SCALABLE NEURAL QUANTUM STATES WITH MODERN LANGUAGE MODELING

Anonymous authors
Paper under double-blind review

ABSTRACT

Scaling neural representations to increasingly large quantum many-body systems remains a central challenge in quantum physics. Despite the success of Neural Quantum States (NQS), their dependence on costly Monte Carlo sampling and second-order optimizations fundamentally limits their scalability to large systems. Here, we introduce FASTNQS, a transformer-based variational framework that combines autoregressive Born sampling, FlashAttention-accelerated wavefunction evaluation, and Muon optimization. Across benchmarks on the transverse-field Ising model, two-dimensional J_1, J_1-J_2, and random-bond Heisenberg models, FASTNQS achieves competitive accuracy against Parallel RNN or ViT-based NQS baselines, while being significantly faster: an approximately $23.6\times$ speedup over ViT-based NQS in end-to-end runtime per step. The results highlight modern language modeling as a practical route toward scalable neural quantum states.

1 INTRODUCTION

Efficient simulation of interacting quantum many-body systems remains difficult because the Hilbert space grows exponentially with system size. Large lattices are nevertheless essential for suppressing finite-size effects, resolving long-range correlations, and extrapolating observables and phase boundaries to the thermodynamic limit. Neural quantum states (NQS) address the representation problem by encoding many-body wavefunctions with trainable networks rather than explicitly storing exponentially many amplitudes (Carleo & Troyer, 2017; Lange et al., 2024; Rigo et al., 2026). Progress toward physically informative system sizes therefore depends on the efficiency of the complete variational calculation as well as the expressivity of the ansatz.

Transformer-based NQS have become strong variational ansatzes for frustrated quantum systems. Self-attention exchanges information directly between distant degrees of freedom, and transformer wavefunctions have reached high accuracy on frustrated magnets and increasingly large two-dimensional lattices (Viteritti et al., 2023; Chen et al., 2026; Viteritti et al., 2026). Standard self-attention incurs quadratic arithmetic and activation memory in the number of tokens (Vaswani et al., 2017; Dao et al., 2022). For an $L \times L$ lattice represented by one token per site, the attention matrix alone grows as $O(L^4)$. Factored Attention removes query-key computation, yet its $L^2 \times L^2$ positional attention matrix remains costly to materialize (Rende & Viteritti, 2025). Transformer-based NQS with conventional attention implementations therefore remain constrained by attention-related resource demands at large system sizes.

The bottleneck extends beyond the attention layer and spans the variational Monte Carlo (VMC) pipeline. Markov chain Monte Carlo (MCMC) sampling requires equilibration and can suffer from autocorrelation and slow mixing, whereas autoregressive NQS generate independent samples from a normalized Born distribution (Sharir et al., 2020; Hibat-Allah et al., 2020). Autoregression removes the Markov chain, but ancestral sampling remains sequential and ordering-dependent because each conditional prediction depends on the generated prefix. Optimization adds a further scaling pressure. Stochastic reconfiguration (SR) and its sample-space variants improve variational optimization by incorporating wavefunction geometry, but their sample-dependent Jacobians and linear-algebra operations grow with the number of samples and model parameters (Sorella, 2001; Rende et al., 2024b; Chen & Heyl, 2024). These coupled costs motivate evaluating NQS scalability across sampling, wavefunction evaluation, and parameter updates.

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.

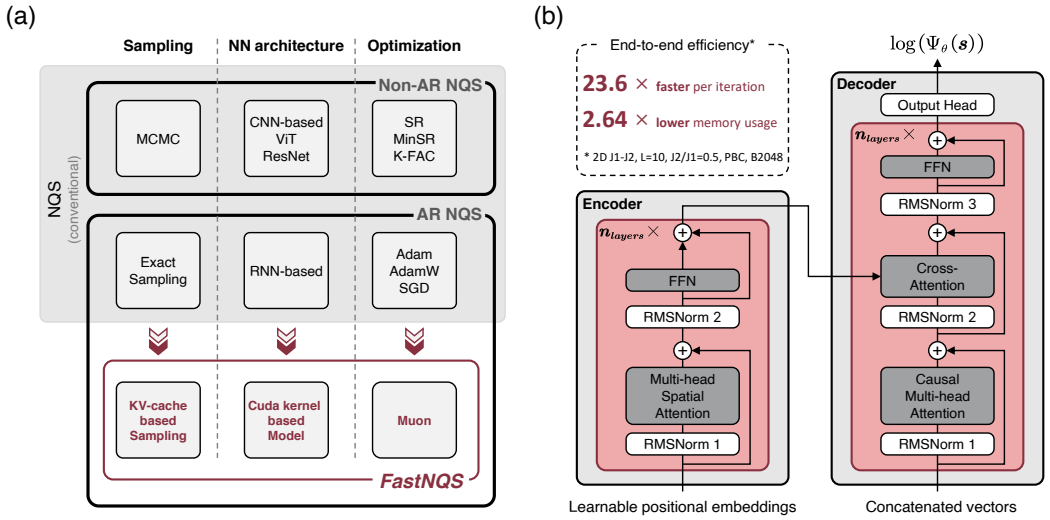


Figure 1: **FastNQS at a glance.** (a) FastNQS improves NQS training efficiency through three components: KV-cached sampling, fused CUDA kernels, and Muon optimization, shown alongside conventional non-autoregressive and autoregressive pipelines (Choo et al., 2019; Sharir et al., 2020; Hibat-Allah et al., 2020; Viteritti et al., 2023; Chen & Heyl, 2024). (b) Overview of the FastNQS architecture. A spatial encoder converts learnable positional embeddings into global lattice context, which conditions the causal decoder through cross-attention. The inset reports end-to-end gains at batch size 2048 on the two-dimensional $L = 10$ J_1-J_2 benchmark with $J_2/J_1 = 0.5$.

Recent NQS studies have adopted individual parts of this computational stack. Patched autoregressive transformers shorten quantum-state sequences (Sprague & Czischek, 2024), Parallel Scan Recurrent Neural Quantum States (PSR-NQS) accelerate recurrent evaluation (Merali et al., 2026), and Proximal Wavefunction Optimization (PWO) improves the training of autoregressive states through a trust-region objective (Duque et al., 2026). These studies address individual VMC bottlenecks, motivating an integrated framework across sampling, wavefunction evaluation, and optimization.

Building on these developments, we introduce FastNQS, a transformer-based variational framework that addresses these bottlenecks jointly rather than optimizing any single stage in isolation. FastNQS combines a spatial positional encoder, KV-cached autoregressive sampling (Pope et al., 2023), FlashAttention (Dao et al., 2022; Dao, 2024), and Muon optimization (Jordan et al., 2024; Liu et al., 2025) to improve computational efficiency across the full VMC pipeline.

Our main contributions are:

- **A scalable NQS framework.** We develop a unified framework combining spatial encoder, KV-cached autoregressive sampling, FlashAttention, and efficient first-order optimization.
- **Boosted computational efficiency.** FastNQS reduces memory by $\mathbf{394}\times$ over minGRU (Merali et al., 2026) on the 32×32 J_1 model, and achieves $\mathbf{23.6}\times$ speedup over ViT-NQS (Rende et al., 2024b) on the 10×10 J_1-J_2 model, with efficiency gains growing with size.
- **Broad applications to quantum many-body spin systems.** FastNQS is tested on the transverse-field Ising, two-dimensional J_1, J_1-J_2, and random-bond Heisenberg models, reaching competitive accuracy in frustrated and challenging disordered systems.

2 Related Work

Neural quantum states. Recent reviews cover the range of NQS representations and applications (Lange et al., 2024; Rigo et al., 2026). Transformer wavefunctions, convolutional transformers, spatial-attention models, and group-convolutional networks demonstrate the value of nonlocal information exchange, physical inductive biases, and symmetry-aware parameterizations (Viteritti

et al., 2023; Chen et al., 2026; Viteritti et al., 2026; Roth et al., 2023; Nutakki et al., 2025; Rende & Viteritti, 2025). Their accuracy depends on these choices, rather than on the transformer backbone alone (Westerhout et al., 2020; Szabó & Castelnovo, 2020). To our knowledge, FASTNQS is the first NQS framework to systematically adapt transformer-based modern language modeling, combining KV caching, FlashAttention, and Muon for scalability.

Autoregressive representations. Autoregressive NQS enable direct sampling from a normalized variational distribution (Sharir et al., 2020). Recurrent wavefunctions, patched transformers, and PSR-NQS extend this idea through sequence models and parallelizable recurrences, including minGRU (Hibat-Allah et al., 2020; Sprague & Czischek, 2024; Feng et al., 2024; Merali et al., 2026). However, conditional normalization changes expressive tradeoffs, and representation quality depends on amplitude correlations and basis ordering (Bortone et al., 2024; Lu et al., 2026). Unlike conventional autoregressive representations, FASTNQS introduces a spatial positional encoder to preserve lattice-wide structural information under causal generation.

Optimization beyond a single default. Stochastic reconfiguration (SR) uses wavefunction geometry to precondition energy gradients, and scalable variants include sample-space reformulations, minSR, and SPRING (Sorella, 2001; Rende et al., 2024b; Chen & Heyl, 2024; Goldshlager et al., 2024). PWO and regularized recurrent NQS offer further alternatives (Duque et al., 2026; Attar et al., 2026). Autoregression and optimization are separate design choices. We investigate Muon as a matrix-structured update for transformer weights that does not explicitly construct a quantum-geometric preconditioner (Jordan et al., 2024; Liu et al., 2025).

Efficient transformer computation. KV caching avoids reprocessing generated prefixes during autoregressive inference (Pope et al., 2023), while FlashAttention reduces memory traffic and intermediate storage through IO-aware tiling (Dao et al., 2022; Dao, 2024). FASTNQS brings these advances into NQS, using KV caching for efficient autoregressive sampling and FlashAttention for scalable wavefunction evaluation. Thus, it can serve as a computational bridge between modern language modeling and quantum many-body modeling.

3 PRELIMINARIES: VARIATIONAL MONTE CARLO

Consider N spin-$1/2$ sites and a computational-basis configuration $\mathbf{s} = (s_1, \ldots, s_N)$, with $s_i \in \{-1, +1\}$ labeling twice the spin-z eigenvalue. For a Hermitian Hamiltonian H, an NQS assigns an amplitude $\psi_\theta(\mathbf{s})$ and minimizes

$$E(\theta) = \frac{\langle \psi_\theta | H | \psi_\theta \rangle}{\langle \psi_\theta | \psi_\theta \rangle} = \mathbb{E}_{\mathbf{s}\sim p_\theta}[E_{\mathrm{loc},\theta}(\mathbf{s})], \qquad E(\theta) \geq E_0, \tag{1}$$

where E_0 is the exact ground-state energy and

$$p_\theta(\mathbf{s}) = \frac{|\psi_\theta(\mathbf{s})|^2}{\langle \psi_\theta | \psi_\theta \rangle}, \qquad E_{\mathrm{loc},\theta}(\mathbf{s}) = \sum_{\mathbf{s}'} H_{\mathbf{s},\mathbf{s}'} \frac{\psi_\theta(\mathbf{s}')}{\psi_\theta(\mathbf{s})}. \tag{2}$$

For a sparse Hamiltonian, the inner sum only visits configurations connected to $\mathbf{s}$ by a nonzero matrix element. VMC estimates the remaining expectation from samples drawn from p_θ and uses the resulting energy estimate to optimize the NQS. The variational bound applies to the exact expectation, whereas finite-sample estimates carry statistical uncertainty.

4 FASTNQS

FASTNQS is an autoregressive NQS that accelerates variational learning along three complementary dimensions: KV-cache-based sampling, CUDA-kernel-based wavefunction evaluation, and Muon optimization. Unlike non-autoregressive NQS, which commonly combine MCMC sampling with non-autoregressive architectures and SR-style optimization (Viteritti et al., 2023; Chen & Heyl, 2024; Rende et al., 2024b), FASTNQS retains direct autoregressive sampling while targeting the major computational costs of the learning pipeline. Figure 1 (a) summarizes the design.

4.1 AUTOREGRESSIVE VARIATIONAL LEARNING

We choose a lattice ordering and optionally group spins into patches. For notational simplicity, we write the resulting sequence as $\mathbf{s} = (s_1, \ldots, s_T)$, where each s_t denotes either a spin or a fixed patch. FASTNQS factorizes the Born distribution into normalized conditionals,

$$p_\theta(\mathbf{s}) = \prod_{t=1}^{T} p_\theta(s_t \mid s_{<t}), \qquad \psi_\theta(\mathbf{s}) = \exp\left[\frac{1}{2}\sum_{t=1}^{T} \log p_\theta(s_t \mid s_{<t}) + i\phi_\theta(\mathbf{s})\right]. \tag{3}$$

This parameterization normalizes p_θ without a global partition function and allows independent samples from the current variational state through ancestral generation. Exact sampling here means exact sampling from the variational Born distribution; it neither yields the exact ground state nor removes Monte Carlo uncertainty from energy estimates.

At each VMC iteration, we draw a fresh batch from p_θ, evaluate the local energies in Eq. equation 2, and update the parameters with the covariance-form energy gradient. For real trainable parameters, let $O_\theta(\mathbf{s}) = \nabla_\theta \log \psi_\theta(\mathbf{s})$. The gradient is

$$\nabla_\theta E = 2\,\mathrm{Re}\left[\mathbb{E}_{p_\theta}[E_{\mathrm{loc},\theta}(\mathbf{s})O_\theta(\mathbf{s})^*] - \mathbb{E}_{p_\theta}[E_{\mathrm{loc},\theta}(\mathbf{s})]\,\mathbb{E}_{p_\theta}[O_\theta(\mathbf{s})^*]\right]. \tag{4}$$

We estimate this expression with the sampled batch, treating both samples and local-energy values as fixed in the score-function estimator. Evaluating local energies requires wavefunction ratios for Hamiltonian-connected configurations, so the cost of known-configuration evaluation is often much larger than the initial sampling cost.

4.2 FASTNQS MODEL ARCHITECTURE

Figure 1 (b) highlights the distinct roles of the encoder and decoder. The encoder receives a learnable absolute embedding for each lattice position without taking any spin information as input, and aggregates these embeddings into a configuration-independent global memory. This supplies lattice-wide spatial context without revealing ungenerated spins; Section 5.3 shows that it improves the frustrated J_1-J_2 result over a plain causal decoder.

The decoder models the wavefunction autoregressively. It concatenates each shifted token embedding with its learnable positional embedding, so position t accesses only $s_{<t}$, and combines causal self-attention with cross-attention to the encoder memory. The output head applies a half-log-softmax to obtain normalized conditional log-amplitudes and, for complex wavefunctions, a separate $\pi\,\mathrm{softsign}$ projection for the phase. We defer lower-level attention and positional-encoding details to Appendix E.

4.3 EFFICIENT SAMPLING, EVALUATION, AND OPTIMIZATION

KV-cached sampling. Ancestral sampling remains sequential, since each token depends on its prefix. Recomputing the entire prefix at every step is unnecessary, however, FASTNQS stores the key and value tensors produced by previous tokens at every decoder layer, appends only the new token at the next step, and reuses the spatial positional encoder memory throughout generation. This avoids repeated prefix evaluation while preserving the same autoregressive distribution.

FlashAttention-accelerated wavefunction evaluation. For a known configuration, all conditional logits can be evaluated in parallel with a masked decoder pass. FASTNQS uses CUDA FlashAttention kernels for causal self-attention and cross-attention when available, reducing attention-memory traffic and intermediate storage (Dao et al., 2022; Dao, 2024). This is particularly relevant for VMC because local-energy estimation evaluates batches of Hamiltonian-connected configurations in addition to the sampled batch. Section 5 evaluates the resulting speed and memory advantages in both wavefunction evaluation and end-to-end VMC updates.

Muon optimization. Muon approximately orthogonalizes momentum updates for matrix-valued parameters through Newton-Schulz iterations, balancing singular directions while requiring less

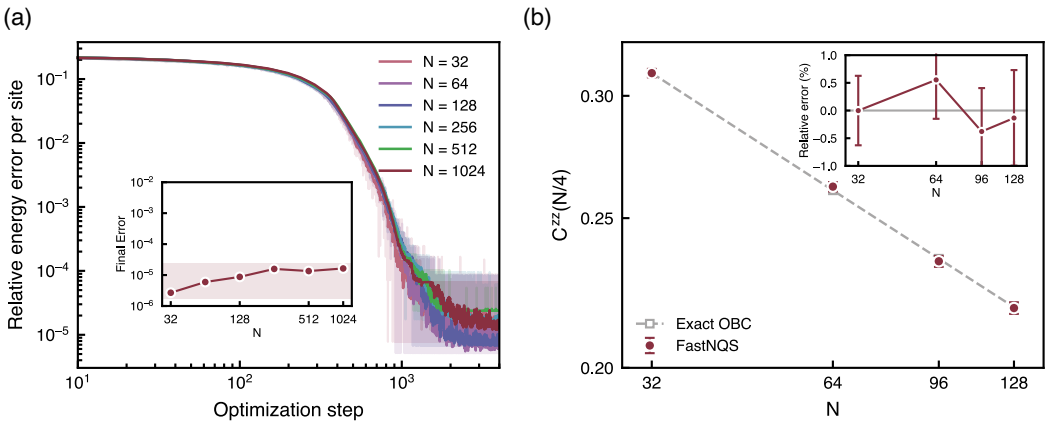


Figure 2: **FastNQS at the critical point of the 1D TFIM.** (a) Relative error in the ground-state energy per site as a function of optimization step for system sizes $N = 32$ through 1024. The inset shows the final relative error for each system size. (b) FastNQS estimates of the spin correlation $C^{zz}(N/4)$ at the critical field $h = 1$, compared with exact results on log-log scales. The inset shows the corresponding relative errors. The observed power-law scaling is consistent with the exact Ising critical behavior (Pfeuty, 1970) and previous NQS finite-size analyses (Kim et al., 2024).

optimizer-state memory than AdamW (Jordan et al., 2024; Liu et al., 2025; Loshchilov & Hutter, 2019). Unlike SR, it exploits parameter-matrix structure rather than variational-state geometry (Sorella, 2001; Chen & Heyl, 2024; Rende et al., 2024b). We apply Muon to matrix-valued parameters and AdamW to the remaining parameters. In the matched two-dimensional $L = 10$ J_1-J_2 ablation in Appendix B, Muon reaches a final-25-step mean energy of -0.491290, compared with -0.449751 for the best of five Adam learning rates, reducing the absolute error relative to the matching reference energy by approximately $7.5\times$.

5 Experiments

Building on the FastNQS architecture and training algorithm, we evaluate the method on four progressively more demanding systems: the exactly solvable 1D transverse-field Ising model (TFIM), the 2D J_1 Heisenberg model, frustrated 1D and 2D J_1-J_2 models, and random-bond Heisenberg lattices. This progression tests energy and correlation accuracy, computational efficiency, robustness to frustration, and performance across disorder realizations.

5.1 One-dimensional Transverse Field Ising Model

We first consider the one-dimensional transverse-field Ising model, a canonical benchmark with an exactly solvable ground state and a quantum critical point. Using Pauli operators and open boundary conditions (OBC), its Hamiltonian is:

$$H_{\mathrm{TFIM}} = -J \sum_{i=1}^{N-1} \sigma_i^z \sigma_{i+1}^z - h \sum_{i=1}^{N} \sigma_i^x, \tag{5}$$

where $J > 0$ is the ferromagnetic Ising coupling and h is the transverse field. We set $J = 1$ and study the critical point $h = 1$ (Pfeuty, 1970). The exact ground-state energy provides a direct benchmark of variational accuracy, while critical spin correlations probe whether the learned state captures long-distance physics.

Figure 2 (a) reports the relative ground-state energy error per site

$$\epsilon_E = \frac{|e_{\mathrm{FastNQS}} - e_{\mathrm{exact}}|}{|e_{\mathrm{exact}}|} \tag{6}$$

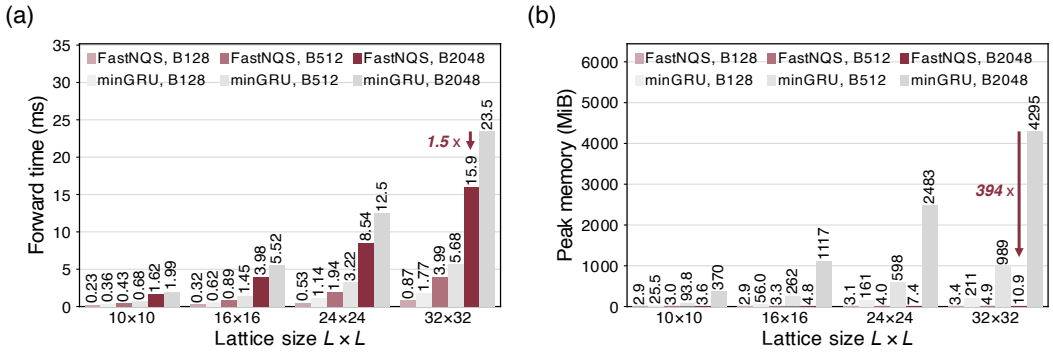


Figure 3: **Forward-evaluation efficiency of FASTNQS on the 2D J_1 Heisenberg model.** (a) Forward latency and (b) reported peak memory of FASTNQS and the RNN-based minGRU baseline (Merali et al., 2026), across lattice sizes and batch sizes, with 798,848 and 805,136 trainable parameters, respectively. Red (gray) bars denote FASTNQS (minGRU), and darker shades indicate larger batch sizes. At $L = 32$ and $B = 2048$, FASTNQS reduces forward latency from 23.5 to 15.9 ms ($1.5\times$) and reported peak memory from 4295 to 10.9 MiB ($394\times$).

Table 1: Ground-state energy per site e_0 for the 2D J_1 Heisenberg model on $L \times L$ square lattices with OBC and PBC. Relative errors are computed with respect to QMC (Sandvik, 2026) estimates. The 2D minGRU (Merali et al., 2026) results are available only for OBC. PBC calculations are performed for $L = 8, 10, 12, 16$, and 24.

	OBC				PBC		
L	FASTNQS	minGRU	QMC	Rel. Err. ($\times 10^{-4}$)	FASTNQS	QMC	Rel. Err. ($\times 10^{-4}$)
6	$\mathbf{-0.60352039}$	-0.603519	-0.6035222	0.0300			
8	-0.61901367	$\mathbf{-0.619025}$	-0.6190371	0.3785	-0.67340267	-0.67349005	1.2974
10	$\mathbf{-0.62861091}$	-0.628605	-0.6286561	0.7188	-0.67137170	-0.67155266	2.6947
12	-0.63511497	$\mathbf{-0.635127}$	-0.6352007	1.3497	-0.67036182	-0.67068192	4.7728
16	-0.64336008	$\mathbf{-0.643396}$	-0.6435317	2.6668	-0.66976982	-0.66997660	3.0864
24	$\mathbf{-0.65182}$	-0.651812	-0.6520251	3.1456	-0.66941994	-0.66960434	2.7539
32	$\mathbf{-0.65619648}$	-0.656089	-0.6563289	2.0176			

as a function of optimization step for $N = 32$ through 1024. For every system size, the error decreases by more than four orders of magnitude and reaches the 10^{-5} level at the end of optimization. The comparable convergence trajectories and final errors across this range indicate that FASTNQS preserves high energy accuracy as the chain length increases.

Figure 2 (b) provides a more stringent test of the learned state through its long-range correlations. We measure the central-chain longitudinal correlation $C^{zz}(r) = \langle \sigma_i^z \sigma_{i+r}^z \rangle$, with i chosen near the chain center. At the critical point, $C^{zz}(N/4)$ scales with system size as $N^{-\eta}$. FASTNQS agrees with the exact correlations across all displayed sizes, with sub-percent relative errors shown in the inset. A linear fit in log-log space yields $\eta = 0.2540$, close to the exact value $\eta = 1/4$.

5.2 TWO-DIMENSIONAL J_1 HEISENBERG MODEL

The spin-$1/2$ antiferromagnetic Heisenberg model on a two-dimensional square lattice is defined by

$$H_{J_1} = J_1 \sum_{\langle i,j \rangle} \mathbf{S}_i \cdot \mathbf{S}_j, \tag{7}$$

where $J_1 > 0$, $\langle i, j \rangle$ denotes nearest-neighbor pairs, and $\mathbf{S}_i$ is a spin-$1/2$ operator. This clean, unfrustrated system separates the computational cost of evaluating an autoregressive wavefunction from the additional difficulty of frustration.

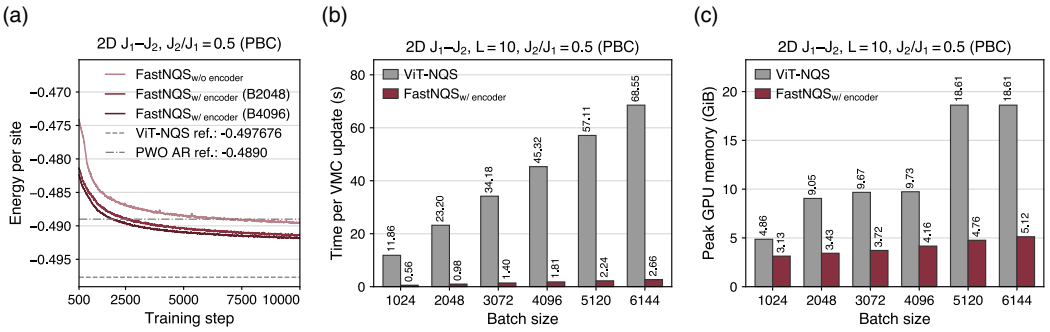


Figure 4: **FastNQS on the frustrated J_1-J_2 Heisenberg model.** (a) Energy convergence on the two-dimensional 10×10 lattice at $J_1 = 1$, $J_2 = 0.5$, and PBC. A spatial positional encoder improves upon a plain causal decoder, and increasing the batch size further lowers the energy, while the symmetry-aware ViT-NQS reference remains lower. (b) End-to-end wall-clock time per VMC optimization step and (c) peak GPU memory, both as functions of batch size on the same two-dimensional benchmark.

To assess the practical efficiency of FastNQS, we compare it with minGRU (Merali et al., 2026), a hardware-efficient autoregressive RNN-based NQS, using its parallel mode. Recurrent models avoid the quadratic attention cost of transformers. We therefore ask whether FastNQS can leverage highly optimized transformer kernels to achieve faster wavefunction evaluation while retaining comparable variational accuracy.

Figure 3 (a)(b) shows that FastNQS achieves both objectives. All efficiency measurements were performed on a single NVIDIA GeForce RTX 5090 GPU. Across lattice sizes from 10×10 to 32×32 and batch sizes from 128 to 2048, it reduces forward latency by 1.23–$2.15\times$ relative to minGRU. The corresponding measured peak memory is also much smaller: at $L = 32$ and batch size 2048, it falls from 4295 MiB for minGRU to 10.9 MiB for FastNQS. These are forward-evaluation measurements rather than total training-memory measurements. In the final training stage, we further fine-tune FastNQS with D_4 symmetry averaging, consistent with the symmetry treatment used for the reported minGRU results. The accuracy before and after D_4 fine-tuning is compared in Appendix C. Meanwhile, Table 1 shows that FastNQS maintains comparable variational accuracy, with errors below 3.15×10^{-4} under OBC and 4.77×10^{-4} under periodic Boundary Conditions (PBC); its OBC energies are comparable to those of minGRU, and the PBC results extend to a boundary condition not reported for that baseline.

5.3 Frustrated J_1-J_2 Heisenberg model

We now turn to the frustrated J_1-J_2 Heisenberg model,

$$H_{J_1\text{-}J_2} = J_1 \sum_{\langle i,j \rangle} \mathbf{S}_i \cdot \mathbf{S}_j + J_2 \sum_{\langle\langle i,j \rangle\rangle} \mathbf{S}_i \cdot \mathbf{S}_j. \tag{8}$$

For a chain, the two sums run over nearest- and next-nearest-neighbor pairs; on the square lattice, they run over nearest-neighbor and diagonal pairs, respectively. We use PBC throughout this comparison. Frustration at $J_1 = 1$ and $J_2 = 0.5$ makes this a substantially more demanding test than the unfrustrated J_1 model: a useful autoregressive NQS must represent both the nontrivial sign structure and correlations that are not naturally aligned with a causal ordering.

We consider both one- and two-dimensional settings. Results for the one-dimensional chains at $J_1 = 1$ and $J_2 = 0.4$ or 0.5 are presented in the Appendix A. Here, we focus on the two-dimensional $L = 10$ square lattice with $J_2/J_1 = 0.5$ under periodic boundary conditions.

Results in this setting reveal an important architectural distinction. In Figure 4 (a), a plain causal decoder reaches the accuracy of the PWO autoregressive reference (Duque et al., 2026) but remains well above the ViT-NQS energy. Adding a configuration-independent site encoder leads to a lower variational energy, even though this encoder receives no spin configuration: it constructs a global memory solely from learnable site embeddings and passes it to the decoder through cross-attention.

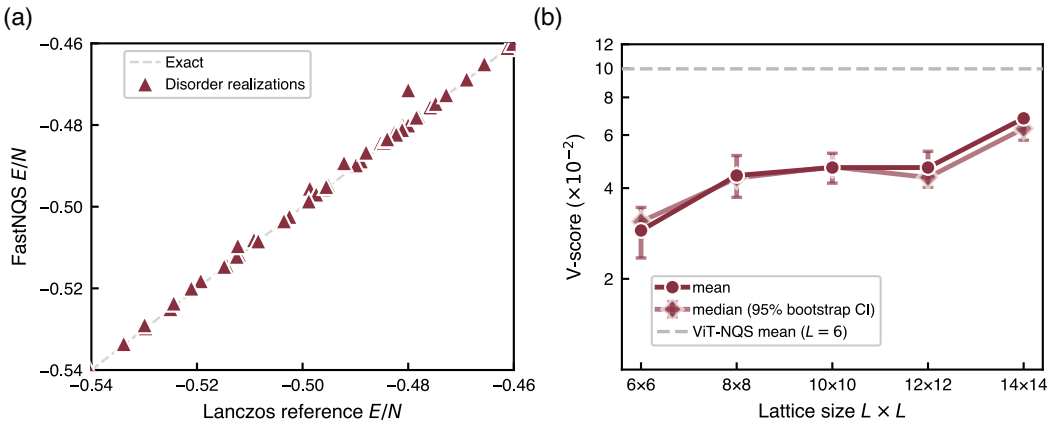


Figure 5: **FastNQS on the disordered random-bond Heisenberg model.** (a) FastNQS ground-state energies per site versus Lanczos reference energies for $n = 50$ disorder realizations at $L = 4$. The dashed line denotes exact agreement. The mean absolute relative energy error is $(1.53 \pm 2.77) \times 10^{-3}$ (mean $\pm$ standard deviation across realizations). (b) Mean and median V-scores over 10 independent disorder realizations at each lattice size. Circles denote the mean; diamonds and their error bars denote the median and its 95% bootstrap confidence interval. The gray dashed line indicates the mean V-score of the ViT-NQS baseline at $L = 6$, on the same 10 disorder realizations.

Increasing the batch size from 2048 to 4096 further improves the converged energy. These results suggest that a global representation of lattice geometry, rather than access to additional spin information, is an important missing ingredient for causal NQS. It also resonates with the observation of (Rende & Viteritti, 2025) that the effectiveness of transformer wave functions depends critically on architectural treatment of spatial structure, not simply on standard attention components.

The encoder-decoder model narrows the gap to the symmetry-aware ViT-NQS reference. A likely remaining factor is translational symmetry: exact symmetry enforcement is not straightforward for a sequential autoregressive factorization, because causal sampling selects an ordering. Developing a symmetry-compatible autoregressive construction is therefore a promising direction.

Finally, Figure 4 (b) and (c) compare the end-to-end VMC cost of FastNQS and ViT-NQS across batch sizes from 1024 to 6144, using nearly matched parameter counts (337,176 versus 339,748) on an RTX 5090. At batch size 2048, FastNQS requires $23.6\times$ less wall-clock time per VMC optimization step (0.982 s versus 23.198 s) while using $2.64\times$ less peak memory (3.43 GiB versus 9.05 GiB). This advantage persists as the Monte Carlo workload grows: at batch size 6144, ViT-NQS requires 68.55 s per VMC optimization step and reaches 18.61 GiB peak memory, compared with 2.66 s and 5.12 GiB, respectively, for FastNQS. The steep cost growth of conventional sampling and second-order optimization leaves increasingly little headroom for larger systems; FastNQS substantially lowers this resource barrier.

5.4 Random-bond Heisenberg model

We consider the random-bond Heisenberg model

$$H_{\text{rand}} = \sum_{\langle i,j \rangle} J_{ij}\, \mathbf{S}_i \cdot \mathbf{S}_j, \tag{9}$$

where the nearest-neighbor couplings satisfy $J_{ij} = +1$ with probability p and $J_{ij} = -1$ with probability $1 - p$. We set $p = 0.5$ and impose periodic boundary conditions. This choice introduces strong frustration through competing ferromagnetic and antiferromagnetic interactions.

We evaluate FastNQS using complementary benchmarks at small and large system sizes. For the 4×4 lattice, where accurate Lanczos calculations remain feasible, we directly compare the variational ground-state energy with the Lanczos reference across 50 independent disorder realizations. For larger lattices, where exact reference energies are no longer readily available, we instead report

the V-score (Wu et al., 2024),

$$V = N \frac{\langle \hat{H}^2 \rangle - \langle \hat{H} \rangle^2}{\langle \hat{H} \rangle^2}, \tag{10}$$

where $N = L^2$ is the number of sites, $\langle \hat{H} \rangle$ is the variational energy. The V-score is a dimensionless variance-based metric that has been shown empirically to correlate with the relative energy error across diverse quantum many-body systems (Wu et al., 2024). We thus report both the mean and median V-scores over 10 independent disorder realizations for each lattice size from 6×6 to 14×14.

Figure 5(a) shows close agreement between the FASTNQS and Lanczos energies for the 4×4 systems, with the points concentrated near the identity line over the full range of reference energies. Across the 50 disorder realizations, the mean absolute relative energy error is $(1.53 \pm 2.77) \times 10^{-3}$. Figure 5(b) extends the benchmark to larger lattices using the V-score. At $L = 6$, we additionally compare FASTNQS with a ViT-NQS baseline trained and evaluated on the same 10 disorder realizations; see Appendix D for details. Both the mean and median V-scores of FASTNQS remain below 7×10^{-2} across all tested system sizes, with no pronounced separation between the two statistics. Together, these results indicate that FASTNQS maintains consistent variational quality across different disorder realizations and up to the largest 14×14 lattice considered here.

6 Discussion and Limitations

Our results show that FASTNQS can be achieved by jointly addressing representation, sampling, wavefunction evaluation, and optimization across the full VMC framework. The spatial positional encoder preserves global lattice structure under causal decoding; KV caching avoids repeated evaluation of generated prefixes during ancestral sampling; FlashAttention reduces memory traffic and intermediate storage during batched wavefunction evaluation; and Muon provides a matrix-structured first-order optimization route without the linear-system solve required by SR. Together, these components reduce complementary costs across the VMC framework, leading to substantial gains in end-to-end efficiency. Across the systems studied here, FASTNQS remains competitive in variational accuracy while reducing computational cost. On the matched 10×10 J_1-J_2 benchmark, it reduces the end-to-end VMC-update time by $23.6\times$ relative to ViT-NQS; at $L = 32$ and batch size 2048, it reduces peak forward-pass memory by a factor of 394 relative to minGRU.

A remaining challenge is to incorporate lattice symmetries, particularly translational symmetry, without sacrificing the direct-sampling advantages of autoregressive wavefunctions. Their sequential factorization fixes an ordering of lattice sites, while conventional projective symmetrization does not preserve the normalization structure required for direct autoregressive sampling (Bortone et al., 2024). Architectural constraints and broken translational symmetry may limit the expressivity of autoregressive NQS, including autoregressive transformers (Chen et al., 2026).

Normalization-preserving alternatives have been developed for autoregressive RNNs by averaging over symmetry-related configurations while retaining direct sampling (Hibat-Allah et al., 2020; 2022). Such constructions incur additional evaluations and retain the ordering inherent to the autoregressive factorization, highlighting a structural trade-off between symmetry enforcement and direct autoregressive sampling. Combining symmetry-aware autoregressive constructions with the computational advantages of FASTNQS therefore remains an important direction.

To alleviate the additional computational overhead of symmetry-aware extensions, the efficiency of FASTNQS can provide greater headroom for evaluating symmetry-related configurations while also enabling studies at larger system sizes. The present two-dimensional J_1-J_2 experiments use the 10×10 lattice as a benchmark, leaving the maximum tractable system size (Viteritti et al., 2026) as an open scaling question that demands more computational resources. Together, these directions would enable more systematic finite-size scaling and studies of long-range correlations and critical behavior while improving the compatibility of scalable autoregressive NQS with lattice symmetries.

AI use statement

In this work, we used generative AI tools only to assist with drafting, language editing, and polishing the manuscript. We did not use generative AI tools for method development, experimental design, implementation, data processing, or interpretation of the results. The authors reviewed and revised

all AI-assisted text and verified all scientific claims and references. We take full responsibility for the final content of this work, including any text produced with the assistance of generative AI.

ETHICS STATEMENT

This work develops and evaluates computational methods for neural quantum states using numerical simulations of quantum spin systems. It does not involve human subjects, personal or sensitive data, or experiments on living subjects. We are not aware of any direct ethical risks arising from the methods or experiments presented in this work. We have aimed to report the methodology, experimental settings, reference data, and results accurately and transparently to support reproducibility and research integrity.

REPRODUCIBILITY STATEMENT

To support reproducibility, we provide anonymized source code as supplementary material, including the model implementation and the scripts used for training and evaluation. Section 4 describes the proposed model, autoregressive learning procedure, sampling and evaluation methods, and optimization strategy. Section 5 reports the benchmark settings, experimental protocols, and evaluation results. Appendix E provides additional implementation-level details of the encoder, decoder, attention mechanisms, positional representations, and output head. Appendix F defines the hyperparameter notation and summarizes the model and optimization configurations used for each physical system. Together, these descriptions and the supplementary code provide the information needed to reproduce the reported experiments.

REFERENCES

Adil Attar, Amine M. Aboussalah, and Mohamed Hibat-Allah. Quantum geometric tensor preconditioning for stable training of recurrent neural quantum states. arXiv:2608.18065, 2026. URL https://arxiv.org/abs/2608.18065.

Massimo Bortone, Yannic Rath, and George H. Booth. Impact of conditional modelling for a universal autoregressive quantum state. *Quantum*, 8:1245, 2024. doi: 10.22331/q-2024-02-08-1245. URL https://doi.org/10.22331/q-2024-02-08-1245.

Giuseppe Carleo and Matthias Troyer. Solving the quantum many-body problem with artificial neural networks. *Science*, 355(6325):602–606, 2017. doi: 10.1126/science.aag2302. URL https://www.science.org/doi/abs/10.1126/science.aag2302.

Ao Chen and Markus Heyl. Empowering deep neural quantum states through efficient optimization. *Nature Physics*, 20:1476–1481, 2024. doi: 10.1038/s41567-024-02566-1. URL https://www.nature.com/articles/s41567-024-02566-1.

Ao Chen, Vighnesh Dattatraya Naik, and Markus Heyl. Convolutional transformer wave functions. *Physical Review Research*, 8:L022040, Jun 2026. doi: 10.1103/7xwp-25y9. URL https://link.aps.org/doi/10.1103/7xwp-25y9.

Kenny Choo, Titus Neupert, and Giuseppe Carleo. Two-dimensional frustrated J_1–J_2 model studied with neural network quantum states. *Physical Review B*, 100(12):125124, 2019. doi: 10.1103/PhysRevB.100.125124. URL https://doi.org/10.1103/PhysRevB.100.125124.

Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning. In *International Conference on Learning Representations*, pp. 35549–35562, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/98ed250b203d1ac6b24bbcf263e3d4a7-Paper-Conference.pdf.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. FlashAttention: Fast and memory-efficient exact attention with io-awareness. In *Advances in Neural Information Processing Systems*, volume 35 of *NeurIPS 2022*, pp. 16344–16359, 2022. URL https://papers.neurips.cc/paper_files/paper/2022/hash/67d57c32e20fd0a7a302cb81d36e40d5-Abstract-Conference.html.

Juan Agustín Duque, Sergio García Heredia, Vinicius Hernandes, Eliška Greplová, Thomas Spriggs, Aaron Courville, and Anna Dawid. One more time: Revisiting neural quantum states from a reinforcement learning perspective, 2026. URL https://arxiv.org/abs/2607.02292.

Leo Feng, Frederick Tung, Mohamed Osama Ahmed, Yoshua Bengio, and Hossein Hajimirsadeghi. Were RNNs all we needed?, 2024. URL https://arxiv.org/abs/2410.01201.

Gil Goldshlager, Nilin Abrahamsen, and Lin Lin. A Kaczmarz-inspired approach to accelerate the optimization of neural network wavefunctions. *Journal of Computational Physics*, 516:113351, 2024. doi: 10.1016/j.jcp.2024.113351. URL https://doi.org/10.1016/j.jcp.2024.113351.

Johnnie Gray. quimb: A python package for quantum information and many-body calculations. *Journal of Open Source Software*, 3(29):819, 2018. doi: 10.21105/joss.00819. URL https://doi.org/10.21105/joss.00819.

Mohamed Hibat-Allah, Martin Ganahl, Lauren E. Hayward, Roger G. Melko, and Juan Carrasquilla. Recurrent neural network wave functions. *Physical Review Research*, 2:023358, Jun 2020. doi: 10.1103/PhysRevResearch.2.023358. URL https://link.aps.org/doi/10.1103/PhysRevResearch.2.023358.

Mohamed Hibat-Allah, Roger G. Melko, and Juan Carrasquilla. Supplementing recurrent neural network wave functions with symmetry and annealing to improve accuracy. arXiv:2207.14314, 2022. URL https://arxiv.org/abs/2207.14314.

Keller Jordan, Yuchen Jin, Vlado Boza, et al. Muon: An optimizer for hidden layers in neural networks. https://kellerjordan.github.io/posts/muon/, 2024.

Jicheol Kim, Dongkyu Kim, and Dong-Hee Kim. Neural-network quantum-state study of the long-range antiferromagnetic ising chain. *Physical Review E*, 109:064123, Jun 2024. doi: 10.1103/PhysRevE.109.064123. URL https://link.aps.org/doi/10.1103/PhysRevE.109.064123.

Hannah Lange, Anka Van de Walle, Atiye Abedinnia, and Annabelle Bohrdt. From architectures to applications: a review of neural quantum states. *Quantum Science and Technology*, 9(4): 040501, 2024. doi: 10.1088/2058-9565/ad7168. URL https://iopscience.iop.org/article/10.1088/2058-9565/ad7168.

Jingyuan Liu, Jianlin Su, Xingcheng Yao, Zhejun Jiang, Guokun Lai, Yulun Du, Yidao Qin, Weixin Xu, Enzhe Lu, Junjie Yan, et al. Muon is scalable for llm training. *arXiv:2502.16982*, 2025. URL https://arxiv.org/abs/2502.16982.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Yiming Lu, Sriram Bharadwaj, Dikshant Rathore, and Di Luo. Information-theoretic scaling laws of neural quantum states. arXiv:2603.23468, 2026. URL https://arxiv.org/abs/2603.23468.

Chanchal K. Majumdar and Dipan K. Ghosh. On next-nearest-neighbor interaction in linear chain. I. *Journal of Mathematical Physics*, 10(8):1388–1398, 1969a. doi: 10.1063/1.1664978. URL https://doi.org/10.1063/1.1664978.

Chanchal K. Majumdar and Dipan K. Ghosh. On next-nearest-neighbor interaction in linear chain. II. *Journal of Mathematical Physics*, 10(8):1399–1402, 1969b. doi: 10.1063/1.1664979. URL https://doi.org/10.1063/1.1664979.

Ejaaz Merali, Mohamed Hibat-Allah, Mohammad Kohandel, Richard T. Scalettar, and Ehsan Khatami. Parallel scan recurrent neural quantum states for scalable variational monte carlo, 2026. URL https://arxiv.org/abs/2605.13807.

Rajah P. Nutakki, Ahmedeo Shokry, and Filippo Vicentini. Design principles of deep translationally-symmetric neural quantum states for frustrated magnets. *Physical Review Research*, 7:043099, 2025. doi: 10.1103/ybgv-35jm. URL https://doi.org/10.1103/ybgv-35jm.

Pierre Pfeuty. The one-dimensional ising model with a transverse field. *Annals of Physics*, 57(1): 79–90, 1970. doi: 10.1016/0003-4916(70)90270-8. URL https://www.sciencedirect.com/science/article/abs/pii/0003491670902708.

Reiner Pope, Sholto Douglas, Aakanksha Chowdhery, Jacob Devlin, James Bradbury, Jonathan Heek, Kefan Xiao, Shivani Agrawal, and Jeff Dean. Efficiently scaling transformer inference. In *Proceedings of Machine Learning and Systems*, volume 5, pp. 606–624, 2023. URL https://proceedings.mlsys.org/paper_files/paper/2023/hash/c4be71ab8d24cdfb45e3d06dbfca2780-Abstract-mlsys2023.html.

Riccardo Rende and Luciano Loris Viteritti. Are queries and keys always relevant? a case study on transformer wave functions. *Machine Learning: Science and Technology*, 6(1):010501, 2025. URL https://doi.org/10.1088/2632-2153/ada1a0.

Riccardo Rende, Sebastian Goldt, Federico Becca, and Luciano Loris Viteritti. Fine-tuning neural network quantum states. *Physical Review Research*, 6(4):043280, 2024a. doi: 10.1103/PhysRevResearch.6.043280. URL https://doi.org/10.1103/PhysRevResearch.6.043280.

Riccardo Rende, Luciano Loris Viteritti, Lorenzo Bardone, Federico Becca, and Sebastian Goldt. A simple linear algebra identity to optimize large-scale neural network quantum states. *Communications Physics*, 7(1):260, Aug 2024b. ISSN 2399-3650. doi: 10.1038/s42005-024-01732-4. URL https://doi.org/10.1038/s42005-024-01732-4.

Jonas B. Rigo, Björn J. Wurst, Rajah Nutakki, Markus Schmitt, and Dante Kennes. Neural quantum states in condensed matter: advances, best practices, and prospects. arXiv preprint arXiv:2608.21291, 2026. URL https://arxiv.org/abs/2608.21291.

Christopher Roth, Attila Szabó, and Allan H. MacDonald. High-accuracy variational Monte Carlo for frustrated magnets with deep neural networks. *Physical Review B*, 108:054410, 2023. doi: 10.1103/PhysRevB.108.054410. URL https://doi.org/10.1103/PhysRevB.108.054410.

Anders W. Sandvik. High-precision ground state parameters of the two-dimensional spin-1/2 heisenberg model on the square lattice. *Journal of Statistical Mechanics: Theory and Experiment*, 2026(4):043101, 2026. doi: 10.1088/1742-5468/ae4f80. URL https://iopscience.iop.org/article/10.1088/1742-5468/ae4f80.

Or Sharir, Yoav Levine, Noam Wies, Giuseppe Carleo, and Amnon Shashua. Deep autoregressive models for the efficient variational simulation of many-body quantum systems. *Physical Review Letters*, 124:020503, Jan 2020. doi: 10.1103/PhysRevLett.124.020503. URL https://link.aps.org/doi/10.1103/PhysRevLett.124.020503.

Sandro Sorella. Generalized lanczos algorithm for variational quantum monte carlo. *Physical Review B*, 64:024512, Jun 2001. doi: 10.1103/PhysRevB.64.024512. URL https://link.aps.org/doi/10.1103/PhysRevB.64.024512.

Kyle Sprague and Stefanie Czischek. Variational Monte Carlo with large patched transformers. *Communications Physics*, 7:90, 2024. doi: 10.1038/s42005-024-01584-y. URL https://www.nature.com/articles/s42005-024-01584-y.

Attila Szabó and Claudio Castelnovo. Neural network wave functions and the sign problem. *Physical Review Research*, 2:033075, 2020. doi: 10.1103/PhysRevResearch.2.033075. URL https://doi.org/10.1103/PhysRevResearch.2.033075.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is All You Need. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Filippo Vicentini, Damian Hofmann, Attila Szabó, Dian Wu, Christopher Roth, Clemens Giuliani, Gabriel Pescia, Jannes Nys, Vladimir Vargas-Calderón, Nikita Astrakhantsev, and Giuseppe Carleo. NetKet 3: Machine learning toolbox for many-body quantum systems. *SciPost Physics Codebases*, pp. 7, 2022. doi: 10.21468/SciPostPhysCodeb.7. URL https://doi.org/10.21468/SciPostPhysCodeb.7.

Luciano Loris Viteritti, Riccardo Rende, and Federico Becca. Transformer variational wave functions for frustrated quantum spin systems. *Physical Review Letters*, 130(23):236401, 2023. doi: 10.1103/PhysRevLett.130.236401. URL https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.130.236401.

Luciano Loris Viteritti, Riccardo Rende, Subir Sachdev, and Giuseppe Carleo. Approaching the thermodynamic limit with neural-network quantum states. arXiv preprint arXiv:2602.02665, 2026. URL https://arxiv.org/abs/2602.02665.

Jie Wang, Jacopo Surace, Irénée Frérot, Benoît Legat, Marc-Olivier Renou, Victor Magron, and Antonio Acín. Certifying ground-state properties of many-body systems. *Physical Review X*, 14 (3):031006, 2024. doi: 10.1103/PhysRevX.14.031006. URL https://doi.org/10.1103/PhysRevX.14.031006.

Tom Westerhout, Nikita Astrakhantsev, Konstantin S. Tikhonov, Mikhail I. Katsnelson, and Andrey A. Bagrov. Generalization properties of neural network approximations to frustrated magnet ground states. *Nature Communications*, 11:1593, 2020. doi: 10.1038/s41467-020-15402-w. URL https://www.nature.com/articles/s41467-020-15402-w.

Dian Wu, Riccardo Rossi, Filippo Vicentini, Nikita Astrakhantsev, Federico Becca, Xiaodong Cao, Juan Carrasquilla, Francesco Ferrari, Antoine Georges, Mohamed Hibat-Allah, Masatoshi Imada, Andreas M. Läuchli, Guglielmo Mazzola, Antonio Mezzacapo, Andrew Millis, Javier Robledo Moreno, Titus Neupert, Yusuke Nomura, Jannes Nys, Olivier Parcollet, Rico Pohle, Imelda Romero, Michael Schmid, J. Maxwell Silvester, Sandro Sorella, Luca F. Tocchio, Lei Wang, Steven R. White, Alexander Wietek, Qi Yang, Yiqi Yang, Shiwei Zhang, and Giuseppe Carleo. Variational benchmarks for quantum many-body problems. *Science*, 386(6719):296–301, 2024. doi: 10.1126/science.adg9774. URL https://www.science.org/doi/10.1126/science.adg9774.

APPENDIX CONTENTS

A Results on One-Dimensional J_1-J_2 model 16

B Optimizer Comparison on the Two-Dimensional J_1-J_2 Model 17

C Effect of Symmetry Operations During Two-Dimensional J_1 Training 18

D ViT-NQS Baseline for the Random-Bond Heisenberg Model 19

E Model Architecture Details 19

Patch Tokens and Concatenated Decoder Vectors 19

Encoder Spatial Attention 19

Decoder Causal Self-Attention and Cross-Attention 20

Output Head 20

F Model and Optimization Hyperparameters 20

One-dimensional Transverse Field Ising Model 20

Two-dimensional J_1 Heisenberg model 21

Frustrated J_1-J_2 Heisenberg model 21

Random-bond Heisenberg model 22

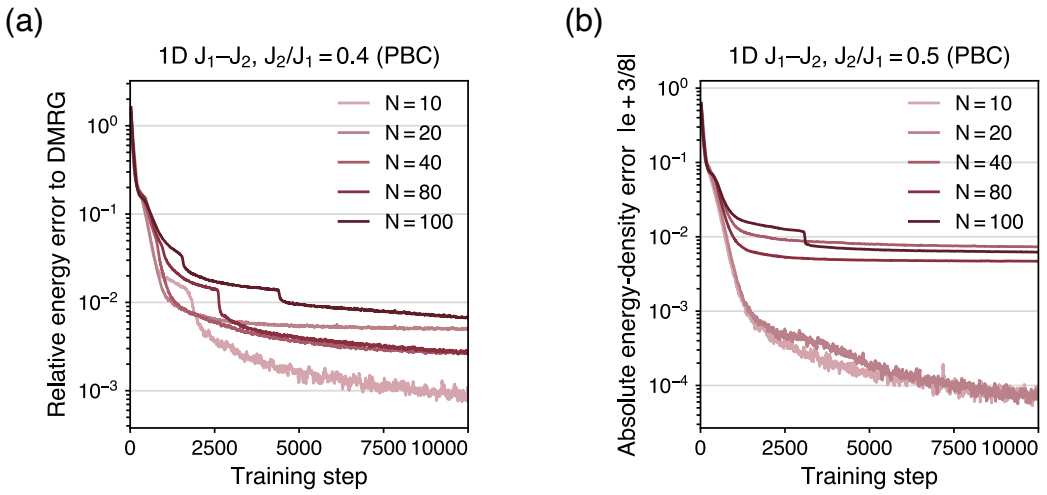


Figure 6: **One-dimensional J_1-J_2 convergence under PBC.** (a) Relative energy error with respect to finite-size DMRG references at $J_2/J_1 = 0.4$. (b) Absolute energy-density error $|e + 3/8|$ with respect to the exact Majumdar-Ghosh value at $J_2/J_1 = 0.5$. Both panels show Muon training for $N \in \{10, 20, 40, 80, 100\}$ over 10,000 steps; the curves are smoothed with a 25-step moving average. The $N = 10$ model has one decoder layer, while the larger models have two.

A Results on One-Dimensional J_1-J_2 model

Table 2: Finite-size PBC-DMRG reference energy densities for the one-dimensional J_1-J_2 chain at $J_1 = 1$ and $J_2 = 0.4$.

N	e_{DMRG}	Provenance
10	-0.384291850317	Local Quimb DMRG; converged through $\chi = 256$
20	-0.381271639917	Local Quimb DMRG; converged through $\chi = 256$
40	-0.38055	Table III of Wang et al. (2024)
80	-0.38038	Local Quimb DMRG; retained to five stable decimal places
100	-0.380387370	Table I of Rende et al. (2024a); $\chi_{\max} = 1000$

We evaluate the one-dimensional model under periodic boundary conditions at two frustration ratios and for chain lengths from $N = 10$ to $N = 100$. At $J_2/J_1 = 0.4$, where no finite-size closed-form energy is available, Figure 6(a) uses the finite-size relative error

$$\epsilon_{\mathrm{rel}}(N) = \frac{|e_\theta(N) - e_{\mathrm{DMRG}}(N)|}{|e_{\mathrm{DMRG}}(N)|}. \tag{11}$$

Table 2 records the numerical value and provenance of every denominator used in the plot. The $N = 10$ and $N = 20$ references were computed locally with two-site DMRG as implemented in Quimb 1.15.0 (Gray, 2018), using successive bond dimensions $32, 64, 128, 256$, truncation cutoffs $10^{-8}, 10^{-9}, 10^{-10}, 10^{-11}$, at most eight sweeps, and an energy tolerance of 10^{-10}. The periodic Hamiltonian MPO explicitly includes all interactions that cross the ring cut and was checked term by term against a dense small-ring Hamiltonian. The local $N = 80$ calculation used bond dimensions $32, 64, 128$, cutoffs $10^{-8}, 10^{-9}, 10^{-10}$, and tolerance 10^{-9}. Its raw final energy density was -0.380383774079, but because the final sweep changed this value by 3.75×10^{-6}, we retain only the five stable decimal places shown in the table.

The remaining two values are literature benchmarks with the same spin-$1/2$ operator normalization and finite PBC geometry. For $N = 40$, we use the PBC-DMRG result reported to five decimal places in Table III of Wang et al. (2024). For $N = 100$, we use the Table I result of Rende et al. (2024a), obtained with PBC-DMRG at maximum bond dimension $\chi = 1000$. In particular, none of the values in Table 2 is an OBC or thermodynamic-limit substitute.

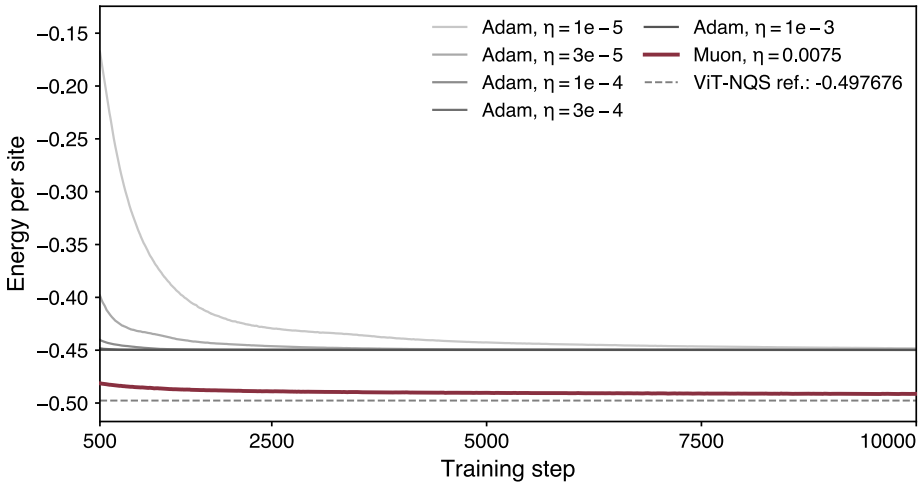


Figure 7: **Matched Muon-Adam optimizer comparison on the two-dimensional J_1-J_2 model.** Energy per site versus training step for the $L = 10$, $J_1 = 1$, $J_2 = 0.5$ PBC benchmark. Muon uses learning rate 7.5×10^{-3}, while Adam is evaluated at five learning rates from 10^{-5} to 10^{-3}. All other experimental settings are identical, and every run is trained for 10,000 steps. Curves are smoothed with a 25-step moving average; the dashed line denotes the matching ViT-NQS reference energy.

At the Majumdar-Ghosh point, $J_2/J_1 = 0.5$, every even chain has ground-state energy $E_0 = -3N/8$, and hence exact energy density $e_0 = -3/8$ (Majumdar & Ghosh, 1969a;b). Thus, $|e + 3/8| = |e - e_0|$ directly measures the absolute energy-density error in Figure 6 (b).

B Optimizer Comparison on the Two-Dimensional J_1-J_2 Model

Table 3: Matched optimizer ablation for the two-dimensional $L = 10$, $J_1 = 1$, $J_2 = 0.5$ model under PBC. Reported values are mean energies per site over the final 25 of 10,000 training steps; lower is better.

Optimizer	Learning rate η	Final-25 mean energy per site
Muon	7.5×10^{-3}	$\mathbf{-0.491290}$
Adam	1×10^{-5}	-0.448232
Adam	3×10^{-5}	-0.449636
Adam	1×10^{-4}	-0.449709
Adam	3×10^{-4}	-0.449744
Adam	1×10^{-3}	-0.449751

We conduct a matched optimizer ablation on the two-dimensional $L = 10$, $J_1 = 1$, $J_2 = 0.5$ model under PBC. Denoting the learning rate by η, as in Figure 7, we compare Muon at its tuned value $\eta = 7.5 \times 10^{-3}$ against Adam over the logarithmically spaced sweep $\eta \in \{10^{-5}, 3 \times 10^{-5}, 10^{-4}, 3 \times 10^{-4}, 10^{-3}\}$. Across all runs, we hold fixed the PatchEncoder architecture, model initialization (seed 93018926), sampling procedure, batch size of 2,048, gradient clipping threshold of 1.0, and training horizon of 10,000 steps. Consequently, the optimizer and its learning rate are the only experimental variables.

Figure 7 shows the resulting energy trajectories, and Table 3 reports the mean energy per site over the final 25 training steps. All five Adam runs completed without numerical or runtime errors. Adam is relatively insensitive to its learning rate over the upper part of the tested range: the final mean energies for learning rates from 10^{-4} to 10^{-3} differ by less than 5×10^{-5}. Nevertheless, none approaches the Muon result. The best Adam setting, 10^{-3}, reaches -0.449751, whereas Muon reaches -0.491290, an energy reduction of 0.041539 per site.

Table 4: Ground-state energy per site e_0 of FASTNQS for the 2D J_1 Heisenberg model under OBC, before and after D_4-symmetry fine-tuning.

L	w/o D_4	+D_4 FT	QMC	Rel. Err. ($\times 10^{-4}$)	
				w/o D_4	+D_4 FT
6	-0.60347897	-0.60352039	-0.6035222	0.7163	0.0300
8	-0.61886328	-0.61901367	-0.6190371	2.8079	0.3785
10	-0.62823659	-0.62861091	-0.6286561	6.6731	0.7188
12	-0.63487679	-0.63511497	-0.6352007	5.0993	1.3497
16	-0.64316750	-0.64336008	-0.6435317	5.6594	2.6668
24	-0.65169251	-0.65182000	-0.6520251	5.1009	3.1456
32	-0.65619648	–	-0.6563289	2.0176	–

Table 5: Ground-state energy per site e_0 of FASTNQS for the 2D J_1 Heisenberg model under PBC, before and after D_4-symmetry fine-tuning.

L	w/o D_4	+D_4 FT	QMC	Rel. Err. ($\times 10^{-4}$)	
				w/o D_4	+D_4 FT
8	-0.67291015	-0.67340267	-0.67349005	8.6104	1.2974
10	-0.67068559	-0.67137170	-0.67155266	12.9114	2.6947
12	-0.66991329	-0.67036182	-0.67068192	11.4604	4.7728
16	-0.66943246	-0.66976982	-0.66997660	8.1218	3.0864
24	-0.66913843	-0.66941994	-0.66960434	6.9580	2.7539

Relative to the matching ViT-NQS reference energy -0.497676, the final absolute energy-density error is 0.006386 for Muon and 0.047925 for the best Adam run. Thus, under this controlled 10,000-step budget, Muon reduces the error by a factor of approximately 7.5 relative to the best learning rate found in the Adam sweep. The convergence curves further show that this advantage is not explained by an isolated final-step fluctuation: Muon enters and remains in a substantially lower-energy regime, while all Adam variants plateau near -0.45.

C Effect of Symmetry Exemplified in Two-Dimensional J_1 Model

We examine whether enforcing the point-group symmetry of the square lattice improves the variational energy of FASTNQS on the two-dimensional J_1 Heisenberg model. For each spin configuration $\mathbf{s}$, we form its orbit under the dihedral group D_4, consisting of four rotations and four reflections, and replace the model probability by the orbit average

$$p_{D_4}(\mathbf{s}) = \frac{1}{8} \sum_{g \in D_4} p_\theta(g\mathbf{s}). \tag{12}$$

Computing this average requires evaluating all eight symmetry-related configurations and therefore substantially increases the cost of each optimization step. We consequently train the autoregressive model without explicit D_4 symmetrization and apply the symmetry-averaged probability only during a subsequent fine-tuning stage. The "w/o D_4" and "+D_4 FT" columns below report the energies immediately before and after this stage, respectively. Relative errors in both tables are computed with respect to the corresponding QMC estimates from Sandvik (2026).

The fine-tuning stage consistently lowers the energy under both boundary conditions. For OBC (Table 4), the relative error decreases at every size for which a fine-tuned result is available: the error range changes from $[0.7163,\ 6.6731] \times 10^{-4}$ before fine-tuning to $[0.0300,\ 3.1456] \times 10^{-4}$ afterward. This corresponds to reductions by factors of 1.6 to 23.9, with the largest relative gain at $L = 6$. The $L = 32$ row is excluded from this comparison because no D_4-fine-tuned result is available.

The same trend holds more uniformly under PBC (Table 5). Across $L = 8$ to 24, D_4 fine-tuning reduces the relative error from $[6.9580,\ 12.9114] \times 10^{-4}$ to $[1.2974,\ 4.7728] \times 10^{-4}$, an improve-

ment by factors of 2.4 to 6.6. Thus, averaging over rotations and reflections provides a consistent accuracy gain even when it is introduced only after the initial training. Restricting the symmetry average to fine-tuning captures much of this benefit without paying its eight-configuration evaluation cost throughout the full optimization.

D ViT-NQS Baseline for the Random-Bond Heisenberg Model

For the $L = 6$ random-bond Heisenberg comparison in Figure 5(b), we constructed the ViT-NQS baseline based on the NetKet ViT implementation (Vicentini et al., 2022) and the architecture of Rende et al. (2024b). The model consists of four transformer layers with embedding dimension $d_{\text{model}} = 60$, 10 attention heads, and 2×2 spin patches. FastNQS and ViT-NQS contain 171,016 and 157,860 trainable parameters, respectively, resulting in comparable model sizes. Because the quenched disorder breaks translational symmetry for each individual realization, we disabled the translation-invariant factored-attention parameterization (`transl_invariant=False`), so that the attention parameters can depend explicitly on the patch positions.

Sampling was performed in the fixed-magnetization sector using a Metropolis exchange sampler. Each optimization step used 1024 parallel Markov chains and 1024 total samples, corresponding to one sample per chain. One sampling sweep consisted of $N = L^2$ exchange proposals per chain. We set `n_discard_per_chain=0`, such that no additional samples were discarded before collecting the batch at each optimization step, while the Markov-chain states were carried forward between successive steps.

The model was optimized for 6000 steps using stochastic gradient descent with a learning rate of 7.5×10^{-3}, together with stochastic reconfiguration using a diagonal shift of 10^{-4}. Both V-score monitoring during training and the final V-score evaluation used 1024 samples. For a paired comparison, ViT-NQS and FastNQS were trained and evaluated on exactly the same 10 disorder realizations, with identical bond configurations for each realization.

E Model Architecture Details

Figure 1 (b) summarizes the encoder-decoder architecture. Here we specify the data represented by each vector and distinguish the three attention operations used by the model.

Patch tokens and concatenated decoder vectors. For an $L \times L$ lattice with square patches of edge length p, each patch contains p^2 spins and is represented by one of $M = 2^{p^2}$ categorical tokens. The patch grid has side length $G = L/p$ and produces a raster-ordered sequence of $T = G^2$ tokens. To preserve the autoregressive factorization, the decoder input is shifted right: position t receives a begin-of-sequence token when $t = 1$, and otherwise the token for the preceding patch, s_{t-1}. Writing $s_0 \equiv s_{\text{BOS}}$, the "concatenated vectors" in Figure 1 (b) are

$$\mathbf{h}_t^{(0)} = \left[\mathbf{e}_{\text{tok}}(s_{t-1}); \mathbf{e}_{\text{pos}}^{\text{dec}}(t)\right], \tag{13}$$

where the learnable token and decoder-position embeddings have dimensions d_{tok} and d_{pos}, respectively, so that $d_{\text{model}} = d_{\text{tok}} + d_{\text{pos}}$. These decoder-position embeddings are distinct from the encoder embeddings described next.

Encoder spatial attention. The encoder receives no spin configuration. It starts from a learnable d_{model}-dimensional embedding $\mathbf{z}_{\mathbf{r}}$ at each location $\mathbf{r}$ on the $G \times G$ patch grid; these are the "learnable positional embeddings" shown at the bottom of the encoder in Figure 1 (b). Each encoder layer applies a multi-head spatial mixer with learnable, content-independent weights, rather than query-key dot-product attention. For every head h, the mixer parameterizes a $G \times G$ kernel indexed by relative displacement on the periodic patch grid. Cyclically shifting this kernel constructs the dense mixing matrix

$$A^{(h)}_{\mathbf{r},\mathbf{r}'} = \alpha^{(h)}_{(\mathbf{r}'-\mathbf{r}) \bmod G}, \qquad \widetilde{\mathbf{z}}^{(h)}_{\mathbf{r}} = \sum_{\mathbf{r}'} A^{(h)}_{\mathbf{r},\mathbf{r}'} \mathbf{V}^{(h)} \mathbf{z}_{\mathbf{r}'}. \tag{14}$$

Thus, every patch-grid location receives information from every other location, while the mixing weights are tied across translations. The mixing coefficients are neither input-dependent nor

softmax-normalized. The spatial mixer operates on pre-RMS-normalized representations, and its head outputs are concatenated and linearly projected before the first residual update. A second pre-RMS-normalized feed-forward block supplies the other residual update in each encoder layer. After the final RMS normalization, the G^2 patch-grid representations are mean-pooled into a single, configuration-independent global memory token.

Decoder causal self-attention and cross-attention. Each decoder layer contains the other two attention operations, with pre-normalization and residual connections. First, standard multi-head self-attention forms queries, keys, and values from the decoder states and applies a causal mask, so the state at position t can depend only on the shifted inputs through position t, equivalently on physical tokens $s_{<t}$. In addition to the learnable decoder-position vectors already concatenated at the input, two-dimensional toroidal rotary position embeddings are applied to the queries and keys. Half of each head's rotary frequencies encode the patch-grid column and half encode the row, using integer Fourier frequencies compatible with the periodic $G \times G$ grid.

Second, each decoder layer contains an unmasked cross-attention module whose memory is obtained by mean-pooling the encoder representations into a single global token. Because this configuration-independent memory encodes only lattice-position information, the module does not expose the decoder to ungenerated spins and therefore preserves the autoregressive factorization. During ancestral sampling, the self-attention keys and values are cached across generation steps, while the encoder memory and its layer-specific projections are computed once and reused across all decoding steps.

Output head. After the final decoder layer, LayerNorm is followed by two linear heads. The first produces amplitude logits $\mathbf{a}_t$, which define the normalized conditional probability

$$p_\theta(s_t = m \mid s_{<t}) = \mathrm{softmax}(\mathbf{a}_t)_m \tag{15}$$

and the corresponding log-magnitude

$$\ell_{t,m} = \frac{1}{2} \log p_\theta(s_t = m \mid s_{<t}). \tag{16}$$

For a complex wavefunction, the second head produces phase logits $\mathbf{b}_t$, which are mapped to bounded phases

$$\varphi_{t,m} = \pi \ \mathrm{softsign}(b_{t,m}). \tag{17}$$

The conditional wavefunction factor for candidate token m is therefore

$$\psi_\theta(s_t = m \mid s_{<t}) = \exp(\ell_{t,m} + i\varphi_{t,m}) \,. \tag{18}$$

Selecting the observed token at each position gives

$$\log \psi_\theta(\mathbf{s}) = \sum_{t=1}^{T} (\ell_{t,s_t} + i\varphi_{t,s_t}) \,. \tag{19}$$

Ancestral sampling uses only the normalized probabilities $p_\theta(s_t \mid s_{<t})$. In the positive-real setting, the phase head is not used.

F Model and Optimization Hyperparameters

This section summarizes the model configurations and optimization hyperparameters used for the four physical systems considered in our experiments. Table 6 defines the architectural notation shared by the system-specific tables below.

One-dimensional Transverse Field Ising Model. For the one-dimensional transverse-field Ising model (TFIM), we use the same model configuration across all system sizes. We set $J = -1$ and $h = 1$, corresponding to the critical point studied in our experiments. The model uses a patch length of 8, token dimension 16, positional embedding dimension 48, 2 transformer layers, and 4 attention heads. Training is performed for 4000 steps with a batch size of 1024. Muon is used for matrix-valued parameters with a learning rate of 10^{-3}, while AdamW with the same learning rate is applied to the remaining parameters.

Table 6: Architectural notation used in the hyperparameter tables.

Symbol	Meaning
p	Edge length of each square patch
L	Linear size (side length) of a two-dimensional lattice
N	Total number of lattice sites
n_{layers}	Number of stacked model layers
n_{heads}	Number of attention heads per layer
d_{model}	Dimension of the model's hidden representations
d_{ff}	Hidden dimension of the feed-forward network

Table 7: **Model hyperparameters for the two-dimensional J_1 Heisenberg model.**

Boundary	System size L	d_{token}	d_{pos}	n_{layers}	n_{heads}	Batch size
OBC	$6, 8, 10, 12, 16, 24, 32$	32	96	6	8	1,024
PBC	8	32	64	6	8	1,024
	$10, 12$	32	80	6	8	1,024
	$16, 24$	32	96	6	8	1,024

Two-dimensional J_1 Heisenberg model For the two-dimensional J_1 Heisenberg model, we use a token dimension of 32, 6 transformer layers, 8 attention heads, and a batch size of 1024 for all system sizes and boundary conditions. Under OBC, the positional embedding dimension is fixed to 96. Under PBC, we use positional embedding dimensions of 64, 80, and 96 for $L = 8$, $L \in \{10, 12\}$, and $L \in \{16, 24\}$, respectively.

For optimization, the initial training stage uses cosine learning-rate annealing with an initial learning rate on the order of 10^{-3}, with small adjustments across system sizes. During the subsequent fine-tuning stage, the learning rate is kept fixed at approximately 10^{-5}, again with minor size-dependent variations. All configurations use a batch size of 1024.

Frustrated J_1-J_2 Heisenberg model We study one-dimensional chains with $N \in \{10, 20, 40, 80, 100\}$ sites at $J_1 = 1$ and $J_2 \in \{0.4, 0.5\}$, as well as the two-dimensional $L \times L$ square lattice with $L = 10$, $J_1 = 1$, and $J_2 = 0.5$. All systems use periodic boundary conditions, and all FASTNQS models are optimized with Muon. For the two-dimensional energy-convergence experiment in Figure 4(a), FASTNQS is trained with batch sizes of 2,048 and 4,096. The cost comparison in Figure 4(b) and (c) uses FASTNQS and a parameter-matched ViT-NQS model with 337,176 and 339,748 trainable parameters, respectively, over batch sizes from 1,024 to 6,144; ViT-NQS is optimized with Adam in this benchmark. The ViT-NQS configuration reported in Table 8 pertains specifically to the cost comparison in panels (b) and (c), whereas the ViT-NQS energy reference in panel (a) is taken from Rende et al. (2024b).

Table 8: **Model and optimization hyperparameters for the one- and two-dimensional frustrated J_1-J_2 Heisenberg models.**

Model	Experiment	System size	Patch size p	n_{layers}	n_{heads}	d_{model}	d_{ff}	Batch size
FASTNQS	Figure 6	$N = 10$	–	1	2	18	72	2,048
		$N = 20$	–	2	2	18	72	2,048
		$N = 40$	–	2	2	18	72	2,048
		$N = 80$	–	2	2	18	72	2,048
		$N = 100$	–	2	2	18	72	2,048
FASTNQS	Figure 4(a)	$L = 10$	2	4	4	64	256	2,048 / 4,096
	Figure 4(b,c)	$L = 10$	2	4	4	64	256	1,024–6,144
ViT-NQS	Figure 4(b,c)	$L = 10$	2	7	4	68	272	1,024–6,144

Random-bond Heisenberg model For the random-bond Heisenberg model, we use the same architecture and optimization settings across all lattice sizes, as summarized in Table 9. The spin-token embedding and positional embedding have dimensions 8 and 40, respectively, giving $d_{\mathrm{model}} = 48$.

Table 9: **Hyperparameters for the random-bond Heisenberg model.**

Hyperparameter	Value
d_{model}	48
n_{layers}	4
n_{heads}	6
Patch size p	2
Initial learning rate	1×10^{-3}
Minimum learning rate	6×10^{-5}
Learning-rate schedule	Cosine
Training steps	10,000
Batch size	4,096