**	+6.48
8	56.42	**61.33**	+4.91
CIFAR-100, ResNet-19			
4	51.59	**58.31**	+6.72
8	57.46	**63.47**	+6.01

4.3 Robustness properties of USHE

Evaluation protocol. We evaluate input-space adversarial robustness on CIFAR-10 and CIFAR-100 using VGG-SNN and ResNet-19 at matched training and inference horizons $T \in \{4, 8\}$. For

Table 2: Comparison between direct, Bernoulli and USHE encodings at latency $T = 4$ (top-1 accuracy, %). Means are followed by standard deviations in gray superscripts (except for APGD CE). Here, the subscript in USHE_N denotes population size N. Bold marks the largest displayed available mean in each dataset–architecture setting, including ties. Evaluation-seed counts are given in the appendix.

	CIFAR-10						CIFAR-100					
Encoder	**Clean**	**FGSM**	**RFGSM**	**PGD 7**	**APGD CE**	**GN**	**Clean**	**FGSM**	**RFGSM**	**PGD 7**	**APGD CE**	**GN**
VGG-SNN												
Direct	**93.72**$^{0.00}$	12.24$^{0.00}$	18.70$^{0.07}$	0.05$^{0.01}$	00.00	**86.28**$^{0.09}$	**74.48**$^{0.00}$	4.62$^{0.00}$	4.92$^{0.03}$	0.02$^{0.01}$	00.00	57.11$^{0.20}$
Bernoulli	79.31$^{0.10}$	**43.52**$^{0.63}$	63.00$^{0.38}$	**38.13**$^{0.05}$	**35.05**	79.17$^{0.04}$	51.52$^{0.11}$	**21.60**$^{0.06}$	35.11$^{0.18}$	**18.26**$^{0.35}$	**16.75**	51.50$^{0.11}$
USHE_4	84.95$^{0.23}$	38.67$^{0.24}$	63.73$^{0.18}$	30.86$^{0.20}$	25.89	84.76$^{0.17}$	58.00$^{0.09}$	17.95$^{0.10}$	35.19$^{0.13}$	14.03$^{0.19}$	11.82	**57.68**$^{0.36}$
USHE_8	81.86$^{0.11}$	42.74$^{0.16}$	**64.23**$^{0.31}$	36.05$^{0.28}$	32.45	81.56$^{0.12}$	53.90$^{0.05}$	20.39$^{0.39}$	**35.21**$^{0.20}$	16.52$^{0.17}$	14.18	53.75$^{0.52}$
USHE_{12}	81.02$^{0.12}$	43.42$^{0.47}$	63.69$^{0.66}$	36.89$^{0.27}$	33.57	81.01$^{0.07}$	53.41$^{0.12}$	20.73$^{0.16}$	**35.21**$^{0.13}$	17.22$^{0.04}$	15.38	53.75$^{0.41}$
ResNet-19												
Direct	**93.96**$^{0.00}$	15.00$^{0.00}$	15.57$^{0.12}$	0.00$^{0.01}$	00.00	83.22$^{0.04}$	**76.87**$^{0.00}$	9.26$^{0.00}$	8.28$^{0.14}$	0.03$^{0.01}$	00.00	57.71$^{0.16}$
Bernoulli	79.17$^{0.02}$	51.01$^{0.24}$	66.15$^{0.22}$	44.89$^{0.39}$	41.66	79.08$^{0.16}$	51.57$^{0.20}$	28.13$^{0.50}$	39.40$^{0.31}$	24.16$^{0.18}$	22.23	51.27$^{0.19}$
USHE_4	85.52$^{0.14}$	48.87$^{0.07}$	**69.26**$^{0.46}$	40.54$^{0.21}$	35.11	**85.28**$^{0.04}$	58.26$^{0.14}$	26.73$^{0.29}$	**40.92**$^{0.21}$	21.37$^{0.08}$	18.45	**58.06**$^{0.20}$
USHE_8	81.73$^{0.03}$	52.87$^{0.55}$	68.91$^{0.33}$	**46.41**$^{0.45}$	42.71	81.61$^{0.28}$	54.02$^{0.13}$	29.10$^{0.51}$	40.77$^{0.09}$	24.91$^{0.13}$	22.52	53.94$^{0.18}$
USHE_{12}	80.99$^{0.11}$	**52.89**$^{0.16}$	68.20$^{0.09}$	46.39$^{0.49}$	**42.92**	80.62$^{0.02}$	52.77$^{0.22}$	**29.16**$^{0.02}$	40.28$^{0.26}$	**24.97**$^{0.08}$	**22.56**	52.78$^{0.09}$

stochastic encoders, predictions use one sampled encoding per input ($m = 1$), with output logits averaged over time. The completed evaluations comprise the fast gradient sign method (FGSM), which constructs a single gradient-sign perturbation (Goodfellow et al., 2015), and seven-step projected gradient descent (PGD-7), which iteratively updates and projects the perturbation (Madry et al., 2018). We are extending this evaluation with RFGSM, which introduces random initialization before gradient-sign updates (Tramèr et al., 2018), and APGD-CE, which uses Auto-PGD with a cross-entropy objective and adaptive step-size selection (Croce & Hein, 2020). These additional evaluations examine whether the comparisons observed under FGSM and PGD-7 persist under alternative attack procedures. We also report accuracy under Gaussian corruption (GN) with standard deviation $\sigma = 8/255$, separately from adversarial accuracy.

Each configuration is trained with one training seed and evaluated using a fixed checkpoint. Then, all the models are tested for three evaluation seeds (except for APGD-CE which was evaluated using one seed) with reported mean and standard deviation in superscript. The reported deviations therefore quantify evaluation variability for the same trained model, rather than variability across independently trained models.

Architecture- and attack-dependent robustness. At $T = 4$ (Table 2), increasing the population from $N = T$ to $2T$ or $3T$ improves FGSM and PGD-7 accuracy in all four dataset–architecture settings. Both intermediate populations outperform Bernoulli on ResNet-19, whereas Bernoulli retains higher PGD-7 accuracy on VGG-SNN. The available APGD-CE results show the same architecture dependence. RFGSM, however, gives a different ranking: all three displayed USHE variants exceed Bernoulli in mean accuracy on both architectures and datasets. For CIFAR-10/VGG-SNN, USHE_8 reaches 64.23% versus 63.00% for Bernoulli, despite trailing under FGSM, PGD-7, and APGD-CE. The corresponding RFGSM gains on CIFAR-100/VGG-SNN are smaller (0.08–0.10 percentage points). Thus, encoder rankings depend on the attack as well as the architecture.

Dependence on latency. At $T = 8$ (see Appendix C.1), Bernoulli exceeds all evaluated finite populations under FGSM and PGD-7. This ordering does not extend uniformly to RFGSM: on CIFAR-10/ResNet-19, USHE_{24} achieves 69.90% versus 69.78% for Bernoulli, although this difference is small relative to evaluation variability. USHE's advantages are therefore most consistent across attacks for ResNet-19 at $T = 4$, but are not confined to that setting.

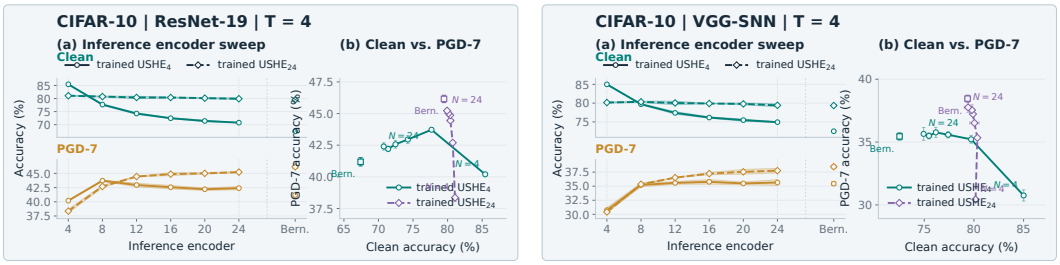


Figure 3: **Inference-time traversal of the USHE family.** USHE_4- and USHE_{24}-trained checkpoints are frozen while only the inference encoder is varied, for ResNet-19 and VGG-SNN at $T = 4$. **Left plots:** clean and PGD-7 accuracy across inference encoders. **Right plots:** the corresponding clean–PGD-7 trajectories. Bernoulli is shown as the limiting reference.

Direct encoding and Gaussian corruption. Direct encoding achieves the highest clean accuracy but falls below 0.1% under PGD-7 in every evaluated setting. For stochastic encoders, Gaussian-corruption accuracy remains close to clean accuracy at the tested noise level.

4.4 From family-level behavior to inference-time effects

Population controls the operating point. We first fix the latency $T = 4$ and train and evaluate matched USHE_N models while varying N. This isolates changes across the encoder family without increasing inference time and provides an empirical counterpart to the Bernoulli limit as N increases. Figure 4 shows that the finite-population regime is non-trivial: clean and Gaussian-noise accuracy approach the Bernoulli reference, whereas FGSM and PGD-7 exhibit pronounced intermediate-N behavior and can exceed the Bernoulli baseline. Thus, N controls distinct clean–robustness operating points at fixed latency rather than merely determining proximity to Bernoulli.

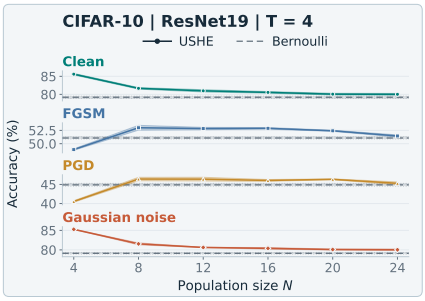


Figure 4: **Accuracy across the USHE family at fixed latency.** Models are trained and evaluated with matched USHE_N encoders at $T = 4$; dashed lines show the corresponding Bernoulli reference.

Separating training and inference effects. The matched sweep changes N during both training and inference. To separate these effects, we freeze checkpoints trained with USHE_4 and USHE_{24} and vary only the inference encoder, regenerating attacks against each resulting classifier. Figure 3 reveals a consistent directional trade-off across VGG-SNN and ResNet-19: moving toward smaller inference populations generally improves clean accuracy while reducing PGD-7 accuracy, whereas larger populations reverse this tendency. The clean–PGD trajectories consequently move predominantly along a trade-off frontier rather than toward simultaneous improvements.

The trajectories also depend on the encoder used during training: the same inference encoder can produce different behavior for USHE_4- and USHE_{24}-trained checkpoints. Hence, N plays distinct roles during optimization and inference, and matched-model robustness differences cannot generally be attributed to the inference encoder alone. We refer to Appendix C.3 for additional results around this topic.

4.5 Comparison with existing robust SNN methods

Baselines. We compare USHE with methods that enhance robustness through training, neuron dynamics, or input transformations. The baselines comprise a standard SNN; adversarial training inspired by HIRE-SNN (Kundu et al., 2021); dynamic LIF neurons (DLIF) (Ding et al., 2024a); homeostatic threshold adaptation (HoSNN) (Geng & Li, 2023); frequency encoding with evolutionary leak factors (FEEL-SNN) (Xu et al., 2024); further with stochastic gating of spike communication

(StoG) (Ding et al., 2024b); and structured temporal orthogonal decorrelation (STOD) (Zhang et al., 2026). For STOD, we distinguish inference with and without orthogonal kernels. These methods modify different components of the network or its training, whereas USHE changes the joint temporal distribution of a binary input code without introducing learned encoder parameters or a specialized adversarial-training objective.

Comparison with robust SNN methods. Direct comparison is complicated by differences in architectures, simulation horizons, and evaluation protocols. Table 3 therefore compiles representative methods at $T \in \{4, 5, 8\}$, distinguishing published results from our reproductions, marked by *. This provides context for USHE's performance rather than a fully controlled comparison across all methods.

At $T = 4$, USHE achieves the highest PGD accuracy among the listed methods in all four dataset–architecture blocks, and the highest FGSM accuracy in three. The advantage is particularly pronounced against the reproduced ResNet-19 baselines: USHE reaches 46.39% and 24.97% PGD accuracy on CIFAR-10 and CIFAR-100, compared with their respective best results of 10.44% and 2.83%. On CIFAR-10/VGG, however, HoSNN retains higher FGSM accuracy. The real-valued-input baselines also generally achieve substantially higher clean accuracy. USHE thus provides competitive attacked accuracy with a binary input interface and four simulation steps, but represents a different clean–robustness trade-off rather than a uniform improvement.

Table 3: Comparison with robust SNN methods on CIFAR-10 and CIFAR-100. All entries report top-1 accuracy (%). * denotes our reproduced results.

Method	T	Input	Clean	FGSM	PGD
CIFAR-10 | VGG-11					
AT (Kundu et al., 2021)	4	Real	91.16	38.20	14.07
DLIF (Ding et al., 2024a)	8	Real	92.22	13.24	0.09
HoSNN (Geng & Li, 2023)	5	Real	92.43	**54.76**	15.32
FEEL (Xu et al., 2024)	8	Real	**93.29**	44.96	28.35
StoG (Ding et al., 2024b)	8	Real	91.64	16.22	0.28
STOD* w/ OK (Zhang et al., 2026)	4	Real	92.51	16.65	2.73
STOD* w/o OK (Zhang et al., 2026)	4	Real	76.15	13.15	3.37
USHE$_{12}$	4	Binary	81.02	43.42	**36.89**
CIFAR-10 | ResNet-19					
STOD* w/ OK	4	Real	**93.07**	20.52	1.54
STOD* w/o OK	4	Real	87.36	17.68	2.07
HoSNN*	4	Real	85.96	37.81	10.44
USHE$_{12}$ (ours)	4	Binary	80.99	**52.89**	**46.39**
CIFAR-100 | VGG-11					
SNN	4	Real	72.39	4.55	0.19
AT (Kundu et al., 2021)	4	Real	69.69	16.31	8.49
DLIF (Ding et al., 2024a)	8	Real	70.79	6.95	0.08
HoSNN (Geng & Li, 2023)	5	Real	71.98	13.48	0.19
FEEL (Xu et al., 2024)	8	Real	**73.79**	9.60	2.04
StoG (Ding et al., 2024b)	8	Real	70.44	8.27	0.49
STOD* w/ OK (Zhang et al., 2026)	4	Real	71.88	6.45	1.28
STOD* w/o OK (Zhang et al., 2026)	4	Real	60.91	6.02	1.38
USHE$_{12}$	4	Binary	53.41	**20.73**	**17.22**
CIFAR-100 | ResNet-19					
STOD* w/ OK	4	Real	**74.61**	12.17	1.76
STOD* w/o OK	4	Real	63.96	11.54	2.83
USHE$_{12}$	4	Binary	52.77	**29.16**	**24.97**

5 Limitations

Despite improving the accuracy–robustness trade-off of stochastic rate encoding, USHE still trails direct encoding in clean accuracy, reflecting a remaining cost of binary temporal representation. Moreover, robustness is not uniform across the USHE family: the population N selects different operating points, and no single member dominates across clean and adversarial conditions. Our experiments focus primarily on short-latency image classification and a limited set of architectures and attacks; broader tasks, longer simulation horizons, and adaptive robustness evaluations remain to be studied.

6 Conclusion

We introduced the USHE family, a population-controlled class of stochastic spike encoders that interpolates between finite-population temporal sampling and Bernoulli encoding while preserving fixed inference latency. Our analysis characterizes its distributional properties, and experiments show that the population size N provides a meaningful control over the clean–robustness operating point of SNNs. Across architectures and datasets, finite USHE populations can improve upon Bernoulli encoding, while controlled encoder substitutions reveal distinct roles for the encoding used during training and inference. These results establish population-controlled temporal dependence as a useful design dimension for spike encoding beyond independent Bernoulli sampling.

REFERENCES

Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 274–283. PMLR, 2018a. URL https://proceedings.mlr.press/v80/athalye18a.html.

Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 284–293. PMLR, 2018b. URL https://proceedings.mlr.press/v80/athalye18b.html.

Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 2206–2216. PMLR, 2020. URL https://proceedings.mlr.press/v119/croce20b.html.

Matteo Croci, Massimiliano Fasi, Nicholas J. Higham, Theo Mary, and Mantas Mikaitis. Stochastic rounding: implementation, error analysis and applications. *Royal Society Open Science*, 9(3): 211631, 2022. doi: 10.1098/rsos.211631.

Shikuang Deng, Yuhang Li, Shanghang Zhang, and Shi Gu. Temporal efficient training of spiking neural network via gradient re-weighting. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=_XNtisL32jv.

Jianhao Ding, Tong Bu, Zhaofei Yu, Tiejun Huang, and Jian Liu. Snn-rat: Robustness-enhanced spiking neural network through regularized adversarial training. *Advances in Neural Information Processing Systems*, 35:24780–24793, 2022.

Jianhao Ding, Zhiyu Pan, Yujia Liu, Zhaofei Yu, and Tiejun Huang. Robust stable spiking neural networks. *arXiv preprint arXiv:2405.20694*, 2024a.

Jianhao Ding, Zhaofei Yu, Tiejun Huang, and Jian K Liu. Enhancing the robustness of spiking neural networks with stochastic gating mechanisms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38(1), pp. 492–502, 2024b.

Wei Fang, Zhaofei Yu, Yanqi Chen, Tiejun Huang, Timothée Masquelier, and Yonghong Tian. Deep residual learning in spiking neural networks. *Advances in Neural Information Processing Systems*, 34:21056–21069, 2021.

Hejia Geng and Peng Li. Hosnn: Adversarially-robust homeostatic spiking neural networks with adaptive firing thresholds. *arXiv preprint arXiv:2308.10373*, 2023.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015. URL https://arxiv.org/abs/1412.6572.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Jaehyun Kim, Heesu Kim, Subin Huh, Jinho Lee, and Kiyoung Choi. Deep neural networks with weighted spikes. *Neurocomputing*, 311:373–386, 2018. doi: 10.1016/j.neucom.2018.05.087.

Youngeun Kim, Hyoungseob Park, Abhishek Moitra, Abhiroop Bhattacharjee, Yeshwanth Venkatesha, and Priyadarshini Panda. Rate coding or direct coding: Which one is better for accurate, robust, and energy-efficient spiking neural networks? In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 71–75. IEEE, 2022.

Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. *https://www.cs.toronto.edu/ kriz/cifar.html*, 2009.

Souvik Kundu, Massoud Pedram, and Peter A Beerel. Hire-snn: Harnessing the inherent robustness of energy-efficient deep spiking neural networks by training with crafted input noise. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5189–5198. IEEE, 2021.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=rJzIBfZAb.

Hesham Mostafa. Supervised learning based on temporal coding in spiking neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 29(7):3227–3235, 2018. doi: 10.1109/TNNLS.2017.2726060.

Bhaskar Mukhoty, Hilal AlQuabeh, Giulia De Masi, Huan Xiong, and Bin Gu. Certified adversarial robustness for rate encoded spiking neural networks. In *The Twelfth International Conference on Learning Representations*, 2024.

Bhaskar Mukhoty, Hilal AlQuabeh, and Bin Gu. Improving generalization and robustness in SNNs through signed rate encoding and sparse encoding attacks. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=qLh6Ufvnuc.

Emre O Neftci, Hesham Mostafa, and Friedemann Zenke. Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks. *IEEE Signal Processing Magazine*, 36(6):51–63, 2019.

Saima Sharmin, Nitin Rathi, Priyadarshini Panda, and Kaushik Roy. Inherent adversarial robustness of deep spiking neural networks: Effects of discrete input encoding and non-linear activations. In *Computer Vision – ECCV 2020*, volume 12374 of *Lecture Notes in Computer Science*, pp. 399–414. Springer, 2020. doi: 10.1007/978-3-030-58526-6_24.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. In *International Conference on Learning Representations*, 2018. URL https://arxiv.org/abs/1705.07204.

Keming Wu, Man Yao, Yuhong Chou, Xuerui Qiu, Rui Yang, Bo Xu, and Guoqi Li. RSC-SNN: Exploring the trade-off between adversarial robustness and accuracy in spiking neural networks via randomized smoothing coding. In *Proceedings of the 32nd ACM International Conference on Multimedia*, 2024. doi: 10.1145/3664647.3680639. URL https://arxiv.org/abs/2407.20099.

Mengting Xu, De Ma, HuaJin Tang, Qian Zheng, and Gang Pan. Feel-snn: Robust spiking neural networks with frequency encoding and evolutionary leak factor. *Advances in Neural Information Processing Systems*, 37:91930–91950, 2024.

Desong Zhang, Jia Hu, and Geyong Min. Breaking gradient temporal collinearity for robust spiking neural networks. In *The Fourteenth International Conference on Learning Representations*, 2026.

A PROOFS OF THEORETICAL PROPERTIES OF USHE

We use the notation from Section 3.3. In particular, $r_m = \{mp\}$, $v_m = r_m(1 - r_m)$, and $\widehat{p}_{T,N} = S_T/T$ is the USHE rate at visible horizon T and latent population size N. Let $Q_N^\star$ denote the full length-N USHE population law. The algorithm need not generate this whole vector: its first T coordinates are sampled by the streaming rule equation 2.

A.1 TWO ELEMENTARY FACTS

Lemma A.1 (An unbiased integer budget)**.** Let $C \in \{0, \dots, N\}$ satisfy $\mathbb{E}[C] = \mu$. Write $k = \lfloor \mu \rfloor$, $r = \mu - k$, and $K = k + B$, where $B \sim \mathrm{Bernoulli}(r)$. Then

$$\mathrm{Var}(C) - r(1 - r) = \mathbb{E}[(C - k)(C - k - 1)] \geq 0. \tag{10}$$

Equality holds if and only if C has the same law as K. Furthermore, if $f : \{0, \dots, N\} \to \mathbb{R}$ has nonnegative second finite differences, then

$$\mathbb{E}[f(K)] \leq \mathbb{E}[f(C)]. \tag{11}$$

Proof. Since $\mathbb{E}[C - k] = r$, expanding the product gives

$$\mathbb{E}[(C - k)(C - k - 1)] = \mathrm{Var}(C) + r^2 - r.$$

For integer C, the product is nonnegative, and it vanishes precisely when $C \in \{k, k + 1\}$. The mean constraint then fixes the probabilities to $1 - r$ and r. When $r = 0$, it forces $C = k$ almost surely, including the boundary case $\mu = N$. This proves equation 10 and its equality condition.

For the convex statement, let $\widetilde{f}$ be the piecewise-affine interpolation of f on $[0, N]$. Nonnegative second differences mean that its consecutive slopes are nondecreasing, so $\widetilde{f}$ is convex. Jensen's inequality yields

$$\mathbb{E}[f(C)] = \mathbb{E}[\widetilde{f}(C)] \geq \widetilde{f}(\mu) = \mathbb{E}[f(K)].$$

At an integer μ, the final equality is simply $\widetilde{f}(\mu) = f(\mu)$. □

Lemma A.2 (Exchangeable binary populations)**.** Let $N > 1$, $Q \in \mathcal{P}_N(p)$, and $C = \sum_{i=1}^N X_i$. Conditional on $C = c$, the vector $X_{1:N}$ is uniform over binary vectors with c ones. For $1 \leq T \leq N$,

$$R_T(Q; p) = \frac{(N - T)p(1 - p)}{T(N - 1)} + \frac{T - 1}{TN(N - 1)} \mathrm{Var}_Q(C). \tag{12}$$

For distinct i, j,

$$\mathrm{Cov}_Q(X_i, X_j) = \frac{\mathrm{Var}_Q(C) - Np(1 - p)}{N(N - 1)}. \tag{13}$$

Proof. Any two binary vectors with the same total count are related by a coordinate permutation. Exchangeability assigns them equal probabilities, proving the conditional-uniformity claim whenever $Q(C = c) > 0$. Hence $\sum_{t=1}^T X_t \mid C = c$ counts a uniform sample of T positions from a population with c ones and $N - c$ zeros.

For the moment identities, exchangeability and binary values give

$$\mathbb{E}_Q[X_i] = p, \qquad \mathbb{E}_Q[X_i X_j] = \frac{\mathbb{E}_Q[C(C - 1)]}{N(N - 1)} \quad (i \neq j).$$

Using $\mathbb{E}_Q[C] = Np$ and $\mathbb{E}_Q[C(C - 1)] = \mathrm{Var}_Q(C) + N^2p^2 - Np$ gives equation 13. Since $\widehat{p}_T(X)$ is unbiased,

$$\begin{aligned} R_T(Q; p) &= \mathrm{Var}_Q(\widehat{p}_T(X)) \\ &= \frac{p(1 - p)}{T} + \frac{T - 1}{T} \frac{\mathrm{Var}_Q(C) - Np(1 - p)}{N(N - 1)}, \end{aligned}$$

which rearranges to equation 12. □

A.2 Unbiased marginals and exact variance

Proof of Proposition 3.1. Conditional on the rounded budget K, sampling without replacement is equivalent to revealing a uniformly permuted population with K ones. Indeed, after observing $s_{1:t-1}$, exactly $K - S_{t-1}$ ones remain among $N - t + 1$ unobserved positions, which gives equation 2. Thus USHE is a prefix of an exchangeable binary population. Moreover,

$$\mathbb{E}[s_t \mid K] = \frac{K}{N}, \qquad \mathbb{E}[K] = Np, \qquad \mathrm{Var}(K) = v_N.$$

Taking expectations proves $\mathbb{E}[s_t] = p$ and $\mathbb{E}[\widehat{p}_{T,N}] = p$. Substituting $\mathrm{Var}(K) = v_N$ into Lemma A.2 gives equation 3 and

$$\mathrm{Cov}(s_i, s_j) = \frac{v_N - Np(1-p)}{N(N-1)}, \qquad i \neq j. \tag{14}$$

To see that this is nonpositive, apply Lemma A.1 to $C \sim \mathrm{Binomial}(N, p)$, which has the same mean Np. It gives $v_N \leq \mathrm{Var}(C) = Np(1-p)$. For $T = N = 1$, the rounded budget is itself $\mathrm{Bernoulli}(p)$ and the only emitted spike equals that budget. □

The variance formula also retains the following input-uniform bound from the main construction. For $N > 1$, $p(1-p) \leq 1/4$ and $v_N \leq 1/4$ imply

$$\begin{aligned} V_{T,N}(p) &\leq \frac{N(N-T) + (T-1)}{4TN(N-1)} \\ &= \frac{N - T + 1}{4TN}. \end{aligned} \tag{15}$$

The same bound holds for $T = N = 1$, since then $V_{1,1}(p) = p(1-p) \leq 1/4$. This bound is not asserted to be attained for every pair (T, N).

A.3 Optimality at a prescribed population size

Proof of Theorem 3.2. Let $Q \in \mathcal{P}_N(p)$ and set $C = \sum_{i=1}^{N} X_i$. Since $\mathbb{E}_Q[C] = Np$, Lemma A.1 gives $\mathrm{Var}_Q(C) \geq v_N$. Together with Lemma A.2, this yields

$$R_T(Q; p) \geq \frac{(N-T)p(1-p)}{T(N-1)} + \frac{(T-1)v_N}{TN(N-1)} = V_{T,N}(p).$$

The USHE population law $Q_N^\star$ belongs to $\mathcal{P}_N(p)$ and has $\mathrm{Var}(K) = v_N$, so the lower bound is attained. Subtracting its risk establishes equation 6. More explicitly, for $k = \lfloor Np \rfloor$,

$$R_T(Q; p) - V_{T,N}(p) = \frac{T-1}{TN(N-1)} \mathbb{E}_Q[(C-k)(C-k-1)]. \tag{16}$$

For $T \geq 2$, the coefficient is strictly positive. Equality therefore holds precisely when C has the law of the rounded budget K in equation 1. Conditional uniformity from Lemma A.2 then fixes the entire latent law:

$$Q(X_{1:N} = x) = \frac{Q(C = \sum_i x_i)}{\binom{N}{\sum_i x_i}}, \qquad x \in \{0,1\}^N.$$

Hence the minimizing latent law is unique and equals $Q_N^\star$. Its first T coordinates follow the USHE construction, as established in the proof of Proposition 3.1.

The minimizer of the count variance does not depend on T. Consequently, one and the same population law minimizes every prefix risk with $2 \leq T \leq N$. For $T = 1$, all admissible laws have $R_1(Q; p) = p(1-p)$, so $Q_N^\star$ is also optimal there, without a uniqueness claim for its latent extension at that single horizon. □

The theorem can equivalently be stated for the visible law. Define

$$\mathcal{E}_{T,N}(p) = \{\mathcal{L}_Q(X_{1:T}) : Q \in \mathcal{P}_N(p)\}. \tag{17}$$

Then the USHE prefix is the unique variance-minimizing law in $\mathcal{E}_{T,N}(p)$ for $T \geq 2$. Indeed, any minimizing visible law has an exchangeable extension to length N by definition; its risk equals that of its extension, forcing that extension to be $Q_N^\star$ by the theorem. For $T = 1$, the visible class contains only $\mathrm{Bernoulli}(p)$. This equivalence does not impose a particular latent count distribution on competing encoders.

A.4 GLOBAL OPTIMALITY AT THE ENDPOINT

Proof of Corollary 3.3. For an arbitrary binary vector $X_{1:T}$, set $C_T = \sum_{t=1}^{T} X_t$. The assumed unbiasedness of its reconstructed rate gives $\mathbb{E}[C_T] = Tp$. Lemma A.1, now applied on $\{0, \dots, T\}$, yields

$$\mathrm{Var}(\widehat{p}_T(X)) = \frac{\mathrm{Var}(C_T)}{T^2} \geq \frac{v_T}{T^2}.$$

Equality holds precisely when C_T has the law of the rounded budget at population size T, equivalently when it is supported on $\{\lfloor Tp \rfloor, \lceil Tp \rceil\}$ under the mean constraint. At $N = T$, USHE emits its whole budget, so $S_T = K$ and attains this bound. No step in this argument requires exchangeability or unbiasedness of each individual X_t. Only the total count distribution is determined; different conditional orderings can attain the same risk. □

The endpoint is also optimal for every convex rate loss over this unrestricted class. If $\phi : [0, 1] \to \mathbb{R}$ is convex, the sequence $f(c) = \phi(c/T)$ is discretely convex on $\{0, \dots, T\}$. Lemma A.1 therefore gives

$$\mathbb{E}[\phi(\widehat{p}_{T,T})] \leq \mathbb{E}[\phi(\widehat{p}_T(X))]. \tag{18}$$

Unlike squared error, an arbitrary convex loss need not have a unique minimizing count distribution; an affine loss, for example, is equal for all admissible encoders.

A.5 ORDERED RISKS AND THE BERNOULLI LIMIT

Proof of Corollary 3.4. For $T \leq N \leq M$, restricting an exchangeable length-M vector to its first N coordinates gives a law in $\mathcal{P}_N(p)$. Consequently,

$$\mathcal{E}_{T,M}(p) \subseteq \mathcal{E}_{T,N}(p). \tag{19}$$

The minimum over the smaller class cannot be lower, so Theorem 3.2 yields $V_{T,N}(p) \leq V_{T,M}(p)$ for $T \geq 2$. The lower endpoint follows from Corollary 3.3. The independent law $\mathrm{Bernoulli}(p)^{\otimes N}$ belongs to $\mathcal{P}_N(p)$ and has prefix-rate variance $p(1-p)/T$, giving the upper bound. For $T = 1$, every rate is $\mathrm{Bernoulli}(p)$ and all inequalities are equalities.

It remains to prove convergence of the joint law, rather than just its variance. We use the convention $d_{\mathrm{TV}}(P, Q) = \sup_A |P(A) - Q(A)|$. Conditional on $K = k$, denote the USHE length-T law by P_k and let $R_k = \mathrm{Bernoulli}(k/N)^{\otimes T}$. To compare them, fix a binary population with k ones and sample T independent uniform indices from its N positions. Conditional on the indices being distinct, their ordered law is uniform sampling without replacement. We may therefore couple sampling with and without replacement to agree whenever no index is repeated. A union bound over pairs of indices gives

$$d_{\mathrm{TV}}(P_k, R_k) \leq \mathbb{P}(\text{some repeated index}) \leq \frac{\binom{T}{2}}{N}.$$

Let $\widetilde{P}_{T,N} = \mathbb{E}[R_K]$ be the corresponding mixture of independent Bernoulli laws. Averaging the preceding couplings over K shows

$$d_{\mathrm{TV}}\Big(\mathcal{L}(s_{1:T}), \widetilde{P}_{T,N}\Big) \leq \frac{T(T-1)}{2N}.$$

Next couple a $\mathrm{Bernoulli}(K/N)$ draw and a $\mathrm{Bernoulli}(p)$ draw using the same uniform random variable, independently at each position conditional on K. The probability of any discrepancy is at most $T|K/N - p|$. For stochastic rounding,

$$\mathbb{E}\left[\left|\frac{K}{N} - p\right|\right] = (1 - r_N)\frac{r_N}{N} + r_N\frac{1 - r_N}{N} = \frac{2v_N}{N}.$$

Thus

$$d_{\mathrm{TV}}\Big(\widetilde{P}_{T,N}, \mathrm{Bernoulli}(p)^{\otimes T}\Big) \leq \frac{2Tv_N}{N}.$$

Combining the bounds and using $v_N \leq 1/4$ gives the explicit estimate

$$\begin{aligned} d_{\mathrm{TV}}\big(\mathcal{L}(s_{1:T}), \mathrm{Bernoulli}(p)^{\otimes T}\big) &\leq \min\left\{1, \frac{T(T-1)}{2N} + \frac{2Tv_N}{N}\right\} \\ &\leq \min\left\{1, \frac{T^2}{2N}\right\}. \end{aligned} \tag{20}$$

At fixed T, this tends to zero uniformly in p. For $T = 1$, the two laws are in fact identical for every N. □

A.6 Optimality for convex rate losses

The variance theorem has a stronger counterpart, but its proof requires more than the moment calculation. We first show that conditional sampling without replacement preserves discrete convexity in the latent spike count.

Lemma A.3 (Convexity in the population count)**.** For fixed $1 \le T \le N$ and convex $\phi : [0, 1] \to \mathbb{R}$, let

$$g_\phi(c) = \mathbb{E}\left[\phi\left(\frac{H_c}{T}\right)\right], \qquad c \in \{0, \dots, N\},$$

where H_c counts the ones in a uniform sample of T positions from a population with c ones and $N - c$ zeros. Then g_ϕ has nonnegative second finite differences.

Proof. For $T = 1$,

$$g_\phi(c) = \left(1 - \frac{c}{N}\right)\phi(0) + \frac{c}{N}\phi(1),$$

which is affine in c. Now take $T \ge 2$ and $0 \le c \le N - 2$. Let J be a uniformly chosen subset of $\{1, \dots, N\}$ of size T and set

$$R = |J \cap \{1, \dots, c\}|, \qquad A = \mathbf{1}\{c + 1 \in J\}, \qquad B = \mathbf{1}\{c + 2 \in J\}.$$

Using nested populations with c, $c + 1$, and $c + 2$ initial ones, the respective sampled counts are R, $R + A$, and $R + A + B$. The pairs (R, A) and (R, B) have the same law. Writing $f(h) = \phi(h/T)$, we obtain

$$\begin{aligned} &g_\phi(c + 2) - 2g_\phi(c + 1) + g_\phi(c) \\ &\quad = \mathbb{E}[f(R + A + B) - f(R + A) - f(R + B) + f(R)] \\ &\quad = \mathbb{P}(A = B = 1)\,\mathbb{E}[f(R + 2) - 2f(R + 1) + f(R) \mid A = B = 1] \ge 0. \end{aligned}$$

The last inequality follows from convexity of ϕ on the equally spaced points $R/T, (R + 1)/T, (R + 2)/T$ whenever $A = B = 1$. On that event $R \le T - 2$, so all three arguments lie in $[0, 1]$. Before conditioning, the mixed difference is zero whenever $A = 0$ or $B = 0$. □

Theorem A.4 (Simultaneous optimality for convex rate losses)**.** For $1 \le T \le N$, $p \in [0, 1]$, every $Q \in \mathcal{P}_N(p)$, and every convex $\phi : [0, 1] \to \mathbb{R}$,

$$\mathbb{E}[\phi(\widehat{p}_{T,N})] \le \mathbb{E}_Q[\phi(\widehat{p}_T(X))]. \tag{21}$$

Thus the same law $Q^\star_N$ minimizes all such losses for all visible prefix lengths. This does not assert uniqueness for each individual convex loss.

Proof. For $Q \in \mathcal{P}_N(p)$, conditional uniformity gives

$$\mathbb{E}_Q[\phi(\widehat{p}_T(X))] = \mathbb{E}_Q[g_\phi(C)], \qquad C = \sum_{i=1}^{N} X_i.$$

The count has mean Np, and g_ϕ is discretely convex by Lemma A.3. Lemma A.1 therefore implies

$$\mathbb{E}_Q[g_\phi(C)] \ge \mathbb{E}[g_\phi(K)] = \mathbb{E}[\phi(\widehat{p}_{T,N})],$$

where $K = \lfloor Np \rfloor + \mathrm{Bernoulli}(r_N)$. This is precisely equation 21. For a fixed p, the choices $\phi(u) = |u - p|^q$, $q \ge 1$, give the claimed convex reconstruction losses. □

Corollary A.5 (Convex ordering and infinite exchangeability)**.** For fixed T, p and integers $T \le N \le M$,

$$\widehat{p}_{T,T} \preceq_{\mathrm{cx}} \widehat{p}_{T,N} \preceq_{\mathrm{cx}} \widehat{p}_{T,M} \preceq_{\mathrm{cx}} \frac{B_T}{T}, \qquad B_T \sim \mathrm{Binomial}(T, p). \tag{22}$$

Here $U \preceq_{\mathrm{cx}} V$ means $\mathbb{E}[\phi(U)] \le \mathbb{E}[\phi(V)]$ for every convex $\phi : [0, 1] \to \mathbb{R}$. Furthermore, for every infinite exchangeable binary sequence $X_1, X_2, \dots$ with $\mathbb{E}[X_i] = p$,

$$\mathbb{E}\left[\phi\left(\frac{B_T}{T}\right)\right] \le \mathbb{E}\left[\phi\left(\frac{1}{T}\sum_{t=1}^{T} X_t\right)\right] \tag{23}$$

for every convex ϕ.

Proof. Restricting $Q_M^\star$ to N coordinates gives a feasible competitor in $\mathcal{P}_N(p)$. Applying Theorem A.4 yields $\widehat{p}_{T,N} \preceq_{\mathrm{cx}} \widehat{p}_{T,M}$. The same argument with $N = T$ gives the first comparison, and the independent Bernoulli population is a feasible competitor for the last comparison. This proves equation 22.

For an infinite exchangeable sequence, its first N coordinates belong to $\mathcal{P}_N(p)$ for every $N \geq T$. Hence Theorem A.4 bounds its first-T convex rate loss from below by that of USHE for every N. By equation 9, the latter converges to the loss of B_T/T. The limit passes through the expectation because all rates lie in the fixed finite set $\{0, 1/T, \ldots, 1\}$, on which ϕ is bounded. This proves equation 23 without requiring a separate representation theorem for infinite exchangeable laws. □

A.7 A MINIMAX INTERPRETATION WITHOUT EXCHANGEABILITY

Exchangeability treats temporal positions symmetrically. An alternative is to allow arbitrary dependence and ordering, but evaluate the worst choice of visible positions. Let $\mathcal{A}_N(p)$ be the class of all laws of binary vectors $X_{1:N}$ with $\mathbb{E}[X_i] = p$ for every i, without imposing exchangeability. For $A \subseteq \{1, \ldots, N\}$, $|A| = T$, write $\widehat{p}_A = T^{-1} \sum_{i \in A} X_i$.

Theorem A.6 (Minimax rate variance over visible positions)**.** For $p \in [0, 1]$ and $1 \leq T \leq N$,

$$\min_{Q \in \mathcal{A}_N(p)} \max_{\substack{A \subseteq \{1,\ldots,N\} \\ |A|=T}} \mathrm{Var}_Q(\widehat{p}_A) = V_{T,N}(p). \tag{24}$$

The USHE population law $Q_N^\star$ attains this minimum simultaneously for every $T \leq N$. No uniqueness is asserted within the larger class $\mathcal{A}_N(p)$.

Proof. Fix $Q \in \mathcal{A}_N(p)$. Independently of X, draw a uniform random permutation π of $\{1, \ldots, N\}$ and set $Y_i = X_{\pi(i)}$. The law $\overline{Q}$ of $Y_{1:N}$ is exchangeable with marginal mean p, so $\overline{Q} \in \mathcal{P}_N(p)$. Since each $\widehat{p}_A$ is unbiased,

$$\begin{aligned}
\max_{|A|=T} \mathrm{Var}_Q(\widehat{p}_A) &\geq \frac{1}{\binom{N}{T}} \sum_{|A|=T} \mathrm{Var}_Q(\widehat{p}_A) \\
&= \frac{1}{\binom{N}{T}} \sum_{|A|=T} \mathbb{E}_Q[(\widehat{p}_A - p)^2] \\
&= R_T(\overline{Q}; p) \geq V_{T,N}(p).
\end{aligned}$$

The last inequality is Theorem 3.2 for $T \geq 2$; for $T = 1$ it follows directly from the fixed Bernoulli marginals. Under $Q_N^\star$, exchangeability makes every T-subset rate identically distributed with variance $V_{T,N}(p)$. Thus the lower bound is attained, for every T by the same population law. □

This minimax statement does not say that every fixed prefix of an arbitrary nonexchangeable competitor has risk at least $V_{T,N}(p)$. A competitor can favor particular positions at the expense of others. The guarantee concerns its worst T-subset, and does not extend to arbitrary nonlinear SNN readouts or to adversarial classification accuracy.

B EXPERIMENTAL DETAILS

B.1 ARCHITECTURES AND TRAINING

We evaluate two SNN architectures on CIFAR-10 and CIFAR-100: VGG-SNN and TET-ResNet-19. VGG-SNN follows the VGG family (Simonyan & Zisserman, 2014) and uses eight convolutional stages with channel widths $(64, 128, 256, 256, 512, 512, 512, 512)$, average pooling, batch normalization, and no dropout; the final classifier is non-spiking. ResNet-19 follows the residual SNN architecture used as in (Deng et al., 2022), building on residual networks (He et al., 2016). It uses a 64-channel stem followed by stages with $(128, 256, 512)$ channels and depths $(3, 3, 2)$, respectively, with a 256-dimensional classifier representation.

All models use stateful LIF neurons with membrane time constant $\tau = 2$, firing threshold $\vartheta = 1$, and hard reset to $v_{\text{reset}} = 0$. Input current is not decayed and the reset is not detached from the computational graph. Gradients through the spike function are estimated using a triangular surrogate with width 1. The same neuron configuration is used for all architectures and encoders.

Unless stated otherwise, models are trained for 200 epochs using SGD with batch size 128, initial learning rate 0.1, momentum 0.9, and weight decay 5×10^{-4}. The learning rate follows cosine decay to zero without warmup. We optimize cross-entropy loss and use FP16 automatic mixed precision. Training uses BatchNorm, no gradient accumulation, and no gradient clipping. The same optimization hyperparameters are used across datasets, architectures, latencies, and encoders.

For CIFAR, inputs are represented in $[0, 1]$ without additional input normalization. AutoAugment, RandAugment, random erasing, MixUp, CutMix, EventMix, and temporal-step dropout are disabled. Training batches use `drop_last=True`. Unless latency is explicitly varied, training and evaluation use matched simulation horizons.

B.2 ROBUSTNESS EVALUATION

We evaluate clean accuracy together with FGSM (Goodfellow et al., 2015), R-FGSM, PGD (Madry et al., 2018), APGD (Croce & Hein, 2020), and additive Gaussian noise. All adversarial attacks use an ℓ_∞ budget of $\epsilon = 8/255$ and constrain perturbed inputs to $[0, 1]$.

The discrete stochastic input encoding is handled during gradient-based attacks using an expected-rate BPDA surrogate (Athalye et al., 2018a). FGSM, R-FGSM, and PGD optimize the summed logits across timesteps, while APGD uses the individual cross-entropy objective. Unless stated otherwise, one stochastic realization is used for gradient estimation and evaluation. Attacks are generated separately for the classifier being evaluated; in particular, encoder-substitution experiments regenerate adversarial examples after replacing the inference encoder rather than transferring attacks generated for the original encoder.

FGSM performs one gradient step at the full $8/255$ perturbation budget. R-FGSM first applies a random initialization of magnitude $4/255$ and then performs a single gradient-based update while projecting the complete perturbation onto the same total $8/255$ ℓ_∞ ball. PGD-7 uses seven projected steps of size 0.01, a random start, and one restart. APGD uses 100 iterations and one restart with its standard adaptive step-size mechanism; we use $\rho = 0.75$ and momentum 0.75. For corruption evaluation, independent Gaussian noise with $\sigma = 8/255$ is added before clipping to $[0, 1]$.

Predictions for FGSM, R-FGSM, PGD, and Gaussian-noise evaluation are obtained from mean logits. APGD evaluation aggregates mean probabilities. The main robustness comparisons report arithmetic means and sample standard deviations over the recorded evaluation seeds; these standard deviations describe evaluation variability and are not confidence intervals.

Table 4: Training and neuron hyperparameters shared across the main experiments.

Training	**Value**	**SNN**	**Value**
Optimizer	SGD	Neuron	LIF
Epochs	200	τ	2.0
Batch size	128	Threshold	1.0
Initial LR	0.1	Reset	Hard, $v = 0$
LR schedule	Cosine	Input decay	No
Minimum LR	0	Detach reset	No
Warmup	None	Surrogate	Triangular
Momentum	0.9	Surrogate width	1.0
Weight decay	5×10^{-4}	Normalization	BatchNorm
Loss	Cross-entropy	Stateful	Yes
Precision	FP16 AMP	Classifier	Non-spiking

Table 5: Robustness evaluation settings. Adversarial attacks use an ℓ_∞ budget of $8/255$ and an expected-rate BPDA surrogate. All perturbed inputs are clipped to $[0, 1]$.

Attack	**Budget**	**Steps**	**Step / init.**	**Restarts**	**Objective**
FGSM	$\epsilon = 8/255$	1	$8/255$	–	Summed logits
R-FGSM	$\epsilon = 8/255$	1	init. $4/255$	1	Summed logits
PGD-7	$\epsilon = 8/255$	7	0.01	1	Summed logits
APGD-CE	$\epsilon = 8/255$	100	Adaptive	1	Cross-entropy
Gaussian noise	$\sigma = 8/255$	–	–	–	–

APGD uses $\rho = 0.75$ and momentum 0.75. R-FGSM's $4/255$ random initialization is contained within the total $8/255$ perturbation budget.

C ADDITIONAL EXPERIMENTS

C.1 ADDITIONAL CLEAN ACCURACY AND ROBUSTNESS ACCURACY RESULTS UNDER VARIOUS ATTACKS AND IMAGE CORRUPTION.

Table 6: Full robustness results on CIFAR-10 (top-1 accuracy, %). Training and evaluation use the same horizon T. Gray superscripts give the reported standard deviations; entries without a supplied SD are shown without a superscript. The subscript in USHE_N denotes population size N. Dashes denote results not supplied. Bold marks the largest available mean per metric and architecture–latency block, including displayed ties. PGD-7 uses seven iterations; GN denotes Gaussian noise with $\sigma = 8/255$.

Encoder	Clean	FGSM	RFGSM	PGD-7	APGD-CE	GN
VGG-SNN | $T = 4$						
Direct	**93.72**$^{0.00}$	12.24$^{0.00}$	18.70$^{0.07}$	0.05$^{0.01}$	00.00	**86.28**$^{0.09}$
Bernoulli	79.31$^{0.10}$	43.52$^{0.63}$	63.00$^{0.38}$	**38.13**$^{0.05}$	**35.05**	79.17$^{0.04}$
USHE_4	84.95$^{0.23}$	38.67$^{0.24}$	63.73$^{0.18}$	30.86$^{0.20}$	25.89	84.76$^{0.17}$
USHE_8	81.86$^{0.11}$	42.74$^{0.16}$	**64.23**$^{0.31}$	36.05$^{0.28}$	32.45	81.56$^{0.12}$
USHE_{12}	81.02$^{0.12}$	43.42$^{0.47}$	63.69$^{0.66}$	36.89$^{0.27}$	33.57	81.01$^{0.07}$
USHE_{16}	80.47$^{0.08}$	43.54$^{0.30}$	—	37.17$^{0.50}$	34.13	80.44$^{0.35}$
USHE_{20}	80.17$^{0.07}$	**43.94**$^{0.32}$	—	37.18$^{0.18}$	—	80.34$^{0.21}$
USHE_{24}	79.90$^{0.10}$	43.70$^{0.40}$	—	37.45$^{0.24}$	—	79.63$^{0.20}$
VGG-SNN | $T = 8$						
Direct	**93.86**$^{0.00}$	14.97$^{0.00}$	24.67$^{0.08}$	0.09$^{0.01}$	00.00	**87.08**$^{0.19}$
Bernoulli	82.97$^{0.14}$	**41.23**$^{0.23}$	**64.06**$^{0.32}$	**34.58**$^{0.49}$	**29.87**	83.01$^{0.14}$
USHE_8	86.27$^{0.01}$	33.47$^{0.10}$	61.53$^{0.18}$	24.36$^{0.05}$	20.06	86.54$^{0.12}$
USHE_{12}	85.19$^{0.07}$	36.60$^{0.05}$	—	28.44$^{0.04}$	—	85.15$^{0.11}$
USHE_{16}	84.64$^{0.01}$	38.32$^{0.52}$	63.36$^{0.17}$	30.08$^{0.11}$	26.23	84.79$^{0.26}$
USHE_{20}	84.74$^{0.02}$	38.46$^{0.15}$	—	—	—	84.58$^{0.09}$
USHE_{24}	84.06$^{0.04}$	39.46$^{0.05}$	63.03$^{0.27}$	32.31$^{0.31}$	28.36	83.88$^{0.08}$
ResNet-19 | $T = 4$						
Direct	**93.96**$^{0.00}$	15.00$^{0.00}$	15.57$^{0.12}$	0.00$^{0.01}$	00.00	83.22$^{0.04}$
Bernoulli	79.17$^{0.02}$	51.01$^{0.24}$	66.15$^{0.22}$	44.89$^{0.39}$	41.66	79.08$^{0.16}$
USHE_4	85.52$^{0.14}$	48.87$^{0.07}$	**69.26**$^{0.46}$	40.54$^{0.21}$	35.11	**85.28**$^{0.04}$
USHE_8	81.73$^{0.03}$	52.87$^{0.55}$	68.91$^{0.33}$	**46.41**$^{0.45}$	42.71	81.61$^{0.28}$
USHE_{12}	80.99$^{0.11}$	**52.89**$^{0.16}$	68.20$^{0.09}$	46.39$^{0.49}$	**42.92**	80.62$^{0.02}$
USHE_{16}	80.52$^{0.10}$	52.85$^{0.16}$	—	46.05$^{0.25}$	42.62	80.45$^{0.23}$
USHE_{20}	80.06$^{0.18}$	52.42$^{0.11}$	—	46.34$^{0.13}$	—	80.10$^{0.12}$
USHE_{24}	80.02$^{0.21}$	51.45$^{0.20}$	—	45.29$^{0.36}$	—	80.04$^{0.18}$
ResNet-19 | $T = 8$						
Direct	**94.86**$^{0.00}$	17.89$^{0.00}$	18.36$^{0.30}$	0.01$^{0.01}$	00.00	83.47$^{0.09}$
Bernoulli	83.71$^{0.00}$	**52.34**$^{0.18}$	69.78$^{0.24}$	**44.86**$^{0.17}$	**40.51**	83.75$^{0.16}$
USHE_8	88.32$^{0.01}$	43.14$^{0.23}$	68.01$^{0.15}$	31.92$^{0.10}$	25.37	**88.24**$^{0.12}$
USHE_{12}	86.07$^{0.07}$	49.86$^{0.41}$	—	41.00$^{0.31}$	—	85.89$^{0.17}$
USHE_{16}	85.67$^{0.12}$	48.50$^{0.25}$	68.74$^{0.18}$	39.84$^{0.36}$	33.78	85.49$^{0.26}$
USHE_{24}	84.85$^{0.09}$	50.82$^{0.35}$	**69.90**$^{0.38}$	42.97$^{0.13}$	37.67	84.61$^{0.08}$

Table 7: Full robustness results on CIFAR-100 (top-1 accuracy, %). Training and evaluation use the same horizon T. Gray superscripts give the reported standard deviations; entries without a supplied SD are shown without a superscript. The subscript in USHE_N denotes population size N. Dashes denote results not supplied. Bold marks the largest available mean per metric and architecture–latency block, including displayed ties. PGD-7 uses seven iterations; GN denotes Gaussian noise with $\sigma = 8/255$.

Encoder	Clean	FGSM	RFGSM	PGD-7	APGD-CE	GN
VGG-SNN | $T = 4$						
Direct	**74.48**$^{0.00}$	4.62$^{0.00}$	4.92$^{0.03}$	0.02$^{0.01}$	00.00	57.11$^{0.20}$
Bernoulli	51.52$^{0.11}$	**21.60**$^{0.06}$	35.11$^{0.18}$	**18.26**$^{0.35}$	**16.75**	51.50$^{0.11}$
USHE_4	58.00$^{0.09}$	17.95$^{0.10}$	35.19$^{0.13}$	14.03$^{0.19}$	11.82	**57.68**$^{0.36}$
USHE_8	53.90$^{0.05}$	20.39$^{0.39}$	**35.21**$^{0.20}$	16.52$^{0.17}$	14.18	53.75$^{0.52}$
USHE_{12}	53.41$^{0.12}$	20.73$^{0.16}$	**35.21**$^{0.13}$	17.22$^{0.04}$	15.38	53.75$^{0.41}$
USHE_{16}	52.57$^{0.20}$	20.80$^{0.22}$	—	17.29$^{0.31}$	15.58	52.68$^{0.26}$
VGG-SNN | $T = 8$						
Direct	**74.78**$^{0.00}$	4.42$^{0.00}$	5.49$^{0.22}$	0.06$^{0.02}$	—	60.11$^{0.08}$
Bernoulli	56.37$^{0.13}$	**19.50**$^{0.14}$	**35.94**$^{0.27}$	**15.51**$^{0.46}$	—	56.33$^{0.00}$
USHE_8	61.29$^{0.03}$	16.06$^{0.23}$	34.46$^{0.23}$	11.54$^{0.15}$	—	**61.34**$^{0.12}$
USHE_{12}	59.54$^{0.14}$	17.45$^{0.15}$	—	13.44$^{0.11}$	—	59.26$^{0.04}$
USHE_{16}	58.65$^{0.12}$	18.59$^{0.00}$	35.88$^{0.25}$	14.31$^{0.10}$	—	58.53$^{0.39}$
USHE_{24}	57.92$^{0.04}$	18.75$^{0.07}$	35.91$^{0.23}$	14.48$^{0.19}$	—	57.64$^{0.09}$
ResNet-19 | $T = 4$						
Direct	**76.87**$^{0.00}$	9.26$^{0.00}$	8.28$^{0.14}$	0.03$^{0.01}$	00.00	57.71$^{0.16}$
Bernoulli	51.57$^{0.20}$	28.13$^{0.50}$	39.40$^{0.31}$	24.16$^{0.18}$	22.23	51.27$^{0.19}$
USHE_4	58.26$^{0.14}$	26.73$^{0.29}$	**40.92**$^{0.21}$	21.37$^{0.08}$	18.45	**58.06**$^{0.20}$
USHE_8	54.02$^{0.13}$	29.10$^{0.51}$	40.77$^{0.09}$	24.91$^{0.13}$	22.52	53.94$^{0.18}$
USHE_{12}	52.77$^{0.22}$	29.16$^{0.02}$	40.28$^{0.26}$	24.97$^{0.08}$	22.56	52.78$^{0.09}$
USHE_{16}	52.17$^{0.05}$	28.69$^{0.32}$	—	24.81$^{0.22}$	**22.99**	52.30$^{0.01}$
USHE_{24}	51.77$^{0.08}$	**29.18**$^{0.17}$	—	**25.16**$^{0.26}$	—	51.90$^{0.21}$
ResNet-19 | $T = 8$						
Direct	**77.30**$^{0.00}$	10.04$^{0.00}$	9.11$^{0.20}$	0.04$^{0.01}$	—	56.31$^{0.03}$
Bernoulli	57.44$^{0.05}$	**27.12**$^{0.06}$	**41.29**$^{0.14}$	**22.27**$^{0.23}$	—	57.31$^{0.28}$
USHE_8	63.38$^{0.05}$	21.51$^{0.00}$	39.62$^{0.19}$	15.59$^{0.13}$	—	**63.28**$^{0.18}$
USHE_{12}	60.88$^{0.27}$	25.76$^{0.26}$	—	20.01$^{0.10}$	—	61.16$^{0.22}$
USHE_{16}	59.79$^{0.00}$	25.43$^{0.04}$	41.09$^{0.07}$	20.29$^{0.22}$	—	59.62$^{0.12}$
USHE_{24}	58.99$^{0.09}$	25.43$^{0.10}$	40.64$^{0.11}$	20.11$^{0.27}$	—	58.79$^{0.48}$

Table 8: Mean layer spike rate (%). We compare Direct and Bernoulli encoding with representative USHE populations $N \in \{T, 2T, 3T\}$. Δ_{B} is the relative change in mean layer spike rate with respect to Bernoulli under the same architecture and latency. Negative values indicate lower activity than Bernoulli and positive values indicate higher activity. Results correspond to the mean of 3 evaluation seeds.

(a) CIFAR-10

Architecture	T	Encoder	Spike rate	Δ_{B}
ResNet-19	4	Direct	5.350	+19.16%
		Bernoulli	4.490	–
		USHE_4	4.616	+2.80%
		USHE_8	4.803	+6.99%
		USHE_{12}	4.594	+2.32%
	8	Direct	4.950	-6.62%
		Bernoulli	5.301	–
		USHE_8	5.114	-3.52%
		USHE_{16}	5.414	+2.13%
		USHE_{24}	5.384	+1.57%
VGG-SNN	4	Direct	4.839	-5.28%
		Bernoulli	5.109	–
		USHE_4	4.461	-12.69%
		USHE_8	4.880	-4.49%
		USHE_{12}	5.009	-1.96%
	8	Direct	5.037	+13.03%
		Bernoulli	4.456	–
		USHE_8	3.989	-10.48%
		USHE_{16}	4.275	-4.06%
		USHE_{24}	4.343	-2.55%

(b) CIFAR-100

Architecture	T	Encoder	Spike rate	Δ_{B}
ResNet-19	4	Direct	8.231	+44.13%
		Bernoulli	5.711	–
		USHE_4	6.863	+20.16%
		USHE_8	6.010	+5.24%
		USHE_{12}	5.985	+4.79%
	8	Direct	8.324	+8.21%
		Bernoulli	7.693	–
		USHE_8	7.707	+0.19%
		USHE_{16}	7.782	+1.16%
		USHE_{24}	7.865	+2.25%
VGG-SNN	4	Direct	7.518	-6.86%
		Bernoulli	8.072	–
		USHE_4	7.175	-11.11%
		USHE_8	7.777	-3.65%
		USHE_{12}	7.988	-1.04%
	8	Direct	7.666	+11.33%
		Bernoulli	6.886	–
		USHE_8	6.321	-8.21%
		USHE_{16}	6.673	-3.09%
		USHE_{24}	6.751	-1.96%

C.2 Spike Activity

We additionally examine whether the accuracy differences between input encoders are accompanied by systematic changes in internal network activity. For each trained model, we report the validation mean layer spike rate at the checkpoint selected by highest validation top-1 accuracy. Table 8 compares Direct and Bernoulli encoding with three representative USHE members, $N \in \{T, 2T, 3T\}$.

The effect of USHE on firing activity is architecture dependent. For VGG-SNN, all reported finite-population encoders reduce the mean spike rate relative to Bernoulli, with the largest reduction consistently occurring at $N = T$. As N increases, the activity moves closer to the Bernoulli level. ResNet-19 shows a different pattern: USHE generally produces comparable or moderately higher activity than Bernoulli, with the largest increase observed for CIFAR-100 at $T = 4$ and $N = T$. Thus, the accuracy improvements obtained with USHE are not explained by a systematic increase in firing activity; changing the temporal encoding law can alter network performance while leaving average activity at a broadly similar scale.

These measurements are descriptive results for the selected checkpoints; the reported experiments use one training seed and therefore do not estimate variability across independently trained models.

C.3 INFERENCE ENCODER SWAPPING



Figure 5: Encoder swapping during inference.
Setting: Cifar-10 ResNet-19 $T = 4$



Figure 6: Encoder swapping during inference.
Setting: Cifar-10 VGG-SNN $T = 4$