https://doi.org/10.15388/vu.thesis.989
https://orcid.org/0000-0001-7802-4334

VILNIUS UNIVERSITY
CENTER FOR PHYSICAL SCIENCES AND TECHNOLOGY

Eimantas Ledinauskas

Neural Quantum States for Many-Body Physics: Limitations and Alternative Optimization Approaches

DOCTORAL DISSERTATION

Natural Sciences,
Physics (N 002)

VILNIUS 2026

The dissertation was prepared between 2022 and 2026 at Vilnius University, Institute of Theoretical Physics and Astronomy.

Academic Supervisor – Prof. Habil. Dr. Egidijus Anisimovas (Vilnius University, Natural Sciences, Physics, N 002).

The Dissertation Defence Panel:

Chairman – Assoc. Prof. Dr. Jevgenij Chmeliov (Vilnius University, Natural Sciences, Physics, N 002).

Members:
Dr. Artūras Acus (Vilnius University, Natural Sciences, Physics, N 002),
Assoc. Prof. Dr. Kęstutis Aidas (Vilnius University, Natural Sciences, Physics, N 002),
Dr. Guillaume Guiglion (Max Planck Institute for Astronomy, Germany, Natural Sciences, Physics, N 002),
Assoc. Prof. Dr. Donatas Narbutis (Vilnius University, Natural Sciences, Physics, N 002).

The dissertation shall be defended at a public meeting of the Dissertation Defence Panel at 15:00 on 18th September 2026 in the Great Physics Auditorium of the Physics Faculty (Vilnius University). Address: Saulėtekio al. 9, Vilnius, Lithuania.

https://doi.org/10.15388/vu.thesis.989
https://orcid.org/0000-0001-7802-4334

VILNIAUS UNIVERSITETAS
FIZINIŲ IR TECHNOLOGIJOS MOKSLŲ CENTRAS

Eimantas Ledinauskas

Neuroninės kvantinės būsenos daugiadalelių sistemų fizikoje: apribojimai ir alternatyvūs optimizavimo metodai

DAKTARO DISERTACIJA

Gamtos mokslai,
Fizika (N 002)

VILNIUS 2026

Disertacija rengta 2022–2026 metais Vilniaus universitete, Teorinės fizikos ir astronomijos institute.

Mokslinis vadovas – prof. habil. dr. Egidijus Anisimovas (Vilniaus universitetas, gamtos mokslai, fizika, N 002).

Gynimo taryba:

Pirmininkas – doc. dr. Jevgenij Chmeliov (Vilniaus universitetas, gamtos mokslai, fizika, N 002).

Nariai:
dr. Artūras Acus (Vilniaus universitetas, gamtos mokslai, fizika, N 002),
doc. dr. Kęstutis Aidas (Vilniaus universitetas, gamtos mokslai, fizika, N 002),
dr. Guillaume Guiglion (Makso Planko astronomijos institutas, Vokietija, gamtos mokslai, fizika, N 002),
doc. dr. Donatas Narbutis (Vilniaus universitetas, gamtos mokslai, fizika, N 002).

Disertacija ginama viešame Gynimo tarybos posėdyje 2026 m. rugsėjo mėn. 18 d. 15 val. Vilniaus universiteto fizikos fakulteto didžiojoje fizikos auditorijoje. Adresas: Saulėtekio al. 9, Vilnius, Lietuva.

Contents

Abbreviations **9**

1 Introduction **10**
1.1 Context and motivation 10
1.2 Aims and novelty 12
1.3 Statements of the thesis 14
1.4 Thesis structure 15

2 Background **17**
2.1 Many-body quantum systems 17
2.1.1 Many-body wave function and Hilbert space 17
2.1.2 Spin lattice models 19
2.1.3 Tight-binding lattice models 29
2.2 Neural networks and deep learning 35
2.2.1 Multi-layer perceptrons 35
2.2.2 Convolutional neural networks 39
2.2.3 Transformer 43
2.2.4 Other NQS-relevant architectures 47
2.2.5 Optimization 48
2.3 Neural quantum states 50
2.3.1 Neural network as a wave function 50
2.3.2 Estimation of observables 52
2.3.3 Variational Monte Carlo 53
2.3.4 Energy gradient 54
2.3.5 Imaginary time evolution 56
2.3.6 Stochastic reconfiguration 57
2.3.7 Minimum-step SR and kernel trick 60
2.3.8 Markov chain Monte Carlo sampling 63
2.3.9 Sampling with autoregressive NQS 64
2.3.10 Enforcing symmetries 69
2.3.11 Multi-state NQS for excited states 72

2.4 Software and implementation 75

3 Supervised imaginary time evolution **78**
3.1 Introduction 78
3.2 SITE method description 79
3.2.1 Target construction by a discrete imaginary-time step . 80
3.2.2 Overlap loss and gradient estimator 81
3.2.3 Adaptive imaginary-time step 83
3.2.4 Relation to energy minimization 84
3.2.5 SITE algorithm 85
3.3 Testing SITE with numerical experiments 86
3.3.1 Neural network architecture 87
3.3.2 Sampling of basis states 90
3.3.3 Implementation 90
3.3.4 Symmetries of J_1–J_2 Heisenberg model 90
3.3.5 Hyperparameters and model selection 91
3.3.6 Comparison with direct energy minimization 92
3.3.7 Scaling and benchmarking against DMRG 93
3.3.8 Energy accuracy limits and comparison to other NQS results 95
3.4 Conclusions 97

4 Universal performance gap in HBHM **99**
4.1 Introduction 99
4.2 Hofstadter–Bose–Hubbard model 101
4.3 Methods 102
4.3.1 Neural-network architectures 102
4.3.2 Ground-state approximation with NQS 103
4.3.3 State sampling 104
4.4 Results 104
4.4.1 Drop of NQS performance in the strong-flux regime . 104
4.4.2 How many parameters are required? 106
4.4.3 Assessing potential causes for the performance drop . 107
4.4.4 Experiments with NQS modifications 115
4.5 Conclusions 116

5 Neural importance resampling **119**
5.1 Introduction 119

5.2 Sampling targets in VMC 120
5.3 Neural importance resampling 121
5.3.1 Importance resampling 121
5.3.2 Autoregressive proposal network 122
5.3.3 Training objective: forward vs. backward Kullback–Leibler divergence 122
5.3.4 Adaptive proposal network retraining 124
5.3.5 Related work 126
5.4 Numerical experiments 127
5.4.1 Transverse-field Ising model 127
5.4.2 Multi-state NQS and implementation details 127
5.4.3 Impact of the sampler: NIR vs. MCMC on the TFIM . 128
5.4.4 Comparing NIR with DMRG 131
5.4.5 Benchmarking on the J_1–J_2 Heisenberg model 132
5.5 Conclusions 133

6 The toric code as a case study for neural quantum states 135
6.1 Introduction 135
6.2 Honeycomb toric code in a uniform field 136
6.3 Multi-state variational formulation 137
6.4 Practical obstacles for NQS 139
6.5 Inductive biases and sampling improvements 139
6.6 Results and interpretation 142
6.6.1 Standard versus improved training 142
6.6.2 Benchmark against exact diagonalization 143
6.6.3 Ablation of the four improvements 144
6.7 Conclusions 147

7 Conclusions 148
7.1 Main findings 148
7.2 Broader lessons 151
7.3 Outlook 153

Bibliography 155

Santrauka lietuvių kalba 170

Curriculum vitae 188

Acknowledgements **189**

List of publications **190**

List of presentations **191**

Abbreviations

BHM	Bose–Hubbard model
CNN	Convolutional neural network
DMRG	Density-matrix renormalization group
ED	Exact diagonalization
ESS	Effective sample size
FCI	Fractional Chern insulator
HBHM	Hofstadter–Bose–Hubbard model
IR	Importance resampling
ITE	Imaginary time evolution
JSD	Jensen–Shannon divergence
KLD	Kullback–Leibler divergence
MCMC	Markov chain Monte Carlo
MHA	Multi-head attention
MLP	Multi-layer perceptron
MPS	Matrix-product state
MSE	Mean-squared error
NES-VMC	Natural excited states variational Monte Carlo
NIR	Neural importance resampling
NQS	Neural quantum state
QMC	Quantum Monte Carlo
RBM	Restricted Boltzmann machine
RNN	Recurrent neural network
SITE	Supervised imaginary time evolution
SNN	Sampling neural network
SR	Stochastic reconfiguration
TFIM	Transverse-field Ising model
VMC	Variational Monte Carlo

1 Introduction

1.1 Context and motivation

Many of the deepest open problems in condensed matter physics and quantum information involve strongly correlated quantum many-body systems. Phenomena such as high-temperature superconductivity, frustrated magnetism, quantum spin liquids, fractional quantum Hall states, and other forms of topological order cannot be understood in terms of isolated particles alone [1–6]. Instead, they emerge from the collective interplay of quantum coherence, interactions, and symmetry. Lattice models provide a controlled setting for studying them: they retain the relevant local degrees of freedom while abstracting away high-energy details, and closely related Hamiltonians can be engineered in ultracold atoms, programmable Rydberg arrays, and quantum-information architectures [7–10]. The spin systems, Bose–Hubbard-type models, and toric code studied here are simple enough to define precisely yet rich enough to expose the central difficulties of the quantum many-body problem.

The numerical difficulty of these many-body models is already evident from the size of the Hilbert space. For N local degrees of freedom with local dimension d, a generic state requires d^N complex amplitudes in a product basis. This exponential growth confines exact diagonalization (ED) to small systems. Quantum Monte Carlo (QMC) can reach large lattices when a sign-problem-free representation is available, but frustration, fermionic statistics, or gauge fields can lead to the sign problem: sign or phase cancellations that make sampling exponentially difficult [11]. Tensor-network methods, especially matrix-product states (MPS) and density-matrix renormalization group (DMRG), are extremely powerful in one dimension and on quasi-one-dimensional geometries, but their cost grows rapidly when genuinely two-dimensional entanglement or long-range structure must be retained [12–14]. Two-dimensional frustrated magnets and bosonic systems with strong gauge fields can therefore challenge both approaches, motivating complementary methods.

At the same time, physically relevant states are not arbitrary vectors in Hilbert

space. Ground states and low-lying states of local Hamiltonians are constrained by locality, symmetries, conservation laws, correlation length, and entanglement structure [15]. This observation motivates a variational *ansatz* (plural: ansätze): a family of trial states $|\psi(\boldsymbol{\theta})\rangle$ whose parameters $\boldsymbol{\theta}$ are adjusted to approximate the target. Instead of storing every amplitude independently, one thus searches within a structured family of states whose number of parameters grows much more slowly than the Hilbert-space dimension. Here "ansatz", "variational wave function", and "trial state" refer to this parameterized family or, when clear from context, to one of its members. The central question is then whether the chosen family is both expressive enough to contain accurate approximations to the target states and tractable enough that those approximations can be found reliably. Neural quantum states (NQS) are one such family: they use neural networks as flexible, trainable parameterizations of many-body wave functions.

Neural networks are natural candidates because modern deep learning combines efficient backpropagation [16], improved training strategies [17, 18], GPU acceleration, and architectures with useful inductive biases. Applications from image recognition [19] to protein-structure prediction [20] demonstrate the value of matching these ingredients to the problem structure. For NQS, the analogous requirement is to match the architecture, objective, optimizer, and sampler to the quantum state.

The modern NQS literature is usually traced to the work of Carleo and Troyer, who used a restricted Boltzmann machine (RBM) as a variational representation of spin-system wave functions and optimized it for ground states and dynamics [21]. RBMs originated as probabilistic energy-based models [22, 23], while VMC already optimized compact physical ansätze from stochastic estimates [24, 25], and earlier work had applied neural networks to quantum eigenvalue problems [26]. Carleo and Troyer combined these ideas with modern stochastic optimization, particularly stochastic reconfiguration (SR) [27–29], to obtain competitive results on interacting spin benchmarks [30, 31]. Since then, the field has expanded rapidly from RBMs to feed-forward networks, convolutional networks, recurrent and autoregressive models, transformers, and symmetry-adapted architectures. Applications now range from ground-state search to finite-temperature states, time evolution, excited states, quantum state tomography, and open-system dynamics [30–32].

NQS are promising because they sit at a useful intersection of physics and machine learning. From the physics side, they are variational wave functions and can be optimized by the variational principle, with observables estimated through

VMC. From the machine-learning side, they inherit automatic differentiation, scalable optimization, and a large design space of architectures. In frustrated two-dimensional spin systems, this combination has already produced ground-state energies that are competitive with leading variational calculations for the J_1–J_2 Heisenberg model and related benchmarks [33–35]. Unlike MPS or DMRG, a generic NQS is not tied to a quasi-one-dimensional bond geometry; unlike ED, it can use far fewer parameters than the Hilbert-space dimension; and unlike a fixed analytical ansatz, it can adapt to correlations not known in advance. These advantages explain why NQS have become one of the most active variational approaches to quantum many-body physics. In practice, they often complement rather than replace established physics-based ansätze, for example in hybrid projected or pair-product constructions. Yet their flexibility is also the source of the main scientific problem addressed in this thesis. The performance of an NQS calculation depends not only on expressiveness but also on the optimization landscape, Monte Carlo sampling, symmetries and constraints, and the representation of sign or phase structure. In some regimes, a standard NQS pipeline works well. In others, it fails for reasons that are not immediately visible from the variational energy alone. In the systems considered here, these issues appear most clearly as three recurring bottlenecks, introduced in the next section. Understanding these failure modes is essential if NQS are to become reliable tools rather than impressive but uneven demonstrations.

1.2 Aims and novelty

The overarching aim of this thesis is to make NQS more reliable as variational tools for lattice many-body systems by determining whether optimization, representation, or sampling limits accuracy in a given physical regime, and by developing methods that target the identified bottleneck. The emphasis is therefore diagnostic as much as constructive: rather than minimizing the energy on a single benchmark, the work aims to expose the mechanisms behind NQS success and failure and to turn them into transferable methodology.

These three bottlenecks, around which the work is organized, are:

- *Trainability*: even an expressive ansatz can be hard to optimize when stochastic gradients are noisy and the variational geometry is poorly conditioned.
- *Representational bias*: a network with many parameters may still be inefficient if its architecture does not match the sign, phase, locality, or

topological structure of the target state.

- *Sampling*: expectation values are estimated from configurations drawn with probability proportional to the squared wave-function amplitude; strong autocorrelations reduce the effective sample size, while poor equilibration or missed regions of configuration space can bias these estimates and destabilize training.

To address these bottlenecks, the thesis pursues the following objectives:

1. Develop an optimization scheme for NQS that preserves the physically motivated direction of imaginary-time evolution while avoiding the explicit metric-tensor inversion of stochastic reconfiguration, and assess its stability and accuracy on the frustrated J_1–J_2 Heisenberg model, where the evidence points to trainability, rather than expressiveness, as the dominant limitation for the generic architectures used here.

2. Establish, in a controlled setting small enough for exact diagonalization, when and why generic NQS architectures fail, using the hard-core Hofstadter–Bose–Hubbard model (HBHM) to separate optimization and sampling effects from intrinsic representational limitations.

3. Introduce a sampling strategy that decouples the variational ansatz from the sampler, remains usable when local Markov chains mix poorly, and supports multi-state determinant NQS for which no exact autoregressive sampler of the full target is available.

4. Demonstrate, on the honeycomb toric code (a setting that combines all three bottlenecks), how nonlocal stabilizer and loop structure can be built into both the ansatz and the sampler, and determine which such inductive biases are necessary to represent topologically ordered low-energy manifolds.

These objectives yield four contributions whose novelty lies both in the techniques and in what they reveal about failures of the NQS pipeline; the numerical results are stated in the next section. First, supervised imaginary time evolution (SITE, [P1]) treats an imaginary-time step as a fixed target to be fitted rather than a linear system to be solved, enabling large generic networks to be trained. It also identifies direct energy minimization as the limiting case in which this target is refreshed after every update. Second, the hard-core Hofstadter–Bose–Hubbard study ([P2]) turns NQS underperformance into a controlled representational diagnosis. The gap persists when exact

ground states are fitted directly and optimization and sampling are removed as causes, providing a stringent small-size benchmark for future architectures. Third, neural importance resampling (NIR, [P3]) makes the sampler a trainable, ansatz-independent component of the variational loop. In particular, it enables determinant multi-state NQS for which no exact sampler of the full target distribution is available. Finally, the toric-code study adds stabilizer-, loop-, and sampling-aware structure one ingredient at a time, determining which physical priors are needed to reach a topologically ordered low-energy manifold and where obstacles remain.

1.3 Statements of the thesis

1. SITE recasts imaginary-time evolution as repeated fitting to a frozen target state. It enables stable first-order optimization of large, generic NQS without solving the metric-tensor system required by SR, while retaining competitive accuracy. Direct energy minimization is recovered when the target is refreshed after every update, providing an explanation for its less stable training dynamics (Chapter 3).

2. Strong magnetic flux exposes a systematic representational mismatch between HBHM ground states and the generic MLP and CNN NQS. In that regime errors increase by several orders of magnitude. The persistence of this gap after removing VMC sampling and testing multiple optimization procedures identifies inefficient representation as the dominant limitation and makes the model a challenging controlled benchmark (Chapter 4).

3. NIR decouples the sampler from the variational ansatz through a trainable autoregressive proposal, importance resampling, and adaptation based on the effective sample size. It stabilizes VMC when local Markov chains become unreliable and enables determinant multi-state NQS for which no exact autoregressive sampler is available (Chapter 5).

4. The toric code demonstrates that exact solvability does not imply accessibility to a generic NQS workflow: raw-spin inputs combined with local sampling can fail even at zero field. Stabilizer and loop features, together with nonlocal sampling, recover the low-energy manifold on small clusters. However, training stability becomes the limiting factor at larger sizes, making this model a challenging benchmark (Chapter 6).

5. Variational energy alone does not reveal whether an inaccurate NQS result is limited by trainability, representation, or sampling. Controlled interventions across the four studies show that each can dominate in a different physical regime and must therefore be diagnosed separately when designing a reliable NQS calculation (Chapters 3–6).

1.4 Thesis structure

The remainder of this thesis is structured as follows.

Chapter 2 introduces the common background used throughout the thesis. It reviews the spin, bosonic, Hofstadter–Bose–Hubbard, and toric-code models; the relevant neural-network architectures; the NQS and VMC formalism, including optimization, sampling, symmetries, and excited states; and the software used in the numerical work.

Chapter 3 presents SITE. It derives the target construction, overlap loss, adaptive time step, and relation to direct energy minimization, then benchmarks the method on the J_1–J_2 Heisenberg model against direct optimization and DMRG.

Chapter 4 investigates the universal performance gap observed for NQS in the hard-core Hofstadter–Bose–Hubbard model. By comparing SR, SITE, exact sampling, and direct supervised learning of exact ground states, the chapter separates optimization effects from representational limitations. It then tests several possible explanations, including amplitude statistics, phase structure, entanglement, loss-landscape ruggedness, architecture choices, input encodings, curriculum learning, and numerical precision.

Chapter 5 introduces NIR as an alternative to MCMC sampling. It develops the importance-resampling construction, autoregressive proposal, KL training objectives, and adaptive effective-sample-size criterion, and tests them on TFIM and frustrated J_1–J_2 benchmarks with emphasis on multi-state NQS.

Chapter 6 studies the honeycomb toric code in a uniform field as a diagnostic case for NQS methods in topologically structured Hilbert spaces. It discusses the multi-state formulation needed for the low-energy manifold, the practical obstacles caused by loop constraints and degeneracy, and the targeted inductive biases and sampling improvements that make small-cluster calculations feasible.

Chapter 7 summarizes the main lessons. The overall message is that NQS are a powerful and flexible family of variational methods, but their reliability depends on matching the ansatz, optimizer, and sampler to the physical structure

of the problem. This is the perspective that connects the methodological chapters to the broader question of when neural variational methods can be used as reliable scientific tools.

2 Background

2.1 Many-body quantum systems

2.1.1 Many-body wave function and Hilbert space

Lattice many-body setting. In this work we consider interacting quantum many-body systems defined on discrete lattices. They provide a versatile framework for phenomena ranging from quantum magnetism and correlated particles to topological phases. Most examples use two-dimensional (2D) square lattices, although the toric-code study uses a honeycomb lattice and the formalism is not tied to either geometry.

The lattice models considered here are often effective rather than fully microscopic descriptions. A continuous material or field theory is replaced by discrete sites carrying the relevant low-energy degrees of freedom, such as localized spins or bosonic occupations. Conversely, for systems such as ultracold atoms in optical lattices, the lattice description can itself be a faithful microscopic model [7]. In either case, the resulting Hamiltonian retains the interactions and correlations of interest in a form amenable to numerical and, sometimes, analytical study.

Hilbert space and symmetry sectors. A lattice with N sites is described by a Hilbert space that factorizes into local state spaces,

$$\mathcal{H} = \bigotimes_{j=1}^{N} \mathcal{H}_j, \tag{2.1}$$

where $\mathcal{H}_j$ is the local state space, for example that of a spin-$\frac{1}{2}$ particle or a bosonic mode with restricted occupancy. For fermionic systems this tensor-product structure should be understood in the usual Fock-space sense, with the canonical anticommutation relations encoded in the operator algebra rather than in the bare tensor factors (see Sec. 2.1.3). The Hamiltonians considered here are local sums of operators acting on finite clusters of nearby sites. Often one

also works not in the full Hilbert space, but in a sector selected by conserved quantities or exact symmetries, such as fixed total magnetization (the total S^z of a spin system, Sec. 2.1.2) or fixed particle number. Besides reducing the number of basis states, quantum numbers such as total particle number, magnetization, crystal momentum, and point-group irrep identify the state or excitation being targeted. Later in the thesis, the same symmetries reappear as useful constraints and inductive biases for variational wave functions (see Sec. 2.3.10).

Product basis and wave-function amplitudes. To work explicitly, it is convenient to introduce a product basis built from local basis states,

$$|\boldsymbol{s}\rangle = \bigotimes_{j=1}^{N} |s_j\rangle, \tag{2.2}$$

where $|s_j\rangle$ is a basis vector of $\mathcal{H}_j$ on site j. Each many-body configuration can therefore be indexed by an N-tuple $\boldsymbol{s} = (s_1, \ldots, s_N)$, allowing the shorthand notation $|\boldsymbol{s}\rangle = |s_1, \ldots, s_N\rangle$. A familiar example is the spin-$\frac{1}{2}$ case, where one may choose $|s_j\rangle \in \{|\uparrow\rangle, |\downarrow\rangle\}$, the eigenstates of the local operator $\hat{S}_j^z$.

In this basis, an arbitrary quantum state can be expressed as

$$|\psi\rangle = \sum_{s_1,\ldots,s_N} \psi(s_1, \ldots, s_N)\, |s_1, \ldots, s_N\rangle, \tag{2.3}$$

with complex amplitudes $\psi(\boldsymbol{s}) \equiv \psi(s_1, \ldots, s_N)$. If all sites have local dimension d, the number of basis states is $\dim \mathcal{H} = d^N$. Even storing the full vector of amplitudes $\psi(\boldsymbol{s})$ therefore becomes infeasible beyond modest N, limiting exact diagonalization to small systems. This motivates the development of approximate methods that avoid working directly with the full Hilbert space.

Numerical difficulty and physical structure. Different numerical methods exploit different structures of physically relevant states. Quantum Monte Carlo can often reach much larger system sizes when a basis exists in which the statistical weights remain nonnegative, but in frustrated systems or in the presence of magnetic flux destructive sign or phase cancellations can make sampling exponentially difficult [11]. Tensor-network methods such as matrix-product states and DMRG exploit locality and limited entanglement very effectively, especially in one dimension and on quasi-one-dimensional geometries, but in genuinely two-dimensional settings their computational cost can still grow

rapidly with the entanglement that must be retained [13]. Variational wave functions, including NQS, complement these methods in regimes where their assumptions become restrictive.

Whether a problem is numerically benign in this sense is basis dependent. In computational many-body language, a Hamiltonian whose off-diagonal matrix elements can be made real and non-positive in a chosen product basis is called *stoquastic* [36]. For connected stoquastic problems the ground-state amplitudes in that basis can be chosen nonnegative, so the corresponding Monte Carlo weights can often be interpreted as ordinary probabilities without destructive sign cancellations. By contrast, frustration or Peierls phases generally spoil this simple structure and lead to sign- or phase-structured wave functions. Because several benchmark models later in the thesis become difficult precisely for this reason, this distinction will recur throughout the later discussion.

A key reason variational methods can work is that low-energy states of local Hamiltonians are not generic vectors in Hilbert space. Ground states of gapped one-dimensional systems obey an entanglement area law, meaning that the entanglement entropy scales with the boundary of a subsystem rather than its volume; analogous scaling, possibly with subleading corrections, is well supported for gapped phases in higher dimensions [15, 37]. Critical states may have scale-free correlations and enhanced entanglement [38], but they remain far more structured than typical random states, which exhibit volume-law entanglement [15]. Tensor networks [13] and NQS [31] exploit this locality-, symmetry-, and entanglement-induced structure to parameterize low-energy wave functions without enumerating all d^N amplitudes.

2.1.2 Spin lattice models

Local spin degrees of freedom. *Spin lattice models* are one of the main classes of lattice many-body systems. Each site carries a finite-dimensional spin degree of freedom, most commonly spin-$\frac{1}{2}$ with $\mathcal{H}_j \cong \mathbb{C}^2$ and observables represented by Pauli matrices or spin operators. In materials these variables often describe effective low-energy doublets rather than bare electronic spins [39]; the resulting models retain the dominant couplings and symmetries while discarding microscopic high-energy detail.

Hamiltonian structure. The Hamiltonian of a spin lattice model is a Hermitian operator acting on the full Hilbert space $\bigotimes_j \mathcal{H}_j$. In the cases relevant here it is

a sum of on-site terms, such as external fields, and interactions on nearby sites:

$$\hat{H} = \sum_j \hat{h}_j^{(\mathrm{loc})} + \sum_{\langle j,k \rangle} \hat{h}_{jk}^{(\mathrm{int})}, \tag{2.4}$$

where $\langle j, k \rangle$ denotes neighbouring pairs of sites.

Depending on the couplings and geometry, such Hamiltonians exhibit magnetic order, quantum phase transitions, collective excitations, and frustrated phases in which competing interactions cannot be minimized simultaneously.

Role as benchmarks. Their combination of simple definitions and nontrivial collective behaviour also makes spin models standard numerical benchmarks. Here the transverse-field Ising model provides a clean sampling test, the Heisenberg model supplies the $SU(2)$-symmetric reference for its frustrated J_1–J_2 extension, and the toric code tests topological order and constrained wave functions. The remainder of this subsection briefly introduces these models, which reappear later as case studies in the thesis.

Transverse-field Ising model

Definition and competition. The *transverse-field Ising model* (TFIM) is one of the simplest quantum spin models that already displays a genuine competition between ordering and quantum fluctuations. In the ferromagnetic form considered here, it consists of spin-$\frac{1}{2}$ degrees of freedom with a nearest-neighbour Ising coupling that favours alignment along the z-axis, together with a transverse magnetic field along x that flips spins in the σ^z basis. Because the interaction term and the transverse-field term do not commute, the model exhibits nontrivial ground states and a zero-temperature quantum phase transition.

On a lattice with sites labeled by j, the Hamiltonian takes the form

$$\hat{H} = -\sum_{\langle j,k \rangle} \hat{\sigma}_j^z \hat{\sigma}_k^z - g \sum_j \hat{\sigma}_j^x , \tag{2.5}$$

where $\langle j, k \rangle$ denotes nearest-neighbour pairs, $\hat{\sigma}_j^z, \hat{\sigma}_j^x$ are the Pauli operators acting on the spin at site j, and g controls the strength of the transverse field relative to the Ising coupling, which we have set to unity. The minus sign in the first term corresponds to a ferromagnetic interaction. The first term favours spin alignment along z, while the transverse field generates local spin flips, competing with this order and driving the system toward a quantum paramagnet.

Limiting regimes and symmetry. Some key qualitative aspects of TFIM:

- When $g = 0$, the model reduces to the classical Ising model: the Hamiltonian is diagonal in the σ^z basis and the spectrum coincides with that of the classical Ising energy function on the same lattice. For the ferromagnetic sign chosen in Eq. (2.5), the ground-state manifold is twofold degenerate, corresponding to the fully ordered configurations with all spins aligned up or all spins aligned down along z.
- When $g \to \infty$, the ground state is non-degenerate and approaches a simple product state in which every spin is polarized along the x-direction (the transverse-field direction). In this limit the state is unentangled and all connected correlations between distinct sites vanish.
- For intermediate g, the model interpolates between a ferromagnetically ordered phase and a quantum paramagnet. On the infinite square lattice, these two phases are separated by a continuous quantum phase transition at $g_c \approx 3.0444$ [40].

Figure 2.1 collects the two terms and the three regimes in a single picture.

The TFIM possesses a global $\mathbb{Z}_2$ spin-flip symmetry generated by

$$\hat{P} = \prod_j \hat{\sigma}_j^x, \tag{2.6}$$

which flips the sign of every z-component, $\hat{P}\,\hat{\sigma}_j^z\,\hat{P}^{-1} = -\hat{\sigma}_j^z$. In the ferromagnetic phase at small g, this symmetry is spontaneously broken in the thermodynamic limit, and the longitudinal magnetization, i.e. the expectation value of the spins themselves rather than the applied field,

$$m_z = \langle \hat{\sigma}_j^z \rangle \tag{2.7}$$

serves as an order parameter with $m_z \neq 0$. In the paramagnetic phase at large g the ground state is unique, the $\mathbb{Z}_2$ symmetry is preserved, and $m_z = 0$.

Critical behaviour and benchmark role. In the path-integral formulation, imaginary time acts as an additional dimension, so the zero-temperature quantum problem can be recast as a classical statistical model in one higher dimension. The 2D TFIM transition consequently belongs to the universality class of the three-dimensional classical Ising model [38]. Unlike its one-dimensional counterpart, which maps to free fermions, the square-lattice TFIM has no exact solution, yet its limiting regimes and critical point are well characterized. It

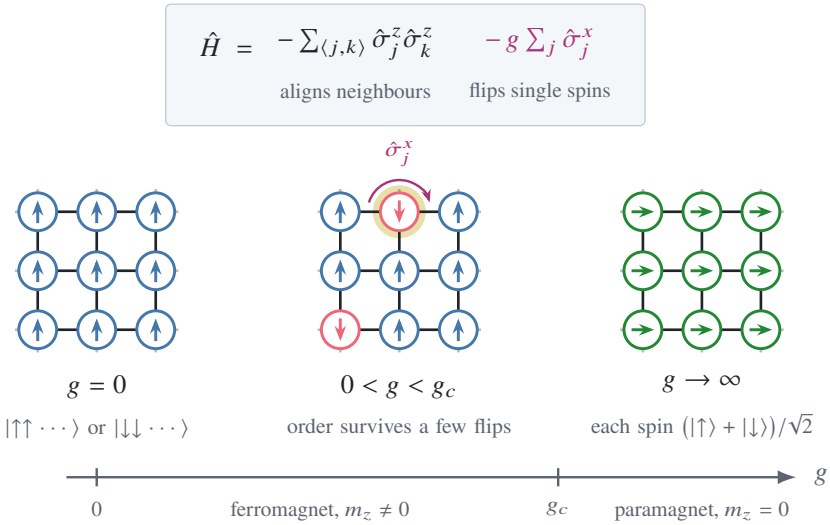

Figure 2.1: Competition between the two terms of the transverse-field Ising Hamiltonian, Eq. (2.5). The Ising term acts on the bonds (solid lines) and aligns neighbouring spins along z, while the transverse field acts on single sites and flips them in the σ^z basis, as marked on the shaded site by the purple arrow. Blue and red arrows denote the local states $|\uparrow\rangle$ and $|\downarrow\rangle$; faded bonds indicate that each panel is a window onto a larger lattice. At $g = 0$ the ground state is one of the two fully aligned configurations. At intermediate g it is a superposition dominated by configurations such as the one drawn, in which the field has flipped a few spins but the ferromagnetic order survives. As $g \to \infty$ every spin is polarized along the field (green), that is, in the superposition $(|\uparrow\rangle + |\downarrow\rangle)/\sqrt{2}$, and the order is lost. The two regimes are separated by the quantum critical point g_c.

therefore provides a nontrivial benchmark without the additional complications of frustration.

Heisenberg model

Definition and symmetry. The *Heisenberg model* is one of the foundational quantum spin lattice models, describing exchange-coupled localized spins on a lattice. In its simplest and most widely studied form, each lattice site j carries a spin-S degree of freedom (often $S = \frac{1}{2}$), and the Hamiltonian couples nearest-neighbour spins through an isotropic exchange interaction that is invariant under global $SU(2)$ spin rotations. Here $SU(2)$ invariance means that the Hamiltonian is unchanged under a global rotation of all spins, or equivalently that no spin direction is preferred.

A standard nearest-neighbour isotropic quantum Heisenberg Hamiltonian on a lattice reads

$$\hat{H} = J \sum_{\langle j,k \rangle} \hat{\boldsymbol{S}}_j \cdot \hat{\boldsymbol{S}}_k, \tag{2.8}$$

where J controls the strength of the exchange coupling, $\hat{\boldsymbol{S}}_j = (\hat{S}_j^x, \hat{S}_j^y, \hat{S}_j^z)$ denotes the spin operator on site j, and the summation $\langle j, k \rangle$ runs over nearest neighbours. For spin-$\frac{1}{2}$ one has $\hat{\boldsymbol{S}}_j = \frac{1}{2}(\hat{\sigma}_j^x, \hat{\sigma}_j^y, \hat{\sigma}_j^z)$ in terms of the Pauli operators. The spin operators satisfy the $SU(2)$ commutation relations $[\hat{S}_j^\alpha, \hat{S}_k^\beta] = \mathrm{i}\delta_{jk}\, \varepsilon_{\alpha\beta\gamma} \hat{S}_j^\gamma$. A direct consequence of the $SU(2)$ invariance is that $\hat{H}$ commutes with the total spin, in particular with its z-component $\hat{S}_{\text{tot}}^z = \sum_j \hat{S}_j^z$, so that the Hamiltonian is block diagonal in sectors of fixed *total magnetization*, the eigenvalue of $\hat{S}_{\text{tot}}^z$. In the σ^z basis used throughout this thesis, a configuration $\boldsymbol{s}$ with $s_j = \pm 1$ has magnetization $\frac{1}{2}\sum_j s_j$, half the excess of up spins over down spins, and a fixed-magnetization sector is simply the set of configurations with a fixed number of up spins. Throughout the thesis, "magnetization" refers to this property of the spin state, the conserved quantum number that labels the sector of the J_1–J_2 calculations in Chapters 3 and 5, and not to an applied magnetic field. Magnetic fields proper appear only as explicit Hamiltonian terms: the transverse field of the TFIM and the perturbing field of the toric code (Sec. 2.1.2) act on the spins themselves, whereas the field of the HBHM (Sec. 2.1.3) acts on the orbital motion of spinless bosons through the hopping phases.

Exchange sign and variants. The sign of the coupling constant J determines the preferred ordering:

- $J < 0$ (ferromagnetic): parallel spin alignment is energetically favoured;
- $J > 0$ (antiferromagnetic): antiparallel spin correlations are energetically favoured.

Anisotropic variants replace the dot product by direction-dependent couplings, giving the XYZ, XXZ, and XY models. The isotropic model often arises as an effective description of localized moments coupled by direct exchange or superexchange and serves as a canonical prototype of quantum magnetism.

Ground-state behaviour. On unfrustrated bipartite lattices with $J > 0$, the classical tendency is Néel order; quantum fluctuations reduce the ordered moment but do not destroy zero-temperature long-range order on the square lattice [10]. Competing interactions or non-bipartite geometries can instead enhance quantum fluctuations and stabilize nonmagnetic phases, including candidate quantum spin liquids [3].

For the purposes of this thesis, the Heisenberg model mainly serves as the canonical unfrustrated $SU(2)$-symmetric reference point, providing the notation and physical baseline from which the frustrated J_1–J_2 extension considered below is obtained.

J_1–J_2 Heisenberg model

Frustrated model. Adding next-nearest-neighbour exchange to the nearest-neighbour Heisenberg Hamiltonian gives the J_1–J_2 *Heisenberg model*. On the square lattice, and in the commonly studied antiferromagnetic regime $J_1 > 0$, $J_2 > 0$, the two couplings compete with one another. The J_2 interaction couples spins on the same Néel sublattice antiferromagnetically and therefore frustrates the two-sublattice Néel order favoured by J_1. This makes the model a paradigmatic setting for studying magnetic frustration and competing phases. The lattice geometry and the origin of the frustration are shown in Fig. 2.2.

On a 2D square lattice the Hamiltonian of the J_1–J_2 Heisenberg model reads

$$\hat{H} = J_1 \sum_{\langle j,k \rangle} \hat{\boldsymbol{S}}_j \cdot \hat{\boldsymbol{S}}_k + J_2 \sum_{\langle\langle j,k \rangle\rangle} \hat{\boldsymbol{S}}_j \cdot \hat{\boldsymbol{S}}_k \,, \tag{2.9}$$

where $\langle j, k \rangle$ and $\langle\langle j, k \rangle\rangle$ denote nearest-neighbour and next-nearest-neighbour (diagonal) pairs, respectively. In the special case $J_2 = 0$, the model reduces to the standard nearest-neighbour Heisenberg Hamiltonian. In the classical large-S limit, Néel order is favoured for $J_2/J_1 < 1/2$, whereas for $J_2/J_1 > 1/2$

two interpenetrating Néel sublattices have a classically free relative orientation; fluctuations select striped states with ordering wave-vectors $(\pi, 0)$ and $(0, \pi)$ [2, 41]. The ordering wave-vector labels the spatial periodicity of the spin pattern in reciprocal space. The vicinity of $J_2/J_1 = 1/2$ is therefore maximally frustrated and provides the main benchmark regime used in this thesis.

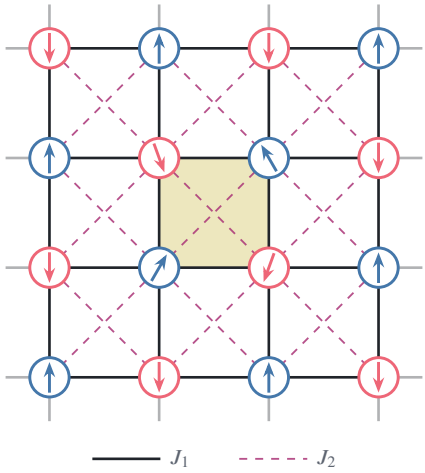

Figure 2.2: The J_1–J_2 Heisenberg model on the square lattice. A spin-$\frac{1}{2}$ sits on every site; solid bonds carry the nearest-neighbour coupling J_1 and dashed diagonals the next-nearest-neighbour coupling J_2. Faded bonds at the boundary indicate that the drawing is a window onto a larger lattice. Arrow directions and colours show the two-sublattice Néel pattern favoured by J_1 alone (blue for up, red for down). On the shaded plaquette the two couplings make incompatible demands: J_1 wants the four spins to alternate, while antiferromagnetic J_2 wants each diagonal pair to anti-align as well, which the Néel pattern violates. The spins there are drawn canted as a reminder that no classical arrangement satisfies all bonds at once.

Physical interpretation and phases. In magnetic materials, exchange can extend beyond immediate neighbours through, for example, longer superexchange paths. When these couplings compete, not all bonds can be minimized simultaneously; the resulting frustration suppresses conventional order, enhances quantum fluctuations, and may stabilize nonmagnetic phases [3].

Numerical studies agree on Néel order at small J_2/J_1 and stripe order at sufficiently large J_2/J_1, but the intervening nonmagnetic regime remains method-dependent [33, 42–46]. Néel order produces correlations at wave-vector

(π, π), whereas stripe order favours $(\pi, 0)$ or $(0, \pi)$ and breaks the lattice C_4 rotation symmetry. Between them, proposed phases include a valence-bond solid, in which spins form singlets in an ordered spatial pattern, and quantum-spin-liquid behaviour; subtle long-distance diagnostics and strong finite-size effects make this intermediate regime difficult to resolve numerically [2, 3, 46]. Recent DMRG evidence favours a valence-bond-solid regime bounded by a continuous Néel-to-valence-bond-solid transition near $J_2/J_1 \approx 0.54$ and a first-order transition to stripe order near $J_2/J_1 \approx 0.61$, with no extended spin liquid; strong finite-size effects nevertheless make thermodynamic extrapolation delicate [46].

Sign structure. In theoretical many-body physics, the J_1–J_2 model plays a dual role: it remains simple enough to be studied with a range of analytic and numerical tools, yet complex enough to capture essential effects of frustration and competing order. For the unfrustrated nearest-neighbour antiferromagnet on a bipartite lattice, a sublattice-dependent phase convention, effectively attaching an alternating sign to one of the two sublattices, allows the ground-state amplitudes in the S^z basis to be chosen nonnegative, a statement known as the Marshall sign rule [47]. Once antiferromagnetic J_2 is added, this simple sign structure is generally lost. Sign-problem-free quantum Monte Carlo formulations based on the bipartite Marshall structure are then no longer available in general, and standard simulations suffer from severe cancellations between positive and negative contributions [11]. This is one of the main reasons the model is a demanding benchmark for variational approaches.

For this thesis, the J_1–J_2 model is especially valuable precisely because of this combination of simplicity and difficulty. At the Hamiltonian level it is still minimal, involving only bilinear spin–spin couplings, but physically it contains frustration, competing orders, and a complicated sign structure. This makes it a natural stress test for NQS-based optimization and sampling methods in later chapters.

Toric code model

Edge qubits and stabilizers. The *toric code model* is an exactly solvable spin-$\frac{1}{2}$ lattice model introduced by Kitaev as a paradigmatic setting for topological quantum memory and anyonic excitations [48, 49]. Unlike the site-spin models above, the spin-$\frac{1}{2}$ degrees of freedom live on the edges of a two-dimensional square lattice. Following the convention used by Simon [49], we work in the

σ^z basis on each edge and denote vertices by v and plaquettes by p. The two elementary stabilizers are

$$\hat{A}_v = \prod_{j \in v} \hat{\sigma}_j^z, \qquad \hat{B}_p = \prod_{j \in p} \hat{\sigma}_j^x, \tag{2.10}$$

where each product contains the four edges incident on the vertex or surrounding the plaquette. In this convention, $\hat{A}_v$ measures the parity of down spins incident on vertex v, while $\hat{B}_p$ flips the four spins around plaquette p. The geometry of the two stabilizers is shown in Fig. 2.3, together with the two-edge overlap that makes them commute. Kitaev's original convention interchanges the roles of σ^x and σ^z in these two operators; the two descriptions are related by a global basis rotation or by a lattice-duality transformation.

Commuting-projector Hamiltonian. The Hamiltonian is

$$\hat{H}_{\mathrm{TC}} = -\frac{\Delta_v}{2} \sum_v \hat{A}_v - \frac{\Delta_p}{2} \sum_p \hat{B}_p, \tag{2.11}$$

with $\Delta_v, \Delta_p > 0$. Each stabilizer squares to the identity and therefore has eigenvalues ± 1. With the factor of $1/2$ in Eq. (2.11), changing a single stabilizer eigenvalue from $+1$ to -1 raises the energy by Δ_v or Δ_p, so these parameters are directly the energy costs of individual vertex and plaquette defects. All stabilizers commute with one another: vertex operators commute among themselves, plaquette operators commute among themselves, and a vertex and plaquette operator either share no edge or share two edges. On a shared edge σ^z and σ^x anticommute, while operators on different edges commute, so in the two-edge case the two minus signs cancel. The model is therefore frustration-free, meaning that every term can be minimized simultaneously, and exactly solvable. Its ground-state subspace is obtained by imposing

$$\hat{A}_v = +1, \qquad \hat{B}_p = +1 \tag{2.12}$$

for every vertex and plaquette.

Loop structure and topology. In the σ^z basis, the constraint $\hat{A}_v = +1$ requires an even number of down spins to meet at each vertex. Equivalently, configurations with nonzero amplitude in the ground-state manifold can be represented as closed loops of down-spin edges, with no open endpoints. The

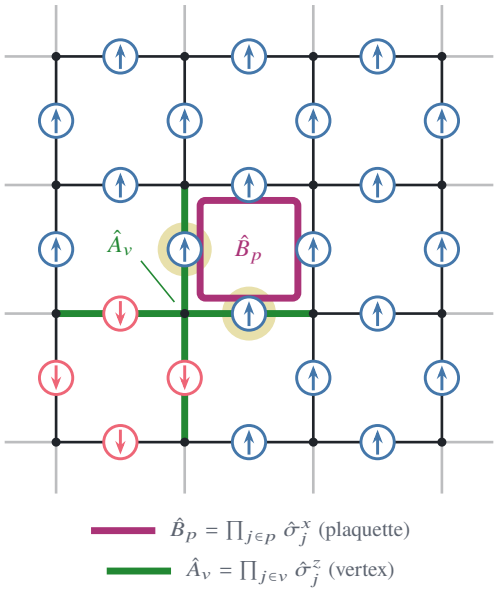

Figure 2.3: The two elementary stabilizers of the square-lattice toric code. Unlike the site-spin models discussed above, a spin-$\frac{1}{2}$ lives on every *edge* (circles, blue for up and red for down); the small dots mark the lattice vertices, which carry no degree of freedom, and faded bonds at the boundary indicate the periodic lattice. The vertex operator $\hat{A}_v$ (green cross) multiplies $\hat{\sigma}^z$ over the four edges meeting at a vertex and therefore measures the parity of down spins there: two of those four are down in the configuration drawn, so $\hat{A}_v = +1$. The plaquette operator $\hat{B}_p$ (purple ring) multiplies $\hat{\sigma}^x$ over the four edges surrounding a plaquette and flips all four. The two operators drawn here overlap on exactly two edges (shaded), and on each of them $\hat{\sigma}^z$ and $\hat{\sigma}^x$ anticommute, so the two minus signs cancel and all stabilizers commute. The down spins shown close into a loop, which is the structure that $\hat{A}_v = +1$ at every vertex imposes.

plaquette constraint $\hat{B}_p = +1$ then makes the ground state an equal-amplitude superposition over loop configurations related by local plaquette flips. For example, in the trivial winding sector one may write a representative ground state as

$$|\psi_0\rangle \propto \prod_p \frac{1+\hat{B}_p}{2} |\uparrow \cdots \uparrow\rangle , \tag{2.13}$$

where $|\uparrow \cdots \uparrow\rangle$ is the $\hat{A}_v = +1$ product state with all edge spins up. On a torus, plaquette flips cannot change the parity of loops winding around either noncontractible cycle. The two winding parities therefore label four ground-state sectors on the torus, or more generally a degeneracy 2^{2g} on a closed orientable surface of genus g. This topology-dependent degeneracy, together with anyonic excitations and the absence of a local order parameter, is a defining manifestation of $\mathbb{Z}_2$ topological order.

Quantum error-correction interpretation. The same structure makes the toric code the canonical example of a topological quantum error-correcting code [6, 50]. The four ground states on a torus encode two logical qubits, noncontractible spin strings act as logical operators, and stabilizer violations reveal the endpoints of local error strings. Its planar counterpart, the surface code, is consequently a central architecture for fault-tolerant quantum computing.

Excitations and benchmark role. Excitations are stabilizer violations. In Simon's convention, an edge operator $\hat{\sigma}_j^x$ anticommutes with the two adjacent vertex operators and therefore creates or moves a pair of vertex defects with $\hat{A}_v = -1$. Similarly, $\hat{\sigma}_j^z$ creates or moves a pair of plaquette defects with $\hat{B}_p = -1$ on the two plaquettes adjacent to the edge. These vertex and plaquette defects are the electric (e) and magnetic (m) anyons, whose mutual braiding produces a phase -1. For variational methods, the toric code is useful because its Hamiltonian is local and exactly solvable while its ground states contain nonlocal loop constraints and long-range entanglement. It therefore tests whether an ansatz can represent topological structure that is not captured by a conventional local order parameter.

2.1.3 Tight-binding lattice models

Scope and notation. Unlike spin models, tight-binding lattice models describe mobile particles hopping between localized states. Here the term is used broadly

for bosonic or fermionic Hamiltonians written in second quantization in a localized single-particle basis, such as atomic-like or Wannier orbitals for electrons and Wannier states of an optical lattice for cold atoms [7].

Second quantization. Second quantization describes indistinguishable particles by occupation-number, or Fock, states rather than explicitly symmetrized or antisymmetrized wave functions [51]. Creation and annihilation operators add or remove particles in the chosen basis, while their commutation or anticommutation relations encode bosonic or fermionic statistics. Hopping, interactions, and particle-number constraints can then be written compactly.

Single-particle tight binding. In the narrower solid-state sense, the tight-binding approximation uses orbitals $|\phi_{j,n}\rangle$ localized near site j, where n labels the orbital, and represents their overlap by hopping matrix elements. A noninteracting Hamiltonian has the form

$$\hat{H}_0 = \sum_{j,n} \epsilon_{j,n} \hat{c}^\dagger_{j,n} \hat{c}_{j,n} + \sum_{(j,n)\neq(k,m)} t_{j,n;k,m} \hat{c}^\dagger_{j,n} \hat{c}_{k,m} \,, \tag{2.14}$$

where $\epsilon_{j,n}$ are on-site energies, $t_{j,n;k,m} = t^*_{k,m;j,n}$ are hopping amplitudes, and spin indices are suppressed. For periodic lattices, Bloch's theorem yields energy bands whose dispersion is set by the lattice geometry and hopping range [52]. Adding interactions yields many-body models such as the Hubbard and Bose–Hubbard families [53]; the bosonic case needed here is introduced next.

Bose–Hubbard model

Definition. The *Bose–Hubbard model* (BHM) [54] is the standard minimal lattice model for interacting bosons. In its simplest single-band form it describes bosons that hop between neighbouring sites and interact through an on-site interaction, making it a paradigmatic example of a strongly correlated bosonic lattice system.

On a lattice with sites labeled by j, the BHM Hamiltonian reads

$$\hat{H} = -J \sum_{\langle j,k \rangle} \left(\hat{b}^\dagger_j \hat{b}_k + \hat{b}^\dagger_k \hat{b}_j \right) + \frac{U}{2} \sum_j \hat{n}_j (\hat{n}_j - 1) \,, \tag{2.15}$$

where $\hat{b}^\dagger_j$ and $\hat{b}_j$ are bosonic creation and annihilation operators, $\hat{n}_j = \hat{b}^\dagger_j \hat{b}_j$ is the number operator, J is the tunnelling (hopping) amplitude, not to be

confused with the exchange couplings of the spin models above, and U is the on-site interaction strength. The operators $\hat{b}_j^\dagger$, $\hat{b}_j$ obey the canonical bosonic commutation relations $[\hat{b}_i, \hat{b}_j^\dagger] = \delta_{ij}$, $[\hat{b}_i, \hat{b}_j] = [\hat{b}_i^\dagger, \hat{b}_j^\dagger] = 0$, and the total particle number $\hat{N}_\mathrm{b} = \sum_j \hat{n}_j$ is conserved. In a grand-canonical description one also includes a chemical-potential term $-\mu \sum_j \hat{n}_j$ to control the average filling. The minus sign in the kinetic term ensures that delocalized, phase-coherent states lower the energy when $J > 0$.

Superfluid and Mott regimes. Physically, the first term describes bosons tunnelling between neighbouring lattice sites, while the second penalizes (for $U > 0$) multiple occupation of a single site. At integer filling, the competition between kinetic delocalization and interaction-induced localization gives two characteristic zero-temperature regimes [55]:

- **Superfluid regime:** when hopping dominates ($J/U \gg 1$ at fixed density), bosons delocalize over the lattice. The phase is compressible and gapless, with long-range phase coherence in $D \geq 2$ and quasi-long-range order in 1D.
- **Mott-insulating regime:** when interactions dominate ($U/J \gg 1$) at integer filling, it becomes energetically costly to change the occupation of a given site by ± 1. In the atomic limit $J = 0$, the ground state is a product of local Fock states; the phase is incompressible and has a finite gap to particle and hole excitations.

Tuning J/U and the chemical potential μ drives transitions between these phases and produces Mott lobes in the $(\mu/U, J/U)$ phase diagram [55]. The two regimes and the mechanism that separates them are contrasted in Fig. 2.4. An especially important limiting case is the hard-core regime $U/J \to \infty$, where double occupation is forbidden and the local occupation is restricted to $n_j \in \{0, 1\}$. Each site then carries a two-dimensional local Hilbert space, making the model closely analogous to a spin-$\frac{1}{2}$ system while retaining the language of bosonic number conservation [56]. This hard-core limit will be relevant later when discussing the Hofstadter–Bose–Hubbard benchmark.

Experimental relevance and extensions. The BHM is realized by ultracold bosonic atoms in optical lattices, where the lattice depth and confinement control J and U [7]. It also provides the starting point for the gauge-field extension used in this thesis.

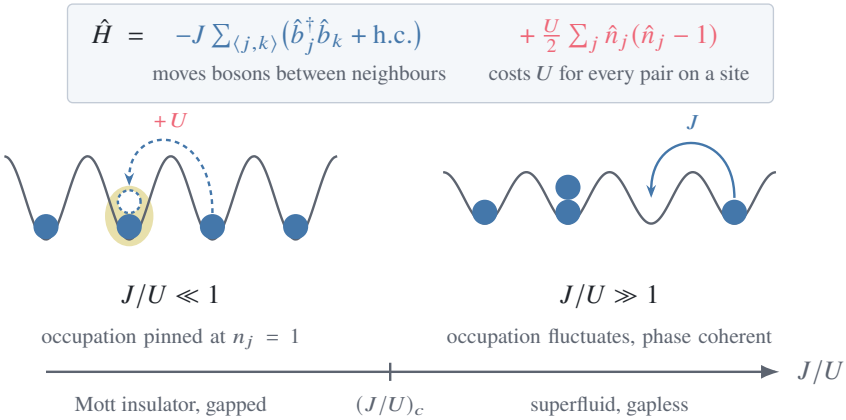

Figure 2.4: Competition in the Bose–Hubbard model for four bosons on four sites. Left: for $J/U \ll 1$, a hop would create double occupation at energy cost U, producing the incompressible Mott regime. Right: for $J/U \gg 1$, hopping delocalizes the particles and site occupations fluctuate in the coherent superfluid regime. The hard-core limit used later lies at the far left, where double occupation is excluded.

Hofstadter–Bose–Hubbard model

Peierls phases and magnetic flux. To incorporate the effect of a (real or synthetic) magnetic field into a tight-binding lattice model, one typically uses the Peierls substitution [57]. Here the field acts on the orbital motion of spinless bosons through complex hopping phases, whose strength is set by the flux per plaquette α. In continuum quantum mechanics, a particle of charge Q coupled to a magnetic vector potential $\boldsymbol{A}$ acquires the minimal coupling $\boldsymbol{p} \to \boldsymbol{p} - Q\boldsymbol{A}$. On a lattice, this is represented by attaching a complex phase to each hopping matrix element:

$$e^{i\phi(j,k)} = \exp\left(i\frac{Q}{\hbar}\int_{\boldsymbol{r}_k}^{\boldsymbol{r}_j} \boldsymbol{A}\cdot d\boldsymbol{r}\right). \tag{2.16}$$

The integral follows the hopping path from k to j. Gauge transformations shift the individual Peierls phases through local basis redefinitions $\hat{b}_j \to e^{i\theta_j}\hat{b}_j$, but leave their oriented sum around a plaquette unchanged. Because the net phase around a plaquette is gauge invariant, a nonzero flux cannot be removed from all bonds simultaneously by a local phase redefinition. In a fixed occupation-number basis, many-body eigenstates at finite flux therefore generically have a nontrivial complex phase structure rather than a purely real, nonnegative amplitude representation. This basis-dependent phase structure is one reason

magnetic-field lattice models are often numerically more demanding than their zero-flux counterparts [11]. Without interactions this construction gives the Hofstadter model, whose magnetic subbands form the Hofstadter butterfly and can carry nonzero Chern numbers [58]. At partial filling, interactions can additionally produce fractional Chern insulators, which possess many-body topological order rather than only single-particle band topology [59–61]. Synthetic gauge fields make this physics accessible in ultracold-atom experiments [62].

Interacting Hofstadter bosons. Building on the ordinary bosonic lattice Hamiltonian, the Hofstadter–Bose–Hubbard model supplements the Hofstadter kinetic term with on-site bosonic interactions. In the version considered here, the Hamiltonian takes the form

$$\hat{H} = -J \sum_{\langle j,k \rangle} \left(\mathrm{e}^{\mathrm{i}\phi(j,k)} \hat{b}_j^\dagger \hat{b}_k + \text{h.c.} \right) + \frac{U}{2} \sum_j \hat{n}_j (\hat{n}_j - 1) \,, \tag{2.17}$$

where $\phi(j,k)$ is the Peierls phase associated with hopping from k to j, chosen such that the oriented sum of phases around each plaquette equals the prescribed magnetic flux, and all other notation is the same as in the previous section on the BHM. A particularly important regime for this thesis is the hard-core limit $U/J \to \infty$, in which the occupation numbers are restricted to $n_j \in \{0, 1\}$ on every site and the on-site Hilbert space again becomes two-dimensional.

Landau gauge convention. For explicit calculations in this work we choose a specific gauge that implements a uniform magnetic field. On a square $L_x \times L_y$ lattice a standard choice is the Landau gauge,

$$\boldsymbol{A} = (0,\, Bx,\, 0), \tag{2.18}$$

which yields Peierls phases that depend linearly on the x-coordinate. Labelling the sites by integer coordinates (j,k), with $x = ja$ and $y = ka$, hopping in the x-direction then carries no phase, while hopping in the y-direction from column j produces the phase $2\pi\alpha j$, where the dimensionless flux per plaquette is

$$\alpha = \frac{QBa^2}{h} = \frac{Ba^2}{\Phi_0} \tag{2.19}$$

for lattice spacing a and flux quantum $\Phi_0 = h/Q$. Concretely, Eq. (2.17) becomes

$$\hat{H} = -J \sum_{j,k} \left[\hat{b}^\dagger_{j+1,k} \hat{b}_{j,k} + \mathrm{e}^{\mathrm{i}2\pi\alpha j}\, \hat{b}^\dagger_{j,k+1} \hat{b}_{j,k} + \mathrm{h.c.} \right] + \frac{U}{2} \sum_{j,k} \hat{n}_{j,k} \left(\hat{n}_{j,k} - 1 \right), \tag{2.20}$$

with appropriate boundary conditions (e.g. periodic) understood; here $\hat{b}^\dagger_{j,k}$ creates a boson on the site with coordinates (j, k), and the pair (j, k) no longer labels a bond as in Eq. (2.17). Moving counter-clockwise around any elementary plaquette, a particle accumulates a total phase $2\pi\alpha$, so the flux is uniform. Because α enters periodically (the spectrum is invariant under $\alpha \to \alpha + 1$) and is symmetric under $\alpha \to -\alpha$, one can restrict attention to $\alpha \in [0, 0.5]$. This gauge-fixed model is illustrated in Fig. 2.5.

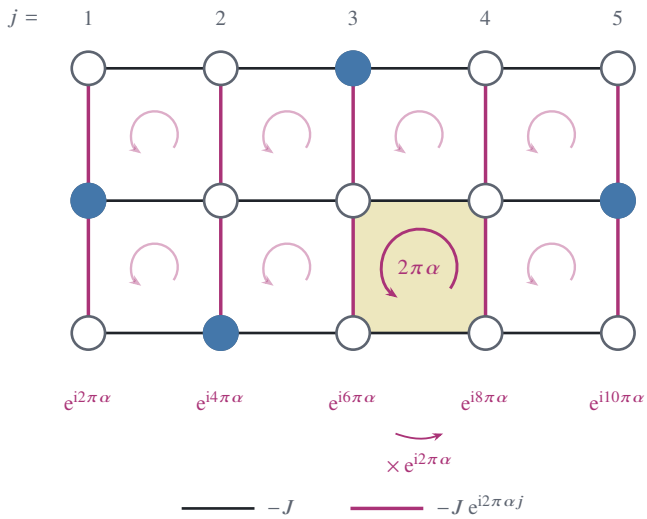

Figure 2.5: The Hofstadter–Bose–Hubbard model in the Landau gauge of Eq. (2.20). Horizontal hopping carries no phase, while an upward hop in column j carries $\mathrm{e}^{\mathrm{i}2\pi\alpha j}$. The individual phases are gauge dependent, but their oriented product around every plaquette is $\mathrm{e}^{\mathrm{i}2\pi\alpha}$ and cannot be gauged away. Blue discs mark occupied sites; in the hard-core limit used here, each site holds at most one boson.

The gauge-fixed HBHM combines nontrivial hopping phases with strong interactions and conserved particle number. For this thesis, its flux-dependent complex wave-function structure makes it a demanding benchmark for variational representations.

2.2 Neural networks and deep learning

This section summarizes the neural-network background relevant to the thesis. Generic MLPs provide the main implicit NQS in the SITE chapter and a baseline for the HBHM study, CNNs supply a structured lattice architecture, and transformers parameterize the autoregressive sampler in NIR. The emphasis is on properties that affect the representation and VMC optimization of lattice wave functions.

2.2.1 Multi-layer perceptrons

The *multi-layer perceptron* (MLP), also called a *fully connected feed-forward network*, is one of the simplest neural-network architectures [63]. An MLP represents a parametric function

$$f_{\boldsymbol{\theta}} : \mathbb{R}^{d_0} \to \mathbb{R}^{d_L}, \tag{2.21}$$

built by stacking affine maps, each followed by a fixed componentwise nonlinearity. In the context of this thesis, MLPs provide a useful baseline architecture: with suitable nonlinearities they are universal approximators, yet they impose relatively little built-in preference for locality, symmetry, or other problem-specific structure. This makes them a useful reference for separating gains due to more structured architectures (e.g. convolutional or attention-based models) from gains due purely to optimization.

Architecture and forward pass. Let $\boldsymbol{h}^{(0)} = \boldsymbol{x} \in \mathbb{R}^{d_0}$ denote the input vector (e.g. a flattened lattice configuration, possibly after preprocessing such as one-hot encoding). For example, a spin-$\frac{1}{2}$ configuration on a 2×2 lattice can be listed row by row as $\boldsymbol{x} = (s_1, s_2, s_3, s_4)$ with $s_i = \pm 1$, discarding the explicit two-dimensional geometry. For larger local bases, each site may instead be one-hot encoded, for example $n_i = 2 \mapsto (0, 0, 1, 0)$ for $n_i \in \{0, 1, 2, 3\}$, before the site vectors are concatenated. An L-layer MLP recursively computes hidden activations

$$\boldsymbol{h}^{(\ell)} = \phi^{(\ell)}\Big(W^{(\ell)}\boldsymbol{h}^{(\ell-1)} + \boldsymbol{b}^{(\ell)}\Big), \qquad \ell = 1, \ldots, L, \tag{2.22}$$

where $W^{(\ell)} \in \mathbb{R}^{d_\ell \times d_{\ell-1}}$ and $\boldsymbol{b}^{(\ell)} \in \mathbb{R}^{d_\ell}$ are trainable weights and biases, and $\phi^{(\ell)}$ is a nonlinear activation function applied elementwise. The network output

is $\boldsymbol{y} = f_{\boldsymbol{\theta}}(\boldsymbol{x}) \equiv \boldsymbol{h}^{(L)}$, and the full parameter set is

$$\boldsymbol{\theta} = \left\{W^{(\ell)}, \boldsymbol{b}^{(\ell)}\right\}_{\ell=1}^{L} . \tag{2.23}$$

The total parameter count scales as

$$N_{\theta} = \sum_{\ell=1}^{L} \left(d_{\ell} d_{\ell-1} + d_{\ell}\right) . \tag{2.24}$$

The layer dimension d_{ℓ} is its *width*, and L is the network *depth*; the parameter count grows rapidly with both. This structure, together with the operations carried out by a single unit, is summarized in Fig. 2.6.

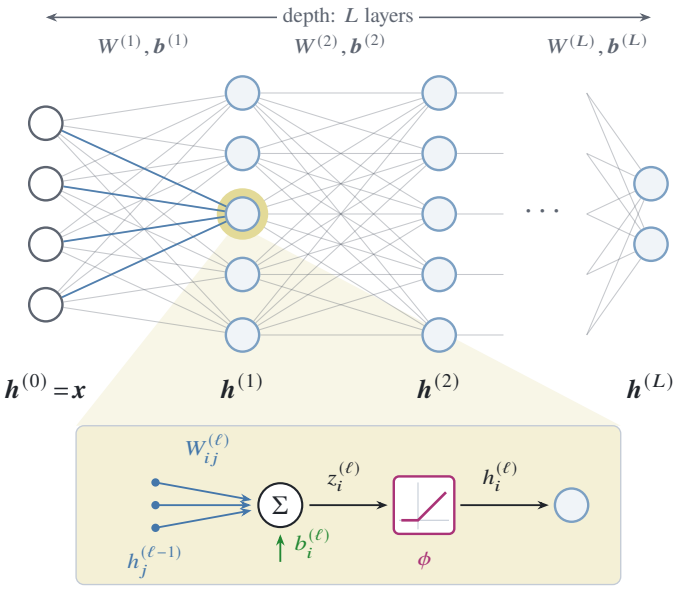

Figure 2.6: Structure of a multi-layer perceptron. Each column is one of the vectors of Eq. (2.22): the input $\boldsymbol{h}^{(0)} = \boldsymbol{x}$ (open circles), the hidden layers, and the output $\boldsymbol{h}^{(L)}$. The number of units in a column is its width d_{ℓ}, and the number of affine maps $W^{(\ell)}, \boldsymbol{b}^{(\ell)}$ separating the columns is the depth L. Every unit is connected to every unit of the preceding column, so $W^{(\ell)}$ carries $d_{\ell} d_{\ell-1}$ independent entries. The panel below expands the highlighted unit: it accumulates the weighted sum $z_i^{(\ell)} = \sum_j W_{ij}^{(\ell)} h_j^{(\ell-1)} + b_i^{(\ell)}$ over the entire preceding layer (blue wires above) and passes it through the elementwise nonlinearity ϕ, drawn here as the ReLU.

Activation functions. Nonlinearities are essential because a composition of affine maps alone reduces to a single affine map. Classical choices include tanh and sigmoid functions, but these can saturate and lead to vanishing gradients in deep networks. A widely used alternative is the piecewise-linear rectified linear unit (ReLU) [64],

$$\mathrm{ReLU}(x) = \max(0, x), \tag{2.25}$$

which often improves optimization due to its non-saturating positive branch. Smoother functions are now common, especially in transformers, including the Gaussian error linear unit (GELU) [65],

$$\mathrm{GELU}(x) = x\,\Phi(x) = \frac{x}{2}\left[1 + \mathrm{erf}\left(\frac{x}{\sqrt{2}}\right)\right], \tag{2.26}$$

where Φ is the cumulative distribution function of the standard normal distribution, and the Swish activation [66],

$$\mathrm{Swish}(x) = x\,\sigma(\beta x), \qquad \sigma(x) = \frac{1}{1 + \mathrm{e}^{-x}}, \tag{2.27}$$

typically with $\beta = 1$. The activation affects both expressiveness and trainability through gradient propagation and the distribution of intermediate activations.

Expressiveness and approximation properties. Classic *universal approximation* results show that even a single-hidden-layer MLP can uniformly approximate any continuous function on a compact domain under suitable conditions on the activation [67, 68]. However, these results are existential and do not guarantee efficient approximation: for generic high-dimensional function classes, the required width can still grow exponentially with input dimension. Depth can alleviate this by representing compositional structure more efficiently, and many functions admit substantially smaller deep representations than shallow ones (even when both are universal in principle).

For physics-motivated inputs such as lattice configurations, a plain MLP treats all input coordinates as unrelated degrees of freedom. As a result, symmetries (translations, point-group operations) and locality are not exploited automatically, and the network must *learn* these structures from data. This is often possible but typically parameter-inefficient compared to architectures with built-in equivariances (Sec. 2.3.10, for the NQS setting).

Training objective and backpropagation. Given training examples $\{(\boldsymbol{x}_i, \boldsymbol{t}_i)\}_{i=1}^{M}$ and a per-sample loss $\mathcal{L}_i(\boldsymbol{\theta})$ measuring the discrepancy between the prediction $f_{\boldsymbol{\theta}}(\boldsymbol{x}_i)$ and the target $\boldsymbol{t}_i$ (e.g. the squared error $\|f_{\boldsymbol{\theta}}(\boldsymbol{x}_i) - \boldsymbol{t}_i\|^2$), MLP parameters are typically learned by minimizing a total loss function,

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{M} \sum_{i=1}^{M} \mathcal{L}_i(\boldsymbol{\theta}) + \lambda\, \Omega(\boldsymbol{\theta}). \tag{2.28}$$

Here Ω is a regularizer. A common choice is *weight decay*, $\Omega(\boldsymbol{\theta}) = \sum_\ell \|W^{(\ell)}\|_F^2$, where $\|\cdot\|_F$ is the Frobenius norm; its contribution to a gradient-descent update is $-2\eta\lambda W^{(\ell)}$, which shrinks weights not sustained by the data term. Gradients $\nabla_{\boldsymbol{\theta}}\mathcal{L}$ are computed efficiently by *backpropagation*, i.e. repeated application of the chain rule through the composition in Eq. (2.22) [16]. In modern implementations, this is handled by automatic differentiation frameworks such as JAX [69].

Remarks for training neural quantum states. NQS training usually replaces labeled targets by a *variational* objective such as the Hamiltonian Rayleigh quotient (Sec. 2.3.3), so the operator supplies the learning signal. Concretely, the basis configurations used to estimate the objective are sampled from the model-induced Born distribution $p_\psi(\boldsymbol{s}) \propto |\psi(\boldsymbol{s};\boldsymbol{\theta})|^2$ (Sec. 2.3.2). Unlike a fixed i.i.d. training set, this distribution changes with $\boldsymbol{\theta}$.

As a consequence, gradients are not obtained simply by differentiating an empirical mean over a static minibatch, because the sampling distribution itself depends on $\boldsymbol{\theta}$. Instead, one uses Monte Carlo estimators based on local quantities (e.g. local energies and logarithmic derivatives), which yield covariance-type gradient formulas (Sec. 2.3.4). The stochastic gradients contain finite-sample noise and, for MCMC, autocorrelation effects. Moreover, redundant wave-function parameters, including the physically irrelevant global phase and normalization, can make the optimization landscape ill-conditioned even when changes in the physical state are small. These features motivate techniques that are less central in classical machine learning, such as geometry-aware preconditioning with SR (Sec. 2.3.6) and explicit control of symmetry sectors and constraints during sampling and optimization (Sec. 2.3.10).

Initialization and regularization. Deep MLPs can be difficult to optimize when activations saturate or gradients explode or vanish, so layer-wise initialization schemes are chosen to preserve activation and gradient variance

approximately. For example, Glorot/Xavier initialization is widely used for tanh-like activations [18], while He initialization is commonly used for ReLU networks [70]. Weight decay, dropout, and early stopping help prevent overfitting in supervised settings; for NQS, damping an ill-conditioned stochastic update (cf. SR diagonal shifts in Sec. 2.3.6) is usually more important.

Overall, MLPs serve as a clean reference point: they provide a flexible nonlinear map from configurations to outputs, but their lack of built-in structure motivates the use of specialized architectures when the target problem exhibits known symmetries, locality, or compositional patterns.

2.2.2 Convolutional neural networks

A key limitation of multi-layer perceptrons is that they treat the input coordinates as an unstructured vector. For lattice problems this ignores local interactions, spatial correlations, and translation symmetries. *Convolutional neural networks* (CNNs) address this by introducing sparse local connectivity and *weight sharing*, which, when combined with boundary handling that preserves translations, builds translation equivariance directly into the architecture [63]. CNNs are therefore a natural baseline for learning on one- and two-dimensional lattices, and they appear frequently in NQS constructions, both for implicit wave-function parameterizations and for autoregressive models [31, 71].

Discrete convolution on a lattice. Consider an input field on a D-dimensional periodic lattice (typically $D = 1, 2$), represented as a tensor

$$\boldsymbol{x} \in \mathbb{R}^{L_1 \times \cdots \times L_D \times C_{\text{in}}}, \tag{2.29}$$

where C_{in} is the number of input channels (e.g. one-hot features for local basis states, or multiple physical feature maps). A convolutional layer produces an output tensor $\boldsymbol{y} \in \mathbb{R}^{L_1 \times \cdots \times L_D \times C_{\text{out}}}$ by applying a bank of local filters. For simplicity, write the lattice coordinate as $\boldsymbol{r}$ and a finite filter support as a set of offsets $\mathcal{K} \subset \mathbb{Z}^D$ (e.g. a $k \times k$ stencil in 2D). Then a standard (cross-correlation) convolution can be written as

$$y_{\boldsymbol{r},\,c} = b_c + \sum_{c'=1}^{C_{\text{in}}} \sum_{\boldsymbol{\delta} \in \mathcal{K}} W_{\boldsymbol{\delta},\,c,c'}\, x_{\boldsymbol{r}+\boldsymbol{\delta},\,c'}, \qquad c = 1, \ldots, C_{\text{out}}, \tag{2.30}$$

followed by an elementwise nonlinearity $y \mapsto \phi(y)$. The independence of $W_{\delta,c,c'}$ from $\boldsymbol{r}$ is the weight-sharing property. Periodic boundary conditions

correspond to interpreting $\boldsymbol{r} + \boldsymbol{\delta}$ modulo $(L_1, \ldots, L_D)$; alternatively one can use padding schemes for open boundaries. Equation (2.30) uses $\mathcal{O}(|\mathcal{K}| C_{\text{in}} C_{\text{out}})$ parameters independently of system size and, at fixed kernel and channel counts, costs linearly in the number of sites. Figure 2.7 illustrates the local support and shared weights behind this scaling.

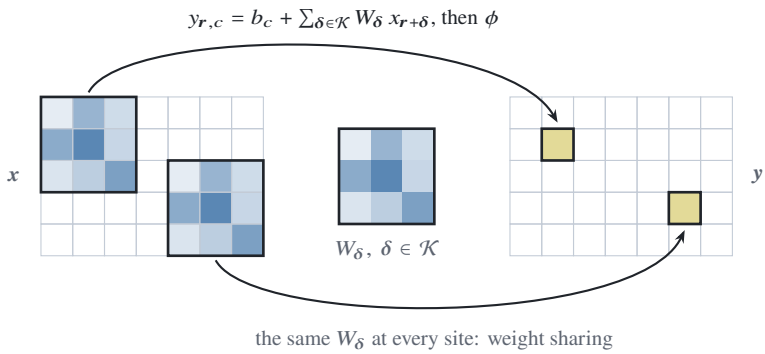

Figure 2.7: One convolutional layer on a two-dimensional lattice. Left: the input $\boldsymbol{x}$, with two windows of the offset set $\mathcal{K}$ of Eq. (2.30) marked on it. Right: the feature map $\boldsymbol{y}$, in which the two shaded sites are the ones those windows produce. Each arc runs from a window to the single site at its centre, so an output value depends on a bounded neighbourhood of the input rather than on all of it, in contrast to the fully connected layer of Fig. 2.6. The nine cells of the filter are shaded according to their weight, and the same nine shades reappear in both windows: one filter is reused at every site, which is the weight sharing that makes the parameter count independent of the lattice size. A single input and output channel is drawn; in general the layer holds $C_{\text{in}} C_{\text{out}}$ such filters.

Translation equivariance and invariance. Let $\hat{T}_{\boldsymbol{R}}$ denote a lattice translation by vector $\boldsymbol{R}$ acting on fields by $(\hat{T}_{\boldsymbol{R}} \boldsymbol{x})_{\boldsymbol{r}} = \boldsymbol{x}_{\boldsymbol{r}-\boldsymbol{R}}$. Because the same filter is applied at every site, a convolutional layer is *equivariant* under translations whenever the boundary handling is itself translation preserving (for example on an infinite lattice or on a finite lattice with periodic boundaries):

$$\text{Conv}(\hat{T}_{\boldsymbol{R}} \boldsymbol{x}) = \hat{T}_{\boldsymbol{R}} \text{Conv}(\boldsymbol{x}) \qquad \text{for all translations } \boldsymbol{R}. \tag{2.31}$$

Intermediate feature maps should be equivariant, whereas many scalar NQS outputs in the trivial momentum sector should be translation *invariant*. The

latter can be obtained by global sum or mean pooling:

$$z_c = \frac{1}{N} \sum_{\boldsymbol{r}} y_{\boldsymbol{r},c}, \qquad N = \prod_{j=1}^{D} L_j, \tag{2.32}$$

followed by a small fully connected "readout" network producing the final output. Figure 2.8 shows this sequence of equivariant feature extraction and invariant pooling.

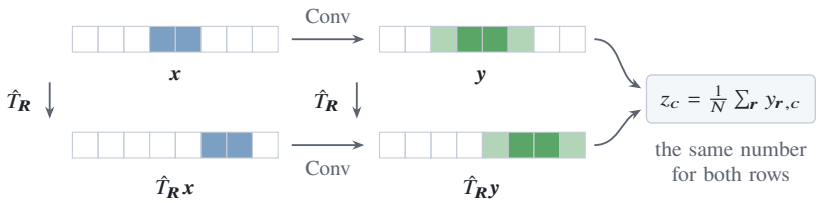

Figure 2.8: Translation equivariance, Eq. (2.31), on a periodic chain. Moving right applies the convolutional layer and moving down translates by $\boldsymbol{R}$; the two routes from the top left agree, because the same filter acts at every site. Colour intensity stands for the value carried by a site, and the response is wider than the input motif because the filter spans three sites. Global pooling, Eq. (2.32), then sends either feature map to the same number: this is how a stack of equivariant layers is turned into the invariant output a scalar readout requires.

Receptive field and depth. Each convolutional layer aggregates information only from a local neighborhood $\mathcal{K}$. Stacking layers increases the *receptive field*: after ℓ layers, each site-dependent feature can depend on inputs within a radius that grows with ℓ (roughly linearly for fixed kernel size and stride). This captures correlations over increasing distances, although global dependencies may still require dilated convolutions or attention.

Strides, padding, and multi-scale structure. Computer-vision CNNs often downsample by strided convolutions or pooling, but this can be undesirable when every lattice site is a physical degree of freedom and translation equivariance should be exact. NQS therefore often use stride-1, same-size convolutions with periodic padding; multiscale designs remain possible when boundary conditions and symmetries are handled consistently.

Residual networks and skip connections. Deep CNNs can suffer from vanishing or exploding gradients and redundant transformations. *Residual networks* (ResNets) mitigate this with skip connections [72]. Instead of learning a direct map $\boldsymbol{h}^{(\ell)} = F^{(\ell)}(\boldsymbol{h}^{(\ell-1)})$, a residual block learns a *residual* update

$$\boldsymbol{h}^{(\ell)} = \boldsymbol{h}^{(\ell-1)} + F^{(\ell)}(\boldsymbol{h}^{(\ell-1)}), \tag{2.33}$$

where $F^{(\ell)}$ is typically a small stack of convolutions and nonlinearities (often with normalization layers). The identity path propagates gradients and represents near-identity maps cheaply, biasing the block toward incremental refinements. Residual CNNs are therefore useful when large receptive fields require many stacked local layers.

CNNs as NQS building blocks. CNNs match the locality and translation symmetry common to lattice Hamiltonians. For implicit NQS, a CNN can map the configuration $\boldsymbol{s}$ (encoded as $\boldsymbol{x}(\boldsymbol{s})$) to one or more scalar outputs interpreted as $\log|\psi(\boldsymbol{s})|$ and, when needed, the phase. For autoregressive NQS, CNNs can be adapted to respect causality by masking (analogous to masked convolutions in autoregressive image models), so that the conditional $p(s_k|\boldsymbol{s}_{<k})$ depends only on the chosen prefix ordering [71]. This illustrates a recurring trade-off: enforcing symmetries by convolution is computationally efficient and parameter-effective, but additional constraints (such as autoregressive causality, fixed particle number, or targeting nontrivial momentum sectors) typically require either explicit projection/symmetrization or specialized equivariant constructions.

Limitations and extensions. While translation equivariance is built in, standard CNNs do not automatically incorporate other lattice symmetries such as rotations and reflections. One can enforce these either by explicit group averaging (Sec. 2.3.10) or by using group-equivariant convolutional layers that are equivariant under a larger symmetry group [73]. Moreover, convolution primarily encodes *local* mixing; representing sign structures, topological features, or long-range entanglement patterns can still be challenging and may require deeper networks, larger receptive fields, or architectures with more global communication mechanisms.

2.2.3 Transformer

A *transformer* replaces fixed local connectivity with content-dependent global communication through *self-attention* [74]. Although introduced for sequences, it applies whenever an input can be represented as tokens, such as words, image patches, or lattice sites. In contrast to convolutions, attention does not impose locality: each position can directly incorporate information from all other positions in a single layer, which can be advantageous for representing long-range correlations. In this thesis, transformers are mainly relevant in their autoregressive form, where they provide the proposal network used for sampling in Chapter 5.

Token embeddings. Here each lattice site is a token whose vocabulary is the local basis. A configuration $\boldsymbol{s} = (s_1, \ldots, s_N)$ becomes a sequence in a chosen site ordering, for example row by row on a square lattice. Each token is represented by a vector $\boldsymbol{x}_i \in \mathbb{R}^{d_{\text{model}}}$, where d_{model} is the *model dimension* (also called the embedding dimension), and the N vectors are collected into a matrix

$$X = \begin{bmatrix} \boldsymbol{x}_1^\top \\ \vdots \\ \boldsymbol{x}_N^\top \end{bmatrix} \in \mathbb{R}^{N \times d_{\text{model}}}. \tag{2.34}$$

For discrete inputs, an *embedding lookup* selects $\boldsymbol{x}_i$ from a trainable table containing one vector per local basis state. This is the learned counterpart of the fixed one-hot encoding of Sec. 2.2.1, and is the encoding used by the autoregressive sampling network of Chapter 5. For continuous inputs, $\boldsymbol{x}_i$ may instead be produced by a small feature encoder. Unlike CNNs, self-attention without positional information is permutation-equivariant, so some notion of position must be injected to distinguish sites. This is commonly achieved by adding a positional encoding $\boldsymbol{p}_i$ to each token, $\boldsymbol{x}_i \mapsto \boldsymbol{x}_i + \boldsymbol{p}_i$, where $\boldsymbol{p}_i$ can be learned or fixed (e.g. sinusoidal encodings).

Scaled dot-product self-attention. Self-attention maps X to a sequence of the same length by forming at each position a weighted average over *all* positions. Three learned projections produce *queries*, *keys*, and *values*:

$$Q = XW^Q, \qquad K = XW^K, \qquad V = XW^V, \tag{2.35}$$

with $W^Q, W^K \in \mathbb{R}^{d_{\text{model}} \times d_k}$ and $W^V \in \mathbb{R}^{d_{\text{model}} \times d_v}$, so that $Q, K \in \mathbb{R}^{N \times d_k}$ and $V \in \mathbb{R}^{N \times d_v}$. The i-th rows of these matrices, $\boldsymbol{q}_i, \boldsymbol{k}_i \in \mathbb{R}^{d_k}$ and $\boldsymbol{v}_i \in \mathbb{R}^{d_v}$, are the query, key, and value vectors of position i. Define the attention output as

$$\text{Attn}(Q, K, V) \equiv \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right) V, \tag{2.36}$$

where $QK^\top \in \mathbb{R}^{N \times N}$ holds the dot products between all pairs of queries and keys, and softmax is applied row-wise, so that each row of the *attention matrix* $A = \text{softmax}(QK^\top/\sqrt{d_k})$ is a probability distribution over the N positions. The result $\text{Attn}(Q, K, V) \in \mathbb{R}^{N \times d_v}$ again has one row per position. In component form, with a_{ij} the entries of A, the output $\boldsymbol{z}_i \in \mathbb{R}^{d_v}$ at position i is

$$\boldsymbol{z}_i = \sum_{j=1}^{N} a_{ij}\, \boldsymbol{v}_j, \qquad a_{ij} = \frac{\exp\left(\boldsymbol{q}_i \cdot \boldsymbol{k}_j/\sqrt{d_k}\right)}{\sum_{j'=1}^{N} \exp\left(\boldsymbol{q}_i \cdot \boldsymbol{k}_{j'}/\sqrt{d_k}\right)}. \tag{2.37}$$

The dot products define content-dependent similarities. The factor $1/\sqrt{d_k}$ compensates for their $\sqrt{d_k}$ scale and prevents the softmax from saturating as d_k grows [74]. This aggregation is illustrated for a single position in Fig. 2.9.

Multi-head attention. A single attention map uses the same weights a_{ij} for all features. *Multi-head attention* (MHA) instead runs H independently projected attention operations and mixes their outputs:

$$\text{head}_h = \text{Attn}\left(XW_h^Q, XW_h^K, XW_h^V\right), \qquad h = 1, \ldots, H, \tag{2.38}$$

$$\text{MHA}(X) = \text{Concat}(\text{head}_1, \ldots, \text{head}_H)\, W^O, \tag{2.39}$$

where each head has its own projections $W_h^Q, W_h^K \in \mathbb{R}^{d_{\text{model}} \times d_k}$ and $W_h^V \in \mathbb{R}^{d_{\text{model}} \times d_v}$ and returns an $N \times d_v$ matrix, the heads are concatenated along the feature axis into an $N \times Hd_v$ matrix, and $W^O \in \mathbb{R}^{(Hd_v) \times d_{\text{model}}}$ maps the result back to $\text{MHA}(X) \in \mathbb{R}^{N \times d_{\text{model}}}$. The standard choice is $d_k = d_v = d_{\text{model}}/H$, allowing different heads to specialize to distinct interaction patterns while the H heads together cost about as much as one full-width head.

Transformer block structure. A standard transformer layer combines multi-head attention with a position-wise feed-forward network (FFN), together with residual connections and layer normalization (LN). Denoting the layer input by

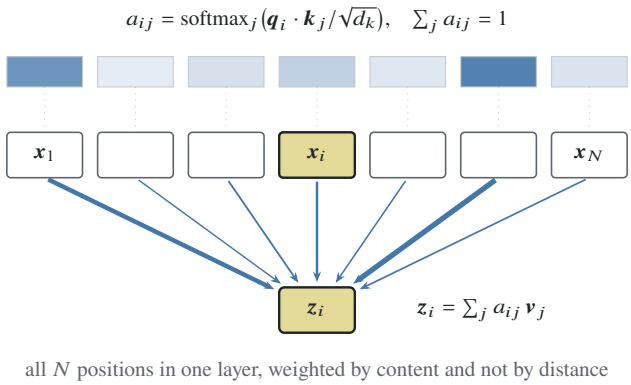

Figure 2.9: Self-attention at a single position, Eq. (2.36). The strip is one row of the attention matrix, the weights a_{ij} at fixed i, shaded in proportion to their value; the arrows repeat the same weights as line widths and show how position i combines the value vectors of all N positions into $\boldsymbol{z}_i$. An output site of the convolutional layer of Fig. 2.7 is fed by a fixed window of the input; here every position contributes in a single layer, with weights set by content rather than by distance, so that the two largest ones drawn here fall far from i. The same holds at every i, and the N rows so obtained form the $N \times N$ matrix, hence the quadratic cost of attention.

$X^{(\ell-1)}$, a commonly used modern implementation is the *pre-norm* form [75],

$$\tilde{X}^{(\ell)} = X^{(\ell-1)} + \mathrm{MHA}\Big(\mathrm{LN}\Big(X^{(\ell-1)}\Big)\Big), \tag{2.40}$$

$$X^{(\ell)} = \tilde{X}^{(\ell)} + \mathrm{FFN}\Big(\mathrm{LN}\Big(\tilde{X}^{(\ell)}\Big)\Big), \tag{2.41}$$

where LN denotes layer normalization [76]: it acts on each token vector separately, shifting and rescaling its d_{model} components to zero mean and unit variance and then applying a learned elementwise scale and offset, which keeps the size of the activations under control as blocks are stacked. The FFN is applied independently at each position, i.e. to every row $\boldsymbol{x}_i$ of its input (written here as a column vector, in the convention of Eq. (2.22)):

$$\mathrm{FFN}(\boldsymbol{x}) = W_2\,\phi(W_1\boldsymbol{x} + \boldsymbol{b}_1) + \boldsymbol{b}_2, \tag{2.42}$$

with $W_1 \in \mathbb{R}^{d_{\mathrm{ff}} \times d_{\mathrm{model}}}$, $W_2 \in \mathbb{R}^{d_{\mathrm{model}} \times d_{\mathrm{ff}}}$, a hidden width d_{ff} that is typically a few times d_{model} (four times in the original transformer), and a nonlinearity ϕ (often GELU in modern transformer implementations). The attention sublayer provides

global mixing across positions, while the FFN provides a local nonlinear re-encoding at each position, as summarized in Fig. 2.10. Residual paths play a similar stabilizing role as in ResNets: they allow deep stacks of blocks while maintaining gradient flow.

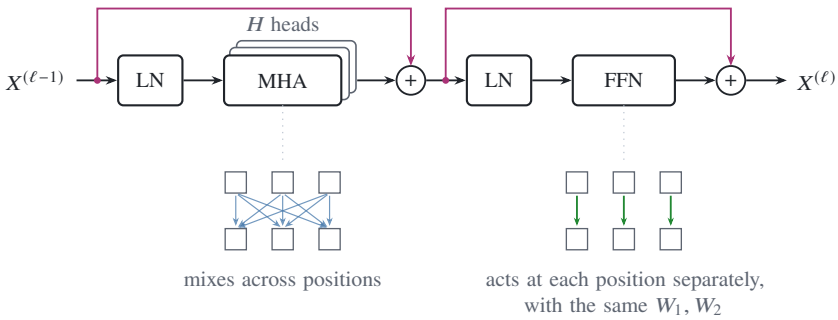

Figure 2.10: One pre-norm transformer block, Eqs. (2.41) and (2.42). The input is normalized, passed through multi-head attention, drawn as H stacked copies of one attention operation, and added back to the unmodified input by the first residual connection; the second normalization, the feed-forward network and the second residual connection repeat the pattern. The panels below show why the block has two sublayers: the attention arrows cross positions, so each output position depends on all of them, while the feed-forward arrows stay within a position, the same W_1, W_2 acting on each one independently. Neither sublayer changes the number of tokens or the model dimension d_{model}, so such blocks can be stacked.

Positional information and lattice structure. Because self-attention alone is permutation equivariant, positional encoding determines which spatial relationships the model represents efficiently. For one- and two-dimensional lattices, it is natural to encode positions by the lattice coordinates $\boldsymbol{r}_i$ rather than by an arbitrary sequential index. Moreover, for translation-invariant problems, *relative* position information (dependence on $\boldsymbol{r}_i - \boldsymbol{r}_j$) can be more appropriate than absolute encodings, because it makes attention scores depend on separations rather than absolute site labels [77]. For periodic lattices, such encodings should also be chosen consistently with the boundary conditions. Unless the architecture is equivariant, the transformer must learn lattice symmetries or be explicitly symmetrized as in Sec. 2.3.10.

Causal masking and autoregressive transformers. For autoregressive modeling, the network must respect an ordering constraint so that predictions at site i depend only on previous sites. In attention-based models this is enforced by a *causal mask* M, an $N \times N$ matrix with $M_{ij} = 0$ for allowed dependencies and $M_{ij} = -\infty$ for forbidden ones (typically $j > i$). The attention operation is then modified to

$$\mathrm{Attn}_{\mathrm{causal}}(Q, K, V) = \mathrm{softmax}\left(\frac{QK^{\mathsf{T}}}{\sqrt{d_k}} + M\right) V, \tag{2.43}$$

ensuring that each position aggregates information only from the permitted prefix: since $\exp(-\infty) = 0$, the masked pairs receive zero weight a_{ij} and the row normalization runs over the permitted positions only. Note that with the mask as written, position i still attends to itself. If the i-th input token is the local state s_i, the output at position i therefore sees the very variable it is meant to predict and cannot be used directly to parameterize the conditional $p(s_i|\boldsymbol{s}_{<i})$ of an autoregressive factorization (Sec. 2.3.9). Autoregressive transformers resolve this by shifting the input sequence by one position: a (typically learnable) start token is prepended and the last token is dropped, so that the i-th input is s_{i-1} (or the start token for $i = 1$) and the output at position i depends on $s_1, \ldots, s_{i-1}$ only. A final position-wise linear map from $\mathbb{R}^{d_{\mathrm{model}}}$ to the number of local basis states, followed by a softmax, then turns the output at position i into the normalized conditional $p(s_i|\boldsymbol{s}_{<i})$, and the product of these conditionals is the autoregressive distribution of Eq. (2.95). This mechanism underlies autoregressive transformer NQS constructions, including the sampling network of Chapter 5.

Scaling and practical remarks. Full self-attention forms the $N \times N$ matrix QK^{T}, requiring $O(N^2)$ weights and comparable memory in standard implementations. For a lattice with $N = \prod_j L_j$ sites, this can become a bottleneck, especially in two dimensions. In return, attention provides a short communication path between distant sites: information from any pair of positions can be mixed in a single layer, whereas purely local convolutions require many layers to couple widely separated sites. Local, sparse, and linear attention variants aim to reduce this cost [78].

2.2.4 Other NQS-relevant architectures

Two further families provide historical context for NQS, although they are not used directly here.

Restricted Boltzmann machines. The first influential NQS construction was based on the *restricted Boltzmann machine* [21]. An RBM is a shallow bipartite Boltzmann machine [22, 23]: visible units represent the physical configuration and hidden units mediate their correlations. In the standard NQS construction, the hidden variables can be summed out analytically for a fixed configuration, yielding an explicit expression for the wave-function amplitude in terms of the visible configuration and the network parameters. RBMs demonstrated that simple neural ansätze can capture nontrivial correlations with competitive accuracy, but their shallow structure is less flexible than deeper feed-forward, convolutional, or attention-based models [31].

Recurrent neural networks. *Recurrent neural networks* (RNNs) process an input sequence one element at a time while updating a hidden state that summarizes the previously seen prefix [79]. This makes them a natural architecture for autoregressive NQS [80]: the wave function or Born distribution is factorized into conditional factors associated with a chosen site ordering, enabling exact ancestral sampling as discussed in Sec. 2.3.9. The hidden state can compress an arbitrarily long prefix into a fixed-size representation without forming the all-to-all matrix of self-attention, giving standard RNN layers linear sequence-length scaling. On the other hand, the same sequential dependence limits parallelism across sites and can make optimization harder, since long-range information must be propagated through many recurrent steps and gradients can degrade unless suitable gated architectures are used (e.g. long short-term memory (LSTM) networks [81]). Transformers offer a different trade-off: they are often easier to train and better suited to accelerator-parallel computation, while RNNs can be more memory-efficient and may remain competitive when sequence length is the dominant bottleneck. RNNs and transformers are therefore complementary autoregressive architectures with different computational trade-offs.

2.2.5 Optimization

Neural networks are typically trained by minimizing a scalar objective $\mathcal{L}(\boldsymbol{\theta})$ with iterative first-order methods. In supervised learning, $\mathcal{L}$ is often a sample average of a per-sample loss over a fixed training set, as in Eq. (2.28). In physics applications such as NQS, $\mathcal{L}$ may instead be a variational objective estimated stochastically (e.g. the energy in VMC, Sec. 2.3.3). In either setting, practical scalability relies on the ability to compute and use gradients efficiently even for

models with millions of parameters.

Automatic differentiation. *Automatic differentiation* (AD) evaluates derivatives of a numerical computation exactly up to floating-point error by systematically applying the chain rule. For training, one typically needs the gradient of a scalar loss with respect to a high-dimensional parameter vector. *Reverse-mode* AD is efficient in this regime: its arithmetic cost is comparable to the forward evaluation, although intermediate quantities must be stored or recomputed. Backpropagation can be viewed as reverse-mode AD specialized to layered neural networks: one computes intermediate activations in the forward pass and then propagates sensitivities backward through the same sequence of operations. Practical AD frameworks (e.g. JAX) therefore allow the user to specify only the forward computation, while gradients are constructed automatically [69].

Gradient descent and stochastic gradients. The basic update rule is gradient descent,

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \Delta\boldsymbol{\theta}_t, \qquad \Delta\boldsymbol{\theta}_t = -\eta\, \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}_t), \tag{2.44}$$

with step size (learning rate) $\eta > 0$. For large datasets, one replaces the full gradient by a minibatch estimate $\boldsymbol{g}_t$, yielding stochastic gradient descent (SGD), $\Delta\boldsymbol{\theta}_t = -\eta\, \boldsymbol{g}_t$. In VMC and related settings, the corresponding stochastic gradient is obtained from Monte Carlo samples rather than from subsampling a fixed dataset. The resulting noise must be controlled by the step size and optimizer.

Momentum. A simple and widely used improvement over plain SGD is *momentum*, which replaces the raw gradient by an exponential moving average:

$$\boldsymbol{u}_t = \mu\, \boldsymbol{u}_{t-1} + (1-\mu)\, \boldsymbol{g}_t, \tag{2.45}$$

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta\, \boldsymbol{u}_t, \tag{2.46}$$

where $\mu \in [0, 1)$ controls the memory of past gradients. Momentum damps rapidly fluctuating gradient components and accelerates consistently aligned ones, which is useful in ill-conditioned landscapes. Variants such as Nesterov momentum modify the point at which the gradient is evaluated [82].

Adaptive methods and Adam. Many optimizers also adapt each parameter's step size from recent gradient magnitudes. A prominent example is Adam [83],

which combines momentum with per-parameter RMS scaling by maintaining exponential moving averages of the gradient and of its elementwise square:

$$\boldsymbol{m}_t = \beta_1 \, \boldsymbol{m}_{t-1} + (1 - \beta_1) \, \boldsymbol{g}_t, \tag{2.47}$$

$$\boldsymbol{v}_t = \beta_2 \, \boldsymbol{v}_{t-1} + (1 - \beta_2) \, \boldsymbol{g}_t^{\odot 2}, \tag{2.48}$$

where $\boldsymbol{g}_t^{\odot 2}$ denotes elementwise squaring. Because these averages are initialized at zero, Adam applies bias corrections $\hat{\boldsymbol{m}}_t = \boldsymbol{m}_t/(1-\beta_1^t)$ and $\hat{\boldsymbol{v}}_t = \boldsymbol{v}_t/(1-\beta_2^t)$, and updates parameters according to

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta \, \frac{\hat{\boldsymbol{m}}_t}{\sqrt{\hat{\boldsymbol{v}}_t} + \epsilon}, \tag{2.49}$$

with $\epsilon > 0$ preventing division by zero. In practice, $\beta_1 \simeq 0.9$ and $\beta_2 \simeq 0.999$ are common defaults. Adam is often effective when gradients are noisy or have heterogeneous scales across parameters, which is typical in deep networks and in stochastic objectives.

Practical remarks. Across these first-order optimizers, the learning rate η is often the most important hyperparameter. Standard techniques include learning-rate schedules (decay, cosine schedules), warmup phases for stabilizing early training in deep models, and gradient clipping to prevent occasional large updates. Regularization can be incorporated explicitly through weight decay and implicitly through the stochasticity of minibatches or Monte Carlo estimators. For NQS, these generic considerations are complemented by geometry-aware preconditioning methods such as SR (Sec. 2.3.6), which can substantially improve conditioning beyond what adaptive first-order methods provide.

2.3 Neural quantum states

2.3.1 Neural network as a wave function

Amplitude parameterization. A *neural quantum state* uses a neural network to parameterize the amplitudes of a many-body wave function in a chosen basis [21]. Using the product basis of Sec. 2.1.1, the wave function is

$$|\psi\rangle = \sum_{\boldsymbol{s}} \psi(\boldsymbol{s}) |\boldsymbol{s}\rangle \,, \tag{2.50}$$

where $\boldsymbol{s} = (s_1, \ldots, s_N)$ denotes a configuration of local basis states. The exponentially large table of amplitudes is replaced by a neural-network function $\psi(\boldsymbol{s}; \boldsymbol{\theta})$, also written $\psi_{\boldsymbol{\theta}}(\boldsymbol{s})$, with comparatively few variational parameters. In the simplest construction, the configuration $\boldsymbol{s}$ is given as input and the network output is interpreted as the corresponding generally unnormalized complex amplitude, or equivalently as its log-amplitude and phase, as sketched in Fig. 2.11. Autoregressive constructions represent the same wave function through normalized conditional factors and are discussed in Sec. 2.3.9.

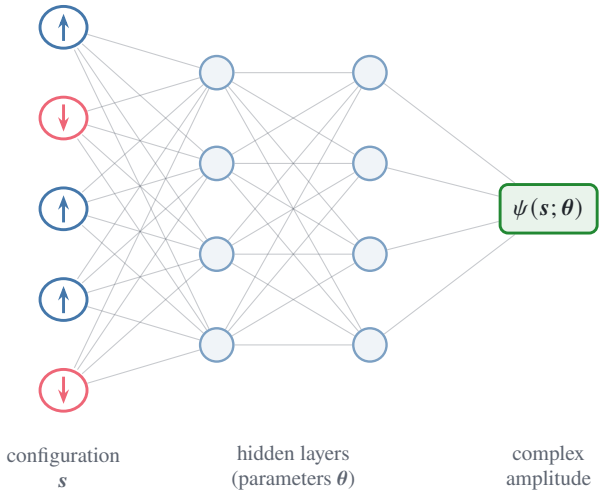

Figure 2.11: A neural quantum state in its simplest form. A basis configuration $\boldsymbol{s}$, here a set of spin-$\frac{1}{2}$ variables (blue for up, red for down), is fed to a neural network whose weights are the variational parameters $\boldsymbol{\theta}$, and the output is read as the complex amplitude $\psi(\boldsymbol{s}; \boldsymbol{\theta})$ (in practice its logarithm, as explained below). Nothing is stored per configuration: the exponentially large table of amplitudes is replaced by a function that is evaluated on the configurations one actually needs.

Two broad NQS constructions are used here. An *implicit* NQS outputs the amplitude or log-amplitude of a full configuration, leaving the Born distribution known only up to normalization. Throughout this thesis, and as is standard in the NQS literature [21, 31], implicit networks output $\log \psi(\boldsymbol{s}; \boldsymbol{\theta})$ rather than ψ itself. The amplitudes of a many-body state span many orders of magnitude across configurations, so the logarithm keeps the network output in a numerically benign range and avoids the underflow and overflow that exponentially small or large values would cause. Moreover, the quantities

entering VMC are naturally logarithmic: the amplitude ratios $\psi(\boldsymbol{s}')/\psi(\boldsymbol{s})$ in the local energy and the logarithmic derivatives $\partial_{\theta_k} \log \psi$ in the gradient (Sec. 2.3.4) are obtained directly as differences of network outputs and by plain backpropagation, respectively. An *autoregressive* NQS uses normalized conditional factors, enabling exact ancestral sampling at the cost of a causal factorization tied to a chosen ordering.

Compression and optimization. This compression is plausible because locality, symmetries, correlations, and entanglement strongly constrain physical low-energy states despite the exponentially large Hilbert space (Sec. 2.1.1). Neural networks provide a flexible nonlinear function class for capturing such structure without explicitly storing all basis amplitudes. The difficulty is basis dependent: in favourable cases one can choose a basis in which the target state is real and nonnegative, whereas frustrated or flux-carrying systems generally require a nontrivial sign or phase in addition to the amplitude profile. This distinction becomes important in several later benchmark problems.

Here the parameters $\boldsymbol{\theta}$ are optimized within VMC for ground and low-lying excited states. NQS have also been applied to tasks such as quantum state tomography and quantum dynamics [31].

2.3.2 Estimation of observables

Local estimators. When direct use of the full wave function is infeasible, observables can be estimated stochastically using the identity

$$\begin{aligned}
\langle \hat{A} \rangle &= \frac{\langle \psi | \hat{A} | \psi \rangle}{\langle \psi | \psi \rangle} = \frac{\sum_{\boldsymbol{s}} \sum_{\boldsymbol{s}'} \psi^*(\boldsymbol{s}) \psi(\boldsymbol{s}') \langle \boldsymbol{s} | \hat{A} | \boldsymbol{s}' \rangle}{\sum_{\boldsymbol{s}} |\psi(\boldsymbol{s})|^2} \\
&= \frac{\sum_{\boldsymbol{s}} |\psi(\boldsymbol{s})|^2 \sum_{\boldsymbol{s}'} \frac{\psi(\boldsymbol{s}')}{\psi(\boldsymbol{s})} \langle \boldsymbol{s} | \hat{A} | \boldsymbol{s}' \rangle}{\sum_{\boldsymbol{s}} |\psi(\boldsymbol{s})|^2} \\
&= \left\langle \sum_{\boldsymbol{s}'} \frac{\psi(\boldsymbol{s}')}{\psi(\boldsymbol{s})} \langle \boldsymbol{s} | \hat{A} | \boldsymbol{s}' \rangle \right\rangle \\
&= \langle A_{\mathrm{loc}}(\boldsymbol{s}) \rangle \,. \qquad (2.51)
\end{aligned}$$

Here we defined the *local estimator*

$$A_{\mathrm{loc}}(\boldsymbol{s}) = \sum_{\boldsymbol{s}'} \frac{\psi(\boldsymbol{s}')}{\psi(\boldsymbol{s})} A_{\boldsymbol{s}\boldsymbol{s}'}, \qquad A_{\boldsymbol{s}\boldsymbol{s}'} = \langle \boldsymbol{s} | \hat{A} | \boldsymbol{s}' \rangle. \qquad (2.52)$$

Here and below, angular brackets applied to configuration-dependent quantities denote expectation values with respect to the Born distribution $p_\psi(\boldsymbol{s}) = |\psi(\boldsymbol{s})|^2/\sum_{\boldsymbol{s}'}|\psi(\boldsymbol{s}')|^2$.

In practice, configurations are sampled from p_ψ and their local estimators are averaged. This is efficient when $A_{\boldsymbol{s}\boldsymbol{s}'}$ is sparse, so the defining sum contains few nonzero terms. This sparsity requirement is typically satisfied by the Hamiltonians discussed in Sec. 2.1, as they generally involve only local interactions.

Products and variances. Similarly, expectation values of operator products can be estimated using the following identity for Hermitian operators $\hat{A}$ and $\hat{B}$:

$$\begin{aligned}
\langle \hat{A}\hat{B}\rangle &= \frac{\sum_{\boldsymbol{s}'}\sum_{\boldsymbol{s}''}\psi(\boldsymbol{s}')\psi^*(\boldsymbol{s}'')(\hat{A}\hat{B})_{\boldsymbol{s}''\boldsymbol{s}'}}{\sum_{\boldsymbol{w}}|\psi(\boldsymbol{w})|^2} \\
&= \frac{\sum_{\boldsymbol{s}'}\sum_{\boldsymbol{s}''}\psi(\boldsymbol{s}')\psi^*(\boldsymbol{s}'')\sum_{\boldsymbol{s}}A_{\boldsymbol{s}''\boldsymbol{s}}B_{\boldsymbol{s}\boldsymbol{s}'}}{\sum_{\boldsymbol{w}}|\psi(\boldsymbol{w})|^2} \\
&= \frac{\sum_{\boldsymbol{s}}|\psi(\boldsymbol{s})|^2\sum_{\boldsymbol{s}''}\frac{\psi^*(\boldsymbol{s}'')}{\psi^*(\boldsymbol{s})}A_{\boldsymbol{s}''\boldsymbol{s}}\sum_{\boldsymbol{s}'}\frac{\psi(\boldsymbol{s}')}{\psi(\boldsymbol{s})}B_{\boldsymbol{s}\boldsymbol{s}'}}{\sum_{\boldsymbol{w}}|\psi(\boldsymbol{w})|^2} \\
&= \frac{\sum_{\boldsymbol{s}}|\psi(\boldsymbol{s})|^2 A^*_{\mathrm{loc}}(\boldsymbol{s})B_{\mathrm{loc}}(\boldsymbol{s})}{\sum_{\boldsymbol{w}}|\psi(\boldsymbol{w})|^2} \\
&= \left\langle A^*_{\mathrm{loc}}(\boldsymbol{s})B_{\mathrm{loc}}(\boldsymbol{s})\right\rangle .
\end{aligned} \tag{2.53}$$

Here B_{loc} is defined analogously to A_{loc}. In the special case $\hat{B} = \hat{A}$, this reduces to $\langle \hat{A}^2\rangle = \langle |A_{\mathrm{loc}}|^2\rangle$, which implies that the variance of $\hat{A}$ can be estimated from the same Monte Carlo sample of local estimators:

$$\mathrm{var}(\hat{A}) = \langle \hat{A}^2\rangle - \langle \hat{A}\rangle^2 = \langle |A_{\mathrm{loc}}|^2\rangle - |\langle A_{\mathrm{loc}}\rangle|^2 = \mathrm{var}(A_{\mathrm{loc}}). \tag{2.54}$$

2.3.3 Variational Monte Carlo

Variational objective. This work mainly employs NQS within *variational Monte Carlo*, which combines the variational principle with Monte Carlo estimation to optimize a parameterized trial state. Its standard form targets the ground state. Excited-state extensions require modified variational objectives and are discussed in Sec. 2.3.11. The method is based on the *variational principle*, which states that for any trial wave function (ansatz) $|\psi(\boldsymbol{\theta})\rangle$ depending on a set of parameters $\boldsymbol{\theta}$, the expectation value of the Hamiltonian provides an upper

bound to the ground-state energy,

$$E(\boldsymbol{\theta}) = \frac{\langle \psi(\boldsymbol{\theta}) | \hat{H} | \psi(\boldsymbol{\theta}) \rangle}{\langle \psi(\boldsymbol{\theta}) | \psi(\boldsymbol{\theta}) \rangle} \geq E_0, \tag{2.55}$$

where E_0 denotes the exact ground-state energy.

Energy estimation and training. The parameters are optimized by minimizing the variational energy, written using the preceding formalism as

$$E(\boldsymbol{\theta}) = \langle E_{\mathrm{loc}}(\boldsymbol{s}; \boldsymbol{\theta}) \rangle, \qquad E_{\mathrm{loc}}(\boldsymbol{s}; \boldsymbol{\theta}) = \sum_{\boldsymbol{s}'} \frac{\psi(\boldsymbol{s}'; \boldsymbol{\theta})}{\psi(\boldsymbol{s}; \boldsymbol{\theta})} H_{\boldsymbol{s}\boldsymbol{s}'}, \tag{2.56}$$

where $H_{\boldsymbol{s}\boldsymbol{s}'} = \langle \boldsymbol{s} | \hat{H} | \boldsymbol{s}' \rangle$ and the average is taken with respect to the normalized Born distribution $p_\psi(\boldsymbol{s})$.

The variational principle applies to the exact objective $E(\boldsymbol{\theta})$: its minimum over the chosen variational family is an upper bound to E_0. In practice, VMC accesses this objective and its derivatives through stochastic estimates, so the quality of the resulting approximation depends both on the expressiveness of the variational ansatz and on the reliability of the sampling procedure. In the context of NQS, the combination of flexible neural network architectures with VMC optimization has been shown to yield accurate results for a wide range of strongly correlated quantum systems [31]. In this work, VMC serves as the central framework for training neural network wave functions and for benchmarking different sampling and optimization strategies. Figure 2.12 summarizes a single VMC iteration and locates the components developed in the remaining sections of this chapter.

2.3.4 Energy gradient

Logarithmic derivatives. First-order methods such as stochastic gradient descent or Adam require the energy gradient [83]. Rather than differentiating a finite-sample estimate directly, one rewrites the exact gradient as a Born-distribution expectation. Writing

$$E = \frac{D}{Z}, \qquad D = \sum_{\boldsymbol{s},\boldsymbol{s}'} \psi^*(\boldsymbol{s}) H_{\boldsymbol{s}\boldsymbol{s}'} \psi(\boldsymbol{s}'), \qquad Z = \sum_{\boldsymbol{s}} |\psi(\boldsymbol{s})|^2, \tag{2.57}$$

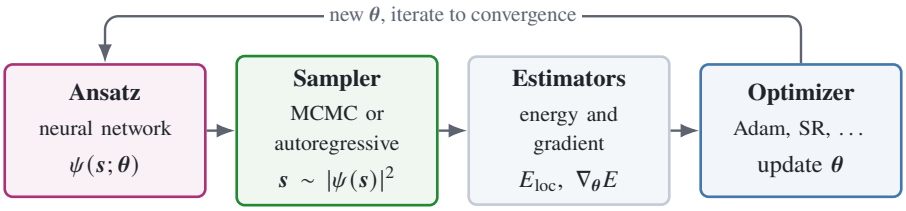

Figure 2.12: One iteration of variational Monte Carlo. The ansatz supplies amplitudes, the sampler draws configurations from the Born distribution $|\psi(\boldsymbol{s})|^2$, these configurations are used to estimate the energy and its gradient, and the optimizer updates the parameters; the loop then repeats with the updated ansatz. The three coloured stages are the ones the practitioner designs, and they correspond to the three bottlenecks around which this thesis is organized: representational bias enters through the ansatz, sampling through the sampler, and trainability through the optimizer. Because the stages are coupled only through the samples and the parameters, each can be replaced independently, which is what the method chapters do.

and defining the logarithmic derivatives

$$O_k(\boldsymbol{s}) = \frac{\partial}{\partial \theta_k} \log \psi(\boldsymbol{s}), \tag{2.58}$$

one finds, for real parameters $\boldsymbol{\theta}$ and a Hermitian Hamiltonian,

$$\begin{aligned} \frac{\partial D}{\partial \theta_k} &= 2\,\mathrm{Re}\left\{ \sum_{\boldsymbol{s},\boldsymbol{s}'} \frac{\partial \psi^*(\boldsymbol{s})}{\partial \theta_k} H_{\boldsymbol{s}\boldsymbol{s}'} \psi(\boldsymbol{s}') \right\} \\ &= 2\,\mathrm{Re}\left\{ \sum_{\boldsymbol{s}} |\psi(\boldsymbol{s})|^2 O_k^*(\boldsymbol{s}) E_{\mathrm{loc}}(\boldsymbol{s}) \right\}, \end{aligned} \tag{2.59}$$

$$\begin{aligned} \frac{\partial Z}{\partial \theta_k} &= 2\,\mathrm{Re}\left\{ \sum_{\boldsymbol{s}} \frac{\partial \psi^*(\boldsymbol{s})}{\partial \theta_k} \psi(\boldsymbol{s}) \right\} \\ &= 2\,\mathrm{Re}\left\{ \sum_{\boldsymbol{s}} |\psi(\boldsymbol{s})|^2 O_k^*(\boldsymbol{s}) \right\}. \end{aligned} \tag{2.60}$$

Using the quotient rule then gives

$$\frac{\partial E}{\partial \theta_k} = 2\,\mathrm{Re}\left\{ \left\langle O_k^* E_{\mathrm{loc}} \right\rangle - \left\langle O_k^* \right\rangle \left\langle E_{\mathrm{loc}} \right\rangle \right\}. \tag{2.61}$$

The same samples estimate the energy and Eq. (2.61), while automatic differentiation provides the logarithmic derivatives $O_k(\boldsymbol{s})$ [69]. For complex-valued optimization parameters one would instead use Wirtinger derivatives. In this thesis we restrict attention to real parameters.

2.3.5 Imaginary time evolution

Projection onto the ground state. Imaginary time evolution (ITE), obtained formally through $t \to -\mathrm{i}\tau$ in the Schrödinger equation, provides a VMC-related view of ground-state optimization. Its evolution operator $\mathrm{e}^{-\tau\hat{H}}$ is non-unitary. For a time-independent Hamiltonian, the imaginary-time Schrödinger equation reads

$$\frac{\partial}{\partial\tau}|\psi(\tau)\rangle = -\hat{H}|\psi(\tau)\rangle, \tag{2.62}$$

with formal solution

$$|\psi(\tau)\rangle = \mathrm{e}^{-\tau\hat{H}}|\psi(0)\rangle. \tag{2.63}$$

Expanding the initial state in the eigenbasis $\hat{H}|n\rangle = E_n|n\rangle$, $|\psi(0)\rangle = \sum_n c_n|n\rangle$, one finds

$$|\psi(\tau)\rangle = \mathrm{e}^{-\tau E_0}\left(c_0|0\rangle + \sum_{n>0} c_n \mathrm{e}^{-\tau(E_n-E_0)}|n\rangle\right). \tag{2.64}$$

If the ground state is non-degenerate and the initial state has nonzero overlap with it ($c_0 \neq 0$), the normalized state therefore approaches $|0\rangle$ as $\tau \to \infty$. More generally, ITE projects onto the component of the initial state within the ground-state eigenspace. This projector principle underlies projector Monte Carlo methods and also motivates the SITE training strategy introduced in Chapter 3.

Normalized flow and VMC connection. Normalizing the state as $|\phi(\tau)\rangle = |\psi(\tau)\rangle/\sqrt{\langle\psi(\tau)|\psi(\tau)\rangle}$ makes the connection to energy minimization explicit:

$$\frac{\partial}{\partial\tau}|\phi(\tau)\rangle = -\left(\hat{H} - E(\tau)\right)|\phi(\tau)\rangle, \qquad E(\tau) = \langle\phi(\tau)|\hat{H}|\phi(\tau)\rangle. \tag{2.65}$$

A direct differentiation then shows that the energy decreases monotonically,

$$\frac{\mathrm{d}E}{\mathrm{d}\tau} = -2\left(\langle\hat{H}^2\rangle - \langle\hat{H}\rangle^2\right) = -2\,\mathrm{var}(\hat{H}) \leq 0, \tag{2.66}$$

and is stationary only when $\mathrm{var}(\hat{H}) = 0$, i.e. when $|\phi(\tau)\rangle$ is an eigenstate of $\hat{H}$. Within VMC the same sample estimates the energy and its variance, since

Sec. 2.3.2 gives $\mathrm{var}(\hat{H}) = \mathrm{var}(E_{\mathrm{loc}})$.

In practice, one often restricts the dynamics to a variational family $|\psi(\boldsymbol{\theta})\rangle$. Projecting the exact ITE flow onto this manifold, for example in the sense of the time-dependent variational principle, yields an evolution equation for the parameters $\boldsymbol{\theta}$. A first-order time discretization of this projected dynamics leads, up to conventions and regularization choices, to the stochastic-reconfiguration update discussed in the next subsection [27, 28].

2.3.6 Stochastic reconfiguration

Unlike standard first-order optimizers, VMC can exploit the geometry of the variational manifold. Nonlinear and redundant NQS parameterizations are often ill-conditioned: the same physical state change can correspond to very different parameter-space gradients. A widely used method for incorporating the variational-manifold geometry is *stochastic reconfiguration*, originally introduced in the context of correlated electron wave functions [27, 29]. In modern terms, SR is a *natural-gradient* method that replaces the Euclidean parameter metric by the metric induced by $|\psi(\boldsymbol{\theta})\rangle$.

Derivation. The key idea is to choose an update $\boldsymbol{\theta} \to \boldsymbol{\theta} + \delta\boldsymbol{\theta}$ such that the resulting change of the variational state best approximates an infinitesimal imaginary-time evolution step. Up to an overall normalization, imaginary-time evolution gives

$$|\psi(\tau + \delta\tau)\rangle \approx \left(1 - \delta\tau\hat{H}\right)|\psi(\tau)\rangle. \tag{2.67}$$

Within the variational family $|\psi(\boldsymbol{\theta})\rangle$, the corresponding first-order change induced by a parameter update is

$$|\psi(\boldsymbol{\theta} + \delta\boldsymbol{\theta})\rangle \approx |\psi(\boldsymbol{\theta})\rangle + \sum_k \delta\theta_k \frac{\partial}{\partial\theta_k}|\psi(\boldsymbol{\theta})\rangle. \tag{2.68}$$

Components parallel to $|\psi(\boldsymbol{\theta})\rangle$ only change the representative of the same physical ray and are projected out. SR chooses $\delta\boldsymbol{\theta}$ by minimizing the leading-order Fubini–Study distance between the projected changes. In the product basis, using $\partial_{\theta_k}\psi(\boldsymbol{s}) = \psi(\boldsymbol{s})O_k(\boldsymbol{s})$ and $\sum_{\boldsymbol{s}'} H_{\boldsymbol{s}\boldsymbol{s}'}\psi(\boldsymbol{s}') = \psi(\boldsymbol{s})E_{\mathrm{loc}}(\boldsymbol{s})$, the projected tangent vectors and the projected imaginary-time displacement have the amplitudes

$$\psi(\boldsymbol{s})\left[O_k(\boldsymbol{s}) - \langle O_k\rangle\right] \qquad \text{and} \qquad -\delta\tau\,\psi(\boldsymbol{s})\left[E_{\mathrm{loc}}(\boldsymbol{s}) - \langle E_{\mathrm{loc}}\rangle\right], \tag{2.69}$$

respectively, where the subtracted averages are the projections onto $|\psi(\boldsymbol{\theta})\rangle$. Up to the normalization $\langle\psi|\psi\rangle$, the squared distance to be minimized is therefore

$$\sum_{\boldsymbol{s}} |\psi(\boldsymbol{s})|^2 \left| \sum_k \delta\theta_k \left[O_k(\boldsymbol{s}) - \langle O_k\rangle\right] + \delta\tau \left[E_{\mathrm{loc}}(\boldsymbol{s}) - \langle E_{\mathrm{loc}}\rangle\right] \right|^2 . \tag{2.70}$$

For real parameters, setting its derivative with respect to each $\delta\theta_k$ to zero yields a linear system of equations for the update direction:

$$\sum_l S_{kl}\, \delta\theta_l = \eta\, f_k, \tag{2.71}$$

where the matrix S is given by the covariance of the logarithmic derivatives,

$$S_{kl} = \mathrm{Re}\left\{\left\langle O_k^* O_l\right\rangle - \left\langle O_k^*\right\rangle \langle O_l\rangle\right\} . \tag{2.72}$$

The force vector f_k is a covariance between the local energy and the logarithmic derivatives,

$$f_k = -2\,\mathrm{Re}\left\{\left\langle O_k^* E_{\mathrm{loc}}\right\rangle - \left\langle O_k^*\right\rangle \langle E_{\mathrm{loc}}\rangle\right\} . \tag{2.73}$$

In the conventions of Eqs. (2.72) and (2.73) the imaginary-time step enters as $\eta = \delta\tau/2$; in practice η is treated as a freely chosen step size (learning rate). All averages are taken with respect to the normalized Born distribution $p_\psi(\boldsymbol{s})$, using the same Monte Carlo samples that are used to estimate the energy. The geometric content of this construction is sketched in Fig. 2.13.

Relation to gradient descent. The connection to the energy-gradient expression derived in the previous section is immediate:

$$\frac{\partial E}{\partial \theta_k} = -f_k. \tag{2.74}$$

Thus, a standard gradient-descent step corresponds to $\delta\boldsymbol{\theta} \propto -\nabla_{\boldsymbol{\theta}} E$, whereas SR applies an additional preconditioning by a matrix S^{-1},

$$\delta\boldsymbol{\theta} \propto -S^{-1}\nabla_{\boldsymbol{\theta}} E, \tag{2.75}$$

which typically improves stability and efficiency of the optimization. In the language of information geometry, S is the Fubini–Study metric tensor for the normalized variational state, or equivalently the pure-state quantum Fisher information matrix up to a conventional overall factor [84], and SR is therefore

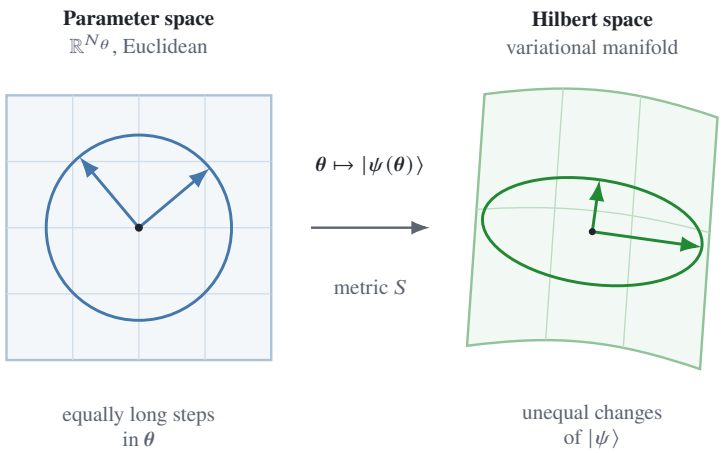

Figure 2.13: Why the geometry of the variational manifold matters. Left: in parameter space all steps of a given length are equivalent, so a plain gradient step treats every direction of $\boldsymbol{\theta}$ alike. Right: the map $\boldsymbol{\theta} \mapsto |\psi(\boldsymbol{\theta})\rangle$ stretches some directions more than others, so the circle of equally long parameter steps becomes an ellipse of unequal changes of the state. The metric S of Eq. (2.72) measures exactly this local distortion, and preconditioning with S^{-1} makes the update uniform in the Fubini–Study distance on the manifold rather than in the arbitrary Euclidean geometry of the parameters.

a natural-gradient method [85].

Regularization. In practice, the matrix S can be ill-conditioned due to finite sampling noise and parameter redundancies, especially for expressive neural networks. A standard stabilization is to apply a diagonal shift,

$$(S + \lambda I)\,\delta\boldsymbol{\theta} = \eta\,\boldsymbol{f}, \tag{2.76}$$

with $\lambda > 0$ chosen such that the linear system remains well conditioned. For large parameter counts, explicitly forming and inverting S is computationally expensive, scaling as $O(N_\theta^2)$ in memory and $O(N_\theta^3)$ in time in the worst case. Therefore, SR is often implemented using iterative solvers such as conjugate gradient, which require only matrix–vector products with S and can be estimated directly from the Monte Carlo sample. These practical considerations make SR significantly more costly than simple gradient-based methods, but it often converges in fewer optimization steps and can be crucial for obtaining accurate results in challenging regimes. In this work, SR serves as an important baseline optimizer for training neural network quantum states.

2.3.7 Minimum-step SR and kernel trick

SR is limited by the cost of solving

$$(S + \lambda I)\,\delta\boldsymbol{\theta} = \eta\,\boldsymbol{f}, \tag{2.77}$$

when the number of variational parameters N_θ becomes large. Modern NQS often have $N_\theta \sim 10^5$–10^7 parameters but only $M \sim 10^3$–10^4 Monte Carlo samples. Forming S is then infeasible, while matrix-free solvers repeatedly apply a noisy finite-sample metric in a much larger parameter space.

Two related works exploit this low-rank structure [34, 35]. Chen and Heyl's *minimum-step stochastic reconfiguration* (MinSR) selects a minimum-norm solution of the underdetermined finite-sample problem, while Rende *et al.* give an exact sample-space reformulation of diagonally shifted SR. The latter is derived below and then related to the minimum-step interpretation.

Low-rank form of the SR metric. Let $\{\boldsymbol{s}_i\}_{i=1}^{M}$ be a Monte Carlo sample drawn from $|\psi(\boldsymbol{s};\boldsymbol{\theta})|^2$. Define the centered logarithmic derivatives

$$\Delta O_k(\boldsymbol{s}_i) = O_k(\boldsymbol{s}_i) - \overline{O}_k, \qquad \overline{O}_k = \frac{1}{M}\sum_{i=1}^{M} O_k(\boldsymbol{s}_i), \tag{2.78}$$

and introduce the complex matrix $Y \in \mathbb{C}^{N_\theta \times M}$,

$$Y_{ki} = \frac{1}{\sqrt{M}}\,\Delta O_k(\boldsymbol{s}_i). \tag{2.79}$$

Then the stochastic estimate of the SR metric can be written compactly as

$$S \approx \mathrm{Re}\left\{YY^\dagger\right\}. \tag{2.80}$$

For complex ψ, $\mathrm{rank}(S) \leq 2M$, with centering reducing it further, so S is highly singular when $N_\theta \gg M$.

To eliminate the explicit real part in Eq. (2.80), we decompose $Y = Y_\mathrm{R} + \mathrm{i}Y_\mathrm{I}$ with $Y_\mathrm{R}, Y_\mathrm{I} \in \mathbb{R}^{N_\theta \times M}$ and obtain

$$S \approx Y_\mathrm{R}Y_\mathrm{R}^\mathsf{T} + Y_\mathrm{I}Y_\mathrm{I}^\mathsf{T} = XX^\mathsf{T}, \qquad X = [\,Y_\mathrm{R}\;\; Y_\mathrm{I}\,] \in \mathbb{R}^{N_\theta \times 2M}. \tag{2.81}$$

For a strictly real wave function, $Y_\mathrm{I} = 0$ and the construction reduces to an $N_\theta \times M$ matrix.

Force vector in the same basis. Similarly, using the centered local energies

$$\Delta E_i = E_\mathrm{loc}(\boldsymbol{s}_i) - \overline{E}_\mathrm{loc}, \qquad \overline{E}_\mathrm{loc} = \frac{1}{M}\sum_{i=1}^{M} E_\mathrm{loc}(\boldsymbol{s}_i), \tag{2.82}$$

the force vector from Eq. (2.73) can be estimated as

$$f_k \equiv -\frac{\partial E}{\partial \theta_k} \approx -\frac{2}{M}\,\mathrm{Re}\left\{\sum_{i=1}^{M} \Delta E_i^* \,\Delta O_k(\boldsymbol{s}_i)\right\}. \tag{2.83}$$

Introducing the complex vector $\varepsilon \in \mathbb{C}^M$,

$$\varepsilon_i = -\frac{2}{\sqrt{M}}\,\Delta E_i^*, \tag{2.84}$$

Eq. (2.83) becomes $\boldsymbol{f} \approx \mathrm{Re}\{Y\boldsymbol{\varepsilon}\}$. With the real/imaginary decomposition $\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}_\mathrm{R} + \mathrm{i}\boldsymbol{\varepsilon}_\mathrm{I}$, one finds an explicit real representation,

$$\boldsymbol{f} \approx Y_\mathrm{R}\boldsymbol{\varepsilon}_\mathrm{R} - Y_\mathrm{I}\boldsymbol{\varepsilon}_\mathrm{I} = X\boldsymbol{g}, \qquad \boldsymbol{g} = \begin{bmatrix} \boldsymbol{\varepsilon}_\mathrm{R} \\ -\boldsymbol{\varepsilon}_\mathrm{I} \end{bmatrix} \in \mathbb{R}^{2M}. \tag{2.85}$$

Sample-space SR update via a matrix identity. Using Eqs. (2.81) and (2.85), the stabilized SR update can be written as

$$\delta\boldsymbol{\theta} = \eta \left(XX^\mathsf{T} + \lambda I_{N_\theta}\right)^{-1} X\boldsymbol{g}. \tag{2.86}$$

At this point one can apply the linear algebra identity

$$\left(XX^\mathsf{T} + \lambda I_{N_\theta}\right)^{-1} X = X\left(X^\mathsf{T}X + \lambda I_{2M}\right)^{-1}, \qquad \lambda > 0, \tag{2.87}$$

to obtain an *exact* alternative SR update,

$$\delta\boldsymbol{\theta} = \eta\, X\left(X^\mathsf{T}X + \lambda I_{2M}\right)^{-1} \boldsymbol{g}. \tag{2.88}$$

Equation (2.88) replaces the $N_\theta \times N_\theta$ system by a $2M \times 2M$ solve and one multiplication by the $N_\theta \times 2M$ matrix X. For real-valued wave functions, $2M$ is replaced by M.

The matrix $X^\mathsf{T}X$ is the Gram matrix of the sample features, so Eq. (2.88) is a kernel-matrix reformulation based on their inner products rather than the full metric XX^T. This is the sense in which the method is related to the "kernel trick" [35]. Because $X^\mathsf{T}X$ and XX^T share the same nonzero eigenvalues, this rewrite introduces no approximation beyond the original Monte Carlo estimation and the chosen diagonal shift.

For $\lambda = 0$ and $N_\theta \gg M$, many parameter updates induce the same tangent-space change. MinSR resolves this non-uniqueness by choosing the smallest Euclidean-norm update, replacing the inverse in Eq. (2.88) by a Moore–Penrose pseudoinverse [34]. The term "minimum-step" therefore refers to the smallest parameter displacement compatible with the projected imaginary-time step.

Algorithmic form and complexity. In practice, Eq. (2.88) can be implemented in three steps:

1. Construct X and $\boldsymbol{g}$ from the Monte Carlo sample $\{\boldsymbol{s}_i\}$.

2. Solve the *sample-space* linear system

$$\left(X^{\mathsf{T}}X + \lambda I_{2M}\right)\boldsymbol{t} = \boldsymbol{g}. \tag{2.89}$$

3. Update parameters via

$$\delta\boldsymbol{\theta} = \eta\, X\boldsymbol{t}. \tag{2.90}$$

The cost is dominated by forming the matrix $X^{\mathsf{T}}X$, which scales as $O(N_\theta M^2)$ operations and requires storing $O(N_\theta M)$ numbers, while solving the resulting system scales as $O(M^3)$. This makes exact finite-sample SR feasible in the deep-learning regime $N_\theta \gg M$, while remaining mathematically equivalent to the original diagonally shifted formulation in Eq. (2.86).

2.3.8 Markov chain Monte Carlo sampling

Estimating VMC energies, gradients, and observables requires samples from the Born distribution:

$$p_\psi(\boldsymbol{s}) = \frac{|\psi(\boldsymbol{s})|^2}{\sum_{\boldsymbol{s}'} |\psi(\boldsymbol{s}')|^2}. \tag{2.91}$$

Because evaluating all configurations is infeasible, a generic implicit NQS is usually sampled indirectly.

Metropolis–Hastings algorithm. A standard choice is Metropolis–Hastings MCMC, which constructs a chain with stationary distribution p_ψ [86, 87]. Starting from an initial configuration $\boldsymbol{s}_t$ and a proposal distribution $q(\boldsymbol{s} \to \boldsymbol{s}')$, one repeats the following steps:

1. Sample a proposal configuration $\boldsymbol{s}'$ from $q(\boldsymbol{s}_t \to \boldsymbol{s}')$.

2. Compute the acceptance probability

$$a(\boldsymbol{s}_t \to \boldsymbol{s}') = \min\left(1, \frac{|\psi(\boldsymbol{s}')|^2\, q(\boldsymbol{s}' \to \boldsymbol{s}_t)}{|\psi(\boldsymbol{s}_t)|^2\, q(\boldsymbol{s}_t \to \boldsymbol{s}')}\right). \tag{2.92}$$

   The normalization of p_ψ is not needed because it cancels in the ratio.

3. Draw a uniform random number $u \in [0, 1]$.

4. If $u \leq a$, accept the proposal by setting $\boldsymbol{s}_{t+1} = \boldsymbol{s}'$; otherwise, reject it and set $\boldsymbol{s}_{t+1} = \boldsymbol{s}_t$.

For symmetric proposals, $q(\boldsymbol{s} \to \boldsymbol{s}') = q(\boldsymbol{s}' \to \boldsymbol{s})$, this simplifies to

$$a(\boldsymbol{s}_t \to \boldsymbol{s}') = \min\left(1, \frac{|\psi(\boldsymbol{s}')|^2}{|\psi(\boldsymbol{s}_t)|^2}\right). \tag{2.93}$$

The resulting transition kernel satisfies detailed balance with respect to p_ψ. If the resulting chain is irreducible and aperiodic, then p_ψ is its unique stationary distribution, and empirical averages along sufficiently long runs converge to the desired expectation values.

Limitations. Despite its asymptotic correctness, Metropolis–Hastings has practical limitations for NQS. Successive samples are correlated, reducing the effective sample size and requiring equilibration and longer runs. Efficiency also depends strongly on the proposal $q(\boldsymbol{s} \to \boldsymbol{s}')$. Local proposals such as single-spin flips are easy to implement and often have reasonable acceptance rates, but they can explore configuration space only diffusively. More global moves may mix faster, yet they are harder to design and often suffer from low acceptance probabilities, especially when the calculation is restricted to a symmetry sector or another constrained Hilbert-space submanifold. When $|\psi(\boldsymbol{s})|^2$ is sharply peaked or multi-modal, tunneling between relevant regions can become very slow, as sketched in Fig. 2.14. Although the chain may still be ergodic in principle, finite runs can then misrepresent the target distribution and yield unreliable estimates of observables and their gradients.

Finally, each Markov chain is inherently sequential because every transition depends on the previous state. Independent chains can be run in parallel, but the samples within each chain remain correlated. This can make MCMC a computational bottleneck when large numbers of effectively independent samples are needed repeatedly during optimization.

2.3.9 Sampling with autoregressive NQS

An autoregressive ansatz instead permits direct sampling of its normalized Born distribution,

$$p(\boldsymbol{s}) = |\psi(\boldsymbol{s})|^2, \tag{2.94}$$

which is the Born distribution p_ψ of Sec. 2.3.2 with unit normalization; the subscript is dropped below. Direct sampling follows by factorizing it into

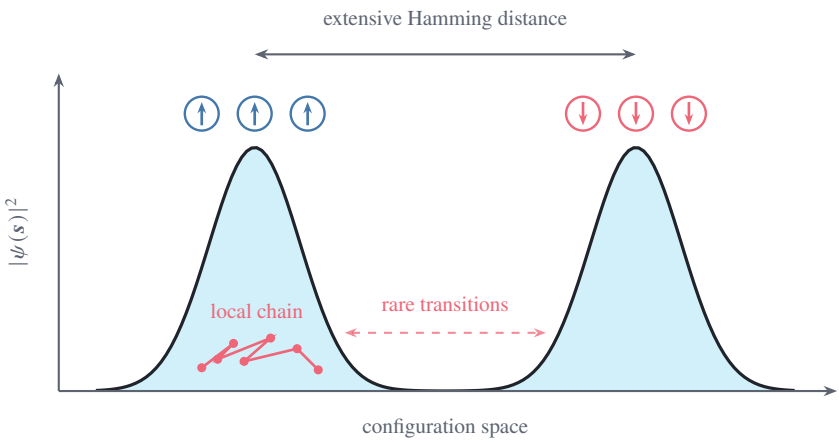

Figure 2.14: How a local Markov chain can misrepresent a multi-modal Born distribution. The sketch corresponds to the ferromagnetic regime of the transverse-field Ising model, where $|\psi(\boldsymbol{s})|^2$ concentrates near two configurations, all spins up and all spins down. These are separated by an extensive Hamming distance, meaning that the number of single-spin flips required to get from one to the other grows with system size. Every such path passes through configurations of negligible weight, so the corresponding proposals are almost always rejected and a chain built from local moves remains in whichever mode it started from. The run then reports an effectively unimodal distribution, and the resulting bias is invisible from the sampled data alone.

normalized conditionals [71, 80]:

$$p(\boldsymbol{s}) = \prod_{k=1}^{N} p(s_k \mid \boldsymbol{s}_{<k}), \qquad \boldsymbol{s}_{<k} = (s_1, \ldots, s_{k-1}), \tag{2.95}$$

where for each prefix $\boldsymbol{s}_{<k}$ the neural network produces the conditional distribution over the local basis states of site k, for example using a transformer [74] or a recurrent neural network (e.g. LSTM [81]). Each factor is a categorical distribution over the local states of site k,

$$\sum_{s_k} p(s_k|\boldsymbol{s}_{<k}) = 1 \qquad \text{for all } \boldsymbol{s}_{<k}, \tag{2.96}$$

so $p(\boldsymbol{s})$ is exactly normalized and supports exact ancestral sampling without a Markov chain.

Ancestral sampling. Sampling proceeds sequentially according to the factorization in Eq. (2.95):

1. Set an empty prefix $\boldsymbol{s}_{<1} = ()$.
2. For $k = 1, \ldots, N$:
   a) Evaluate the conditional distribution $p(s_k|\boldsymbol{s}_{<k})$ using the autoregressive network.
   b) Sample s_k from this distribution and append it to the prefix.
3. Return $\boldsymbol{s} = (s_1, \ldots, s_N)$.

The result is distributed exactly according to $p(\boldsymbol{s})$; independent random-number streams produce independent draws without burn-in or autocorrelations. One such sweep is illustrated in Fig. 2.15.

Evaluation of phase. If phase is important, one can factorize the wave function itself into a product of *normalized conditional wavefunctions* [71],

$$\psi(\boldsymbol{s}) = \prod_{k=1}^{N} \psi_k(s_k \mid \boldsymbol{s}_{<k}), \tag{2.97}$$

where for any fixed prefix $\boldsymbol{s}_{<k}$ the conditional satisfies the ℓ_2 normalization

$$\sum_{s'_k} \left|\psi_k(s'_k|\boldsymbol{s}_{<k})\right|^2 = 1. \tag{2.98}$$

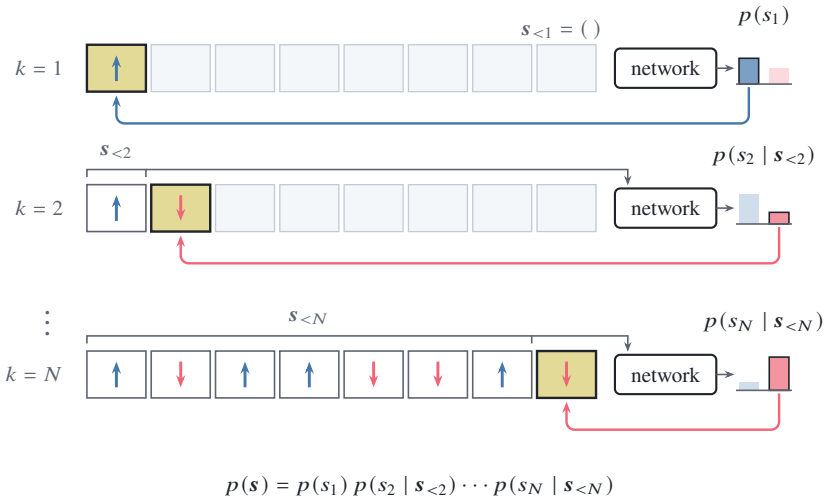

Figure 2.15: One ancestral sweep of an autoregressive NQS, Eq. (2.95), for spin-$\frac{1}{2}$ degrees of freedom. Each row is one step, drawn as the loop it is: the sites already fixed form the prefix $s_{<k}$ (bracket), which is fed to the network; the network returns the normalized conditional $p(s_k|s_{<k})$ over the local basis states of site k (bars, blue for up and red for down); and the value drawn from it is written back into the highlighted site, enlarging the prefix that the next step starts from. It is one network, evaluated once per site. The bracket stops short of site k, so the sites still open to its right never enter, which is the causal structure the factorization requires. The draw follows the conditional rather than its maximum, as in the step $k = 2$ shown here. After N steps the product of the conditionals is exactly $p(s)$, so the sweep returns one exact sample with no burn-in and no autocorrelation, unlike the chain of Fig. 2.14; the price is that the N steps must be taken in sequence.

Given network outputs $v_{k,s}(s_{<k}) \in \mathbb{C}$ (one complex logit for each local value s at fixed prefix), the conditional is parameterized as

$$\psi_k(s_k \mid s_{<k}) = \frac{\exp\big(v_{k,s_k}(s_{<k})\big)}{\sqrt{\sum_{s'} \left|\exp\big(v_{k,s'}(s_{<k})\big)\right|^2}}\,, \tag{2.99}$$

so that the full log-amplitude can be accumulated as

$$\log \psi(s) = \sum_{k=1}^{N} \left[v_{k,s_k}(s_{<k}) - \frac{1}{2} \log\left(\sum_{s'} \left|\exp\big(v_{k,s'}(s_{<k})\big)\right|^2 \right) \right]. \tag{2.100}$$

The magnitude and phase follow from the real and imaginary parts of $\log \psi(\boldsymbol{s})$:

$$|\psi(\boldsymbol{s})| = \exp\big(\mathrm{Re}\ \log \psi(\boldsymbol{s})\big), \qquad \varphi(\boldsymbol{s}) = \mathrm{Im}\ \log \psi(\boldsymbol{s}) = \sum_{k=1}^{N} \mathrm{Im}\ v_{k,s_k}(\boldsymbol{s}_{<k}), \tag{2.101}$$

since the ℓ_2-normalization term is real. Equivalently, since $|\exp(v)|^2 = \exp(2\ \mathrm{Re}\ v)$, the imaginary parts affect only the phase and not the sampling probabilities. Sampling is performed from the Born probability $p(\boldsymbol{s}) = |\psi(\boldsymbol{s})|^2 = \prod_k \big|\psi_k(s_k|\boldsymbol{s}_{<k})\big|^2$, i.e., sequentially from the conditional probabilities $\big|\psi_k(\cdot|\boldsymbol{s}_{<k})\big|^2$.

Sampling in constrained Hilbert spaces. Many lattice problems are restricted to a subspace, for example fixed particle number or fixed total magnetization (total S^z, i.e. a fixed number of up spins). Autoregressive sampling can incorporate such constraints by masking out locally forbidden choices during generation and renormalizing the remaining probabilities. Concretely, let $m_k(s_k, \boldsymbol{s}_{<k}) \in \{0, 1\}$ indicate whether choosing s_k is compatible with the constraint given the current prefix and the remaining sites. Provided the mask is prefix-consistent, so that the denominator remains nonzero at every step, one then samples from the masked conditional distribution

$$p^{(\mathrm{c})}(s_k|\boldsymbol{s}_{<k}) = \frac{m_k(s_k, \boldsymbol{s}_{<k})\, p(s_k|\boldsymbol{s}_{<k})}{\sum_{s'_k} m_k(s'_k, \boldsymbol{s}_{<k})\, p(s'_k|\boldsymbol{s}_{<k})}\,, \tag{2.102}$$

which guarantees that every generated configuration satisfies the global constraint, without rejection.

Practical considerations. Autoregressive sampling removes the need to design proposal moves and avoids metastability due to poor mixing, which can be particularly beneficial when $|\psi(\boldsymbol{s})|^2$ is sharply peaked or multi-modal. Independent samples can also be generated in parallel across a batch on accelerators.

A remaining limitation is sequential dependence: one configuration requires N ordered conditional evaluations, although cached architectures can reuse intermediate computations. The ordering also affects how efficiently correlations and symmetries are represented; for example, a 2D raster scan breaks rotational symmetry unless further symmetrized.

Because the factorization is tied to this ordering, it is not permutation- or

translation-invariant by default, and explicit group averaging becomes more expensive.

This trade-off between ansatz flexibility and sampling convenience is central for the present thesis. The variational states used later for explicit symmetry projection and for the determinant multi-state construction of Sec. 2.3.11 remain implicit, because those operations need not preserve an autoregressive factorization. Chapter 5 therefore introduces a separate autoregressive sampler, allowing exact proposal generation without constraining the target NQS itself.

2.3.10 Enforcing symmetries

Many lattice Hamiltonians have a symmetry group G, for example lattice translations, reflections, and global spin inversion. In this subsection we focus on discrete symmetries of finite systems, so the relevant formulas reduce to finite sums. Only unitary symmetries are needed here; antiunitary operations such as time reversal require a modified treatment. If $\hat{U}_g$ denotes the unitary operator representing the symmetry operation $g \in G$, symmetry of the Hamiltonian means

$$[\hat{H}, \hat{U}_g] = 0 \qquad \forall g \in G. \tag{2.103}$$

Consequently, symmetry operations map eigenstates to states of the same energy, and the Hilbert space decomposes into symmetry sectors, i.e. subspaces carrying definite irreducible representations (irreps) of G. For familiar Abelian symmetries these sectors are labeled by quantum numbers such as momentum or $\mathbb{Z}_2$ parity.

In VMC it is often advantageous to restrict the variational family to the sector of interest. This improves the conditioning of the optimization, prevents the ansatz from drifting into symmetry-broken directions, and can reduce the complexity needed to represent the target state accurately.

NQS enforce symmetries either *by construction* or by explicit group averaging, and the approaches can be combined.

Architecture-based symmetry enforcement. One can choose a network whose functional form is itself invariant or equivariant under G. Here *invariant* means that the output does not change when the input configuration is transformed, while *equivariant* means that the output transforms in the corresponding way when the input is transformed. Equivariance is useful because it can be converted into invariance, for example by summing over symmetry-related

positions at the end of the network.

For a configuration $\boldsymbol{s} = (s_1, \ldots, s_N)$ on a periodic lattice, lattice translations act by permuting site indices. Convolutional neural networks (CNNs) naturally implement translation equivariance through weight sharing; when combined with a translation-invariant pooling (e.g. summation over spatial positions) in the final stage, the resulting scalar output becomes translation invariant. More generally, one can use group-equivariant layers that are explicitly constructed to be equivariant with respect to lattice point-group symmetries (rotations/reflections) [73].

Discrete internal symmetries can also be built in directly. For example, consider a $\mathbb{Z}_2$ spin-flip symmetry $\boldsymbol{s} \mapsto -\boldsymbol{s}$. If the desired state is even under this transformation, it is sufficient to ensure that the network output satisfies $\psi(\boldsymbol{s}) = \psi(-\boldsymbol{s})$. This can be achieved by using spin-flip invariant input features (e.g. products $s_i s_j$) or, more generally, by constructing the network from spin-flip invariant intermediate features. The same general idea applies to other exact constraints, such as fixed particle number or gauge constraints: one can parameterize the ansatz in terms of allowed degrees of freedom or, in autoregressive models, enforce the constraint through prefix-compatible masks.

Architecture-based enforcement is computationally efficient: evaluating $\psi(\boldsymbol{s})$ costs essentially the same as an unconstrained network. However, it can be nontrivial to design architectures that capture the full symmetry group while retaining high expressiveness. Moreover, for autoregressive models the required causal structure can conflict with naive symmetric designs, since a fixed site ordering explicitly breaks permutation and rotation symmetries unless special symmetry-aware constructions are used.

Symmetry projection by group averaging. Alternatively, one evaluates a base NQS $\psi(\boldsymbol{s})$ on all symmetry-related configurations and combines the amplitudes with weights that impose the desired transformation law.

Let $g \in G$ act on basis states as $\hat{U}_g|\boldsymbol{s}\rangle = |g\boldsymbol{s}\rangle$ (possibly up to a phase that can be absorbed into the definition of the action). If the target state should be fully symmetric, one simply averages with equal weights:

$$\psi_{\text{sym}}(\boldsymbol{s}) = \frac{1}{|G|} \sum_{g \in G} \psi(g\boldsymbol{s}) \,, \tag{2.104}$$

which guarantees $\psi_{\text{sym}}(g\boldsymbol{s}) = \psi_{\text{sym}}(\boldsymbol{s})$ for all $g \in G$.

To target a more general symmetry sector, labelled by an irrep Γ, one uses

the standard projector

$$\hat{P}^{(\Gamma)} = \frac{d_\Gamma}{|G|} \sum_{g \in G} \chi^{(\Gamma)}(g)^* \hat{U}_g, \tag{2.105}$$

where d_Γ is the dimension of the irrep and $\chi^{(\Gamma)}(g)$ is its character. For one-dimensional sectors, $\chi^{(\Gamma)}(g)$ is simply the sign or phase that the state acquires under the symmetry operation g. Applied to the variational state, this yields the projected amplitudes

$$\psi^{(\Gamma)}(\boldsymbol{s}) = \langle \boldsymbol{s} | \hat{P}^{(\Gamma)} | \psi \rangle = \frac{d_\Gamma}{|G|} \sum_{g \in G} \chi^{(\Gamma)}(g)^* \psi(g^{-1}\boldsymbol{s}) \,. \tag{2.106}$$

For one-dimensional irreps this directly fixes the transformation law of the scalar amplitudes. For higher-dimensional irreps, the character projector selects the whole subspace transforming in that way, but not a particular partner state inside it; resolving those requires the more detailed matrix-element projectors built from the full representation matrices $D^{(\Gamma)}_{mn}(g)$.

A common special case is translational symmetry on a periodic lattice, where irreps are labeled by crystal momentum $\boldsymbol{k}$. Denoting a lattice translation by $\boldsymbol{R}$ as $\hat{T}_{\boldsymbol{R}}$ and the number of distinct translations (unit cells) by N_{cell}, the momentum-projected ansatz takes the form

$$\psi_{\boldsymbol{k}}(\boldsymbol{s}) = \frac{1}{N_{\text{cell}}} \sum_{\boldsymbol{R}} \mathrm{e}^{-\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{R}} \psi(\hat{T}_{\boldsymbol{R}}\boldsymbol{s}) \,. \tag{2.107}$$

This discrete Fourier sum satisfies

$$\psi_{\boldsymbol{k}}(\hat{T}_{\boldsymbol{R}}\boldsymbol{s}) = \mathrm{e}^{\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{R}} \psi_{\boldsymbol{k}}(\boldsymbol{s}), \tag{2.108}$$

so under a translation it picks up the phase expected for momentum $\boldsymbol{k}$.

The normalization prefactors in Eqs. (2.104)–(2.107) are not essential in VMC, since the energy and its gradients are invariant under global rescaling of ψ; nevertheless, they are convenient for numerical stability.

Computational trade-offs and practical implementation. Group averaging requires no architectural changes but increases wave-function evaluation cost by roughly $|G|$ because every amplitude becomes a sum over transformed configurations. This also makes local estimators more expensive, since ratios such as $\psi(\boldsymbol{s}')/\psi(\boldsymbol{s})$ become ratios of sums. The overhead is often manageable

because transformed inputs can be evaluated in one accelerator batch and many groups are modest in size. Translational projection on large periodic lattices is more expensive, since in that case $|G|$ grows with the number of unit cells, but it is still frequently feasible in VMC.

A common strategy uses architectural equivariance for easy symmetries and explicit averaging for the remainder. For example, a CNN backbone can capture local translational structure, while explicit averaging is reserved for point-group operations or a global $\mathbb{Z}_2$ symmetry. For target states in a nonzero-momentum sector, an explicit translational projection such as Eq. (2.107) is often the simplest option, although symmetry-adapted constructions can also target nontrivial momentum sectors.

Remarks for autoregressive NQS. For autoregressive models, amplitude projection remains valid but generally destroys the probability factorization and hence exact ancestral sampling. One must then sample $|\psi^{(\Gamma)}(\boldsymbol{s})|^2$ by MCMC or use symmetry-aware autoregressive constructions [71, 88]. This highlights a key difference between implicit and autoregressive NQS: enforcing symmetries by architecture is often the more natural choice when exact sampling is a primary goal.

2.3.11 Multi-state NQS for excited states

Optimizing several excited-state ansätze independently can cause *state collapse*, in which different networks converge to the same lowest-energy state. Moreover, many established multi-state VMC schemes recover individual energies by solving generalized eigenvalue problems from stochastic estimates of overlap and Hamiltonian matrices, which can lead to biased gradients or numerically unstable optimization.

A different approach, *natural excited states variational Monte Carlo* (NES-VMC) [89], avoids explicit orthogonalization, admits unbiased estimators of the objective and gradients, and maps the lowest-K problem to ground-state VMC on an *extended system*. A more rigorous formalization was later presented in Ref. [90], which casts the construction in terms of the Grassmannian geometry of Hilbert space. Below we present the original formulation in the lattice notation used throughout this thesis.

Total ansatz as a determinant. Let $\{\psi_a(\boldsymbol{s})\}_{a=1}^{K}$ be K (not necessarily orthogonal) single-state NQS ansätze. Introduce K independent copies of the

configuration space and denote a configuration of the extended system by $\boldsymbol{S} = (\boldsymbol{s}_1, \ldots, \boldsymbol{s}_K)$. Define the $K \times K$ matrix of amplitudes

$$\Psi(\boldsymbol{S})_{\ell a} \equiv \psi_a(\boldsymbol{s}_\ell), \qquad \ell, a = 1, \ldots, K. \tag{2.109}$$

Equivalently, the full matrix is

$$\Psi(\boldsymbol{S}) = \begin{pmatrix} \psi_1(\boldsymbol{s}_1) & \cdots & \psi_K(\boldsymbol{s}_1) \\ \vdots & \ddots & \vdots \\ \psi_1(\boldsymbol{s}_K) & \cdots & \psi_K(\boldsymbol{s}_K) \end{pmatrix}. \tag{2.110}$$

The *total* (multi-state) wave-function amplitude is then defined as the determinant

$$\Psi_{\text{tot}}(\boldsymbol{S}) \equiv \det \Psi(\boldsymbol{S}). \tag{2.111}$$

This resembles a Slater determinant with many-body wave functions in place of orbitals. If the ansätze become linearly dependent, $\det \Psi(\boldsymbol{S})$ vanishes, so the Born weight suppresses collapse without explicit orthogonality constraints.

Extended Hamiltonian and the variational objective. Let $\hat{H}$ be the original Hamiltonian acting on one copy. On the extended system, define

$$\hat{\mathcal{H}} \equiv \sum_{\ell=1}^{K} \hat{H}_\ell, \tag{2.112}$$

where $\hat{H}_\ell$ acts as $\hat{H}$ on $\boldsymbol{s}_\ell$ and as the identity on all other copies. NES-VMC minimizes the Rayleigh quotient of the total ansatz,

$$E_{\text{tot}} = \frac{\langle \Psi_{\text{tot}} | \hat{\mathcal{H}} | \Psi_{\text{tot}} \rangle}{\langle \Psi_{\text{tot}} | \Psi_{\text{tot}} \rangle}. \tag{2.113}$$

At the exact minimum, the span of $\{\psi_a\}$ coincides with the subspace of the lowest K eigenstates of $\hat{H}$, and E_{tot} equals the sum of the K lowest eigenenergies. The objective is invariant under invertible mixing of the single-state ansätze: $\psi_a \mapsto \sum_b A_{ba}\psi_b$ changes Ψ_{tot} only by the constant factor $\det A$. Thus the optimized $\{\psi_a\}$ determine a subspace and must be "demixed" to recover individual eigenstates.

Matrix local energy and unbiased Monte Carlo estimation. NES-VMC promotes the scalar local energy to a matrix by applying $\hat{H}$ to every ansatz in every copy:

$$\begin{aligned}(\hat{H}\Psi)(\boldsymbol{S})_{\ell a} &\equiv (\hat{H}\psi_a)(\boldsymbol{s}_\ell) \\ &= \sum_{\boldsymbol{s}'} H_{\boldsymbol{s}_\ell \boldsymbol{s}'}\, \psi_a(\boldsymbol{s}')\,.\end{aligned} \tag{2.114}$$

Then the *local energy matrix* is

$$\begin{aligned}E_{\mathrm{loc}}^{(\mathrm{mat})}(\boldsymbol{S}) &\equiv \Psi(\boldsymbol{S})^{-1}\,(\hat{H}\Psi)(\boldsymbol{S}) \\ &= \begin{pmatrix} \psi_1(\boldsymbol{s}_1) & \cdots & \psi_K(\boldsymbol{s}_1) \\ \vdots & \ddots & \vdots \\ \psi_1(\boldsymbol{s}_K) & \cdots & \psi_K(\boldsymbol{s}_K) \end{pmatrix}^{-1} \begin{pmatrix} \hat{H}\psi_1(\boldsymbol{s}_1) & \cdots & \hat{H}\psi_K(\boldsymbol{s}_1) \\ \vdots & \ddots & \vdots \\ \hat{H}\psi_1(\boldsymbol{s}_K) & \cdots & \hat{H}\psi_K(\boldsymbol{s}_K) \end{pmatrix}.\end{aligned} \tag{2.115}$$

The inverse is well defined on all configurations with nonzero sampling weight, since $\Psi(\boldsymbol{S})$ is singular exactly when $\Psi_{\mathrm{tot}}(\boldsymbol{S}) = 0$. Sampling extended configurations $\boldsymbol{S}$ from the probability distribution $p(\boldsymbol{S}) \propto |\Psi_{\mathrm{tot}}(\boldsymbol{S})|^2$ defines the energy matrix

$$E^{(\mathrm{mat})} \equiv \left\langle E_{\mathrm{loc}}^{(\mathrm{mat})}(\boldsymbol{S}) \right\rangle_{\boldsymbol{S}\sim|\Psi_{\mathrm{tot}}|^2}, \qquad E_{\mathrm{tot}} = \mathrm{Tr}\, E^{(\mathrm{mat})}. \tag{2.116}$$

The sample average of $E_{\mathrm{loc}}^{(\mathrm{mat})}$ estimates $E^{(\mathrm{mat})}$ without bias. Because Eq. (2.113) is an ordinary extended-system Rayleigh quotient, the gradient of $\mathrm{Tr}\, E^{(\mathrm{mat})}$ admits the usual unbiased VMC estimator.

Recovering individual energies and states (demixing). After optimization, the accumulated energy matrix is diagonalized,

$$E^{(\mathrm{mat})} = V\,\Lambda\,V^{-1}, \tag{2.117}$$

where $\Lambda = \mathrm{diag}(E_0, \ldots, E_{K-1})$ contains estimates of the K lowest individual energies, labelled in increasing order so that E_0 is the ground-state estimate, as elsewhere in this thesis, and the columns of V are the corresponding eigenvectors. The corresponding approximate eigenstates are obtained as linear combinations of the single-state ansätze,

$$\widetilde{\psi}_n(\boldsymbol{s}) = \sum_{a=1}^{K} \psi_a(\boldsymbol{s})\, V_{an}, \qquad n = 0, \ldots, K-1. \tag{2.118}$$

For nondegenerate levels, these combinations converge to orthogonal eigenstates; within degenerate subspaces the basis remains non-unique. Thus optimization determines the low-energy *subspace*, while diagonalization selects a basis within it.

Other observables and transition matrix elements. For any operator $\hat{A}$ acting on a single copy, one defines analogously to the local estimator of Sec. 2.3.2

$$A_{\text{loc}}^{(\text{mat})}(\boldsymbol{S}) \equiv \Psi(\boldsymbol{S})^{-1}\,(\hat{A}\Psi)(\boldsymbol{S}), \qquad A^{(\text{mat})} = \left\langle A_{\text{loc}}^{(\text{mat})}(\boldsymbol{S}) \right\rangle_{\boldsymbol{S}\sim|\Psi_{\text{tot}}|^2}. \tag{2.119}$$

To express observables in the same demixed basis as the energies, one transforms with the eigenvector matrix V from Eq. (2.117),

$$\widetilde{A}^{(\text{mat})} = V^{-1}\,A^{(\text{mat})}\,V. \tag{2.120}$$

The diagonal elements $\widetilde{A}_{nn}^{(\text{mat})}$ estimate $\langle\widetilde{\psi}_n|\hat{A}|\widetilde{\psi}_n\rangle$, while the off-diagonal elements encode transition matrix elements within the computed subspace.

Computational remarks. Relative to single-state VMC, the method uses a K-fold configuration space and evaluates all K^2 amplitudes $\psi_a(\boldsymbol{s}_\ell)$ and entries of $(\hat{H}\Psi)(\boldsymbol{S})$. The additional linear-algebra overhead is the repeated evaluation of a $K \times K$ determinant and matrix inverse in Eqs. (2.111) and (2.115), which scales as $O(K^3)$ per Monte Carlo sample. In practice, K is typically small (a few to a few tens), so the dominant cost usually remains the evaluation of the underlying neural-network wave functions.

2.4 Software and implementation

The numerical experiments reported in the following chapters share a common software stack, summarized here for reference. Chapter-specific deviations (alternative libraries, hardware) are noted where they occur.

Numerical framework. All code is written in Python [91]. Performance-critical NQS evaluation, local estimators, samplers, and gradients are implemented in JAX [69]. Its automatic differentiation provides the log-derivatives $O_k(\boldsymbol{s}) = \partial_{\theta_k} \log \psi(\boldsymbol{s};\boldsymbol{\theta})$ needed for energy, overlap-loss, and SR updates (Secs. 2.3.4 and 2.3.6), while JIT compilation and automatic vectorization

(*vmap*) parallelize configurations and Markov chains on an accelerator. Optimizer updates (Adam and its variants) are implemented through Optax [92]. Neural-network components are built either with Flax [93] or with Equinox [94], depending on the chapter.

Reference implementations. For variational baselines based on stochastic reconfiguration we use the NetKet package [95, 96], which provides tested implementations of MCMC samplers, the SR linear solver, and the underlying Hamiltonian operators. Exact-diagonalization reference energies for small lattices are obtained with QuSpin [97, 98]. Density matrix renormalization group calculations are performed with TeNPy [99], including extrapolations in the bond dimension where needed.

In-house NQS library. Apart from the NetKet SR baselines, the algorithms are implemented in one in-house JAX and Equinox package. Its modules define the lattices and Hamiltonian matrix elements, provide MLP, convolutional, residual, and Transformer components, wrap them into single-state, determinant multi-state, and autoregressive NQS, and implement exact, MCMC, ancestral, and importance-resampling samplers. A common trainer supplies the VMC loop used by direct energy minimization, SITE, and NIR.

The library is used to implement SITE (Chapter 3), the supervised-learning experiments and diagnostics for the HBHM performance gap (Chapter 4), and NIR with multi-state determinant NQS (Chapter 5). The Python environment is managed with uv.

Much of this functionality overlaps with NetKet [95, 96], but rebuilding the core VMC machinery clarified each component and made the standard loop easier to modify. In particular, SITE's adaptive fixed-target training, the HBHM supervised diagnostics, and the coupling of an autoregressive proposal to determinant multi-state NQS in NIR all required control over components usually fixed by a framework.

Testing and validation. The in-house code is checked at two levels. Unit tests compare lattice enumeration, matrix elements, amplitudes, samplers, energies, and gradients with hand-computed or fully enumerated small instances. End-to-end tests compare variational energies and wave functions with QuSpin ED and, beyond its reach, TeNPy DMRG; these calculations also provide the reference energies quoted later.

Hardware. NQS training in this thesis was carried out on a single Nvidia RTX 4090 GPU, while the CPU-bound exact-diagonalization and DMRG reference calculations were performed on an AMD Ryzen Threadripper 2990WX workstation. Wall-clock comparisons in the following chapters are reported with this setup in mind, and should be read as indicative rather than as fair benchmarks across hardware.

3 Supervised imaginary time evolution

This chapter is based on the results published in [P1].

3.1 Introduction

As discussed in Chapter 2, NQS provide a flexible variational parameterization of many-body wave functions (Sec. 2.3), while VMC offers a scalable way to estimate observables and optimize variational parameters from stochastic samples (Sec. 2.3.3). In this setting, the central practical challenge is often not expressiveness of the ansatz but *trainability*: the variational energy is evaluated through noisy Monte Carlo estimates, and the resulting optimization landscape in parameter space can be rugged and poorly conditioned [100]. Consequently, straightforward first-order optimization of the energy may converge slowly or become trapped in plateaus and saddle points, especially for expressive neural-network architectures with many parameters.

A widely used remedy is to incorporate information about the geometry of the variational manifold. SR (Sec. 2.3.6) implements this idea by replacing the raw gradient direction with a natural-gradient update obtained from a local metric tensor estimated from samples. While SR can substantially improve robustness, it requires solving a large linear system whose dimension scales with the number of variational parameters, introducing both significant computational overhead and additional numerical stabilization issues (regularization and iterative solution procedures). Recent work attempts to alleviate this bottleneck within the SR framework itself, for instance via the minSR and kernel-trick reformulation (Sec. 2.3.7), which reduces the core linear-algebra problem to matrices whose dimension is set by the Monte Carlo batch size rather than by N_θ [34, 35].

The supervised imaginary time evolution (SITE) approach, developed in this thesis, addresses the same practical limitation from a different angle. Instead of minimizing the energy directly, SITE constructs a *target* wave function by applying an approximate imaginary-time step to the current variational state,

and then updates the network parameters by maximizing the overlap with this target in a supervised optimization loop. By avoiding explicit metric-tensor estimation and the solution of linear systems, this reframing retains a physically meaningful update direction while enabling stable first-order optimization at large parameter counts typical of modern deep learning models.

This general strategy is related to earlier work in which approximate imaginary-time evolution is recast as a supervised regression problem for the wave function [101–103]. Relative to that line of work, the distinctive ingredients of SITE are the use of an overlap-based loss rather than a more naive mean-square regression loss, an adaptive imaginary-time step chosen from low-order energy moments, and a fixed-target training loop that updates the target only after clear energy improvement. The overlap-loss choice is also consistent with related work on neural-network real-time evolution, where overlap-based objectives were found to behave more naturally than direct amplitude regression [104–106].

This chapter is structured as follows. Sec. 3.2 introduces the SITE method: the imaginary-time target construction, the overlap loss and its gradient estimator, the adaptive time-step selection, the relation to direct energy minimization, and the full training algorithm. Sec. 3.3 describes the numerical setup used to validate SITE, covering the MLP-based NQS architecture, MCMC sampling, implementation details, and the J_1–J_2 Heisenberg model benchmark, followed by comparisons with direct energy minimization and DMRG across system sizes. Sec. 3.4 summarizes the main findings and outlines directions for future work.

3.2 SITE method description

We consider a variational wave function $|\psi(\boldsymbol{\theta})\rangle$ in the product basis $|\boldsymbol{s}\rangle$, represented by a NQS (Sec. 2.3). Our goal is to approximate ground states by iterating discrete imaginary-time steps (Sec. 2.3.5), but without solving the SR linear system. SITE does so by converting each (approximate) imaginary-time step into a supervised learning problem: given a *fixed* target state $|\psi_T\rangle$ produced from a frozen copy of the current NQS, we update $\boldsymbol{\theta}$ by maximizing the overlap between $|\psi(\boldsymbol{\theta})\rangle$ and $|\psi_T\rangle$. The resulting update, and the geometry behind it, are summarized in Fig. 3.1.

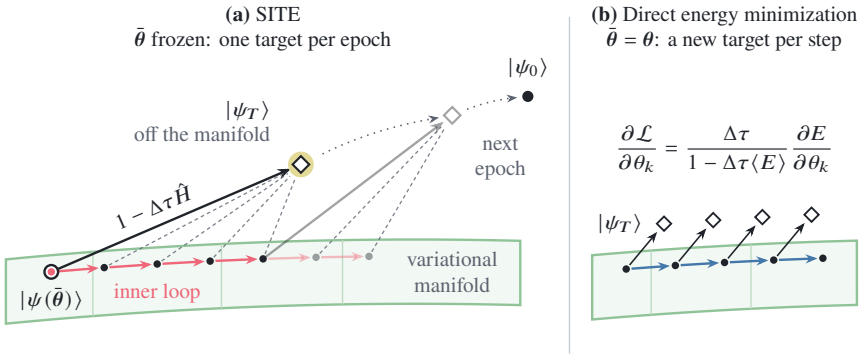

Figure 3.1: The SITE update seen in Hilbert space. **(a)** One epoch. The parameters are frozen at $\bar{\boldsymbol{\theta}}$, and the Euler step of Eq. (3.1) carries the state off the variational manifold to the target $|\psi_T\rangle$, which in general no choice of parameters reproduces. Minimizing the overlap loss of Eq. (3.3) then moves the state along the manifold toward that one target, so the dashed lines, which measure how far the state still is from it, shorten at every step of the inner loop. The epoch ends once the running energy falls below E_{thr}, in general before the closest reachable state is attained; the next epoch freezes the state reached and aims one step further ahead, and the successive targets advance along the imaginary-time flow toward the ground state $|\psi_0\rangle$. **(b)** The limit $\bar{\boldsymbol{\theta}} = \boldsymbol{\theta}$, in which the target is rebuilt from the current state after every update. The first step of an epoch in (a) is of this kind, and is the energy gradient up to the scalar factor of Eq. (3.13) shown in the panel, so what separates SITE from direct energy minimization is the steps that follow, taken against a target that no longer moves. The two schemes are compared on the J_1–J_2 model in Fig. 3.5(a).

3.2.1 Target construction by a discrete imaginary-time step

At the beginning of an epoch, we copy the current parameters into a frozen set $\bar{\boldsymbol{\theta}}$ and construct the target by a single Euler step,

$$|\psi_T\rangle = \left(1 - \Delta\tau\, \hat{H}\right) |\psi(\bar{\boldsymbol{\theta}})\rangle. \tag{3.1}$$

In the product basis this corresponds to

$$\psi_T(\boldsymbol{s}) = \psi(\boldsymbol{s}; \bar{\boldsymbol{\theta}}) - \Delta\tau \sum_{\boldsymbol{s}'} H_{\boldsymbol{s}\boldsymbol{s}'}\, \psi(\boldsymbol{s}'; \bar{\boldsymbol{\theta}}), \qquad H_{\boldsymbol{s}\boldsymbol{s}'} = \langle \boldsymbol{s}|\hat{H}|\boldsymbol{s}'\rangle. \tag{3.2}$$

For local Hamiltonians, $H_{\boldsymbol{s}\boldsymbol{s}'}$ is sparse, so Eq. (3.2) can be evaluated efficiently on a Monte Carlo batch of configurations in the same way as local estimators

in VMC (Sec. 2.3.2). In practice we use only a *single* Euler step to define the target. Applying a second explicit imaginary-time step would require acting with the Hamiltonian on the already transformed state and would therefore increase the computational cost from linear to quadratic in system size for local Hamiltonians. Because an NQS cannot fit the target exactly, the resulting evolution is only approximate and carries optimization noise. Nevertheless, this type of learned imaginary-time evolution admits convergence guarantees under suitable assumptions [102].

3.2.2 Overlap loss and gradient estimator

We aim to adjust the parameters $\boldsymbol{\theta}$ to reduce the distance between the target state $|\psi_T\rangle$ and the variational NQS state $|\psi(\boldsymbol{\theta})\rangle$. We experimented with several loss functions and found that minimizing the following negative log-overlap loss performs well in practice:

$$\mathcal{L}(\boldsymbol{\theta}) = -\log\left[\frac{|\langle\psi(\boldsymbol{\theta})|\psi_T\rangle|^2}{\langle\psi(\boldsymbol{\theta})|\psi(\boldsymbol{\theta})\rangle\,\langle\psi_T|\psi_T\rangle}\right]. \tag{3.3}$$

Similarly to the energy gradient, the gradient of this loss can be estimated from a batch of samples $\boldsymbol{s} \sim |\psi(\boldsymbol{s};\boldsymbol{\theta})|^2$ using the logarithmic derivatives $O_k(\boldsymbol{s}) = \partial_{\theta_k} \log\psi(\boldsymbol{s};\boldsymbol{\theta})$ introduced in Sec. 2.3.4 and writing $|\psi\rangle = |\psi(\boldsymbol{\theta})\rangle$:

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial\theta_k} &= \frac{\partial}{\partial\theta_k}\log\langle\psi|\psi\rangle - \frac{\partial}{\partial\theta_k}\log|\langle\psi|\psi_T\rangle|^2 \\ &= 2\,\mathrm{Re}\left\{\left\langle O_k^*(\boldsymbol{s})\right\rangle - \frac{\left\langle R(\boldsymbol{s})\,O_k^*(\boldsymbol{s})\right\rangle}{\langle R(\boldsymbol{s})\rangle}\right\}\end{aligned} \tag{3.4}$$

where $R(\boldsymbol{s}) = \psi_T(\boldsymbol{s})/\psi(\boldsymbol{s};\boldsymbol{\theta})$ and we have used the following intermediate results:

$$
\begin{aligned}
-\frac{\partial}{\partial\theta_k}\log|\langle\psi|\psi_T\rangle|^2 &= -2\,\mathrm{Re}\left\{\frac{\partial}{\partial\theta_k}\log\langle\psi|\psi_T\rangle\right\} \\
&= -2\,\mathrm{Re}\left\{\frac{1}{\sum_{\boldsymbol{s}'}\psi^*(\boldsymbol{s}')\psi_T(\boldsymbol{s}')}\sum_{\boldsymbol{s}}\psi_T(\boldsymbol{s})\frac{\partial}{\partial\theta_k}\psi^*(\boldsymbol{s})\right\} \\
&= -2\,\mathrm{Re}\left\{\frac{1}{\sum_{\boldsymbol{s}'}|\psi(\boldsymbol{s}')|^2\cdot\frac{\psi_T(\boldsymbol{s}')}{\psi(\boldsymbol{s}')}}\right. \\
&\qquad\left.\times\sum_{\boldsymbol{s}}|\psi(\boldsymbol{s})|^2\frac{\psi_T(\boldsymbol{s})}{\psi(\boldsymbol{s})}\frac{\partial}{\partial\theta_k}\log\psi^*(\boldsymbol{s})\right\} \\
&= -2\,\mathrm{Re}\left\{\frac{1}{\left\langle\frac{\psi_T(\boldsymbol{s})}{\psi(\boldsymbol{s})}\right\rangle}\left\langle\frac{\psi_T(\boldsymbol{s})}{\psi(\boldsymbol{s})}\frac{\partial}{\partial\theta_k}\log\psi^*(\boldsymbol{s})\right\rangle\right\},
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial}{\partial\theta_k}\log\langle\psi|\psi\rangle &= \frac{1}{\langle\psi|\psi\rangle}\frac{\partial}{\partial\theta_k}\sum_{\boldsymbol{s}}\psi(\boldsymbol{s})\psi^*(\boldsymbol{s}) \\
&= 2\,\mathrm{Re}\left\{\frac{1}{\langle\psi|\psi\rangle}\sum_{\boldsymbol{s}}\psi(\boldsymbol{s})\frac{\partial}{\partial\theta_k}\psi^*(\boldsymbol{s})\right\} \\
&= 2\,\mathrm{Re}\left\{\frac{1}{\langle\psi|\psi\rangle}\sum_{\boldsymbol{s}}|\psi(\boldsymbol{s})|^2\frac{\partial}{\partial\theta_k}\log\psi^*(\boldsymbol{s})\right\} \\
&= 2\,\mathrm{Re}\left\{\left\langle\frac{\partial}{\partial\theta_k}\log\psi^*(\boldsymbol{s})\right\rangle\right\}.
\end{aligned}
$$

In practice, $O_k(\boldsymbol{s})$ is obtained by automatic differentiation of the NQS, and the Monte Carlo averages are replaced by empirical means over a finite batch. At an intuitive level, minimizing this loss aligns the current state with a target produced by an imaginary-time step. Since imaginary-time evolution relatively amplifies lower-energy components and suppresses higher-energy ones, repeatedly fitting such targets pushes the variational state toward lower energy. This effect becomes meaningful only when the target is held fixed long enough for the optimizer to make noticeable progress toward it.

3.2.3 Adaptive imaginary-time step

Within the Euler approximation, the expected energy of the target state in Eq. (3.1) can be written in terms of energy moments of the frozen state $|\psi(\bar{\boldsymbol{\theta}})\rangle$:

$$\langle E_T \rangle = \frac{\langle \psi_T | \hat{H} | \psi_T \rangle}{\langle \psi_T | \psi_T \rangle} = \frac{\langle E \rangle - 2\Delta\tau \langle E^2 \rangle + \Delta\tau^2 \langle E^3 \rangle}{1 - 2\Delta\tau \langle E \rangle + \Delta\tau^2 \langle E^2 \rangle}, \tag{3.5}$$

where $\langle E^n \rangle \equiv \langle \psi(\bar{\boldsymbol{\theta}}) | \hat{H}^n | \psi(\bar{\boldsymbol{\theta}}) \rangle / \langle \psi(\bar{\boldsymbol{\theta}}) | \psi(\bar{\boldsymbol{\theta}}) \rangle$. Minimizing Eq. (3.5) with respect to $\Delta\tau$ yields a quadratic equation with the following solutions

$$\Delta\tau = \frac{B \pm \sqrt{B^2 + 4A\sigma^2}}{2A}, \tag{3.6}$$

with

$$A = \langle E^2 \rangle^2 - \langle E \rangle \langle E^3 \rangle, \qquad B = \langle E \rangle \langle E^2 \rangle - \langle E^3 \rangle, \qquad \sigma^2 = \langle E^2 \rangle - \langle E \rangle^2. \tag{3.7}$$

Here σ^2 is the energy variance of the frozen state, i.e. $\mathrm{var}(\hat{H})$ in the notation of Sec. 2.3.5. In SITE we select the root in Eq. (3.6) that yields a positive step and the smaller value of $\langle E_T \rangle$. This significantly accelerates convergence to the ground state compared with using a fixed, manually set time step.

The required moments can be estimated from the same Monte Carlo sampling machinery as in VMC (Sec. 2.3.2). In particular,

$$\langle E \rangle = \langle E_{\mathrm{loc}}(\boldsymbol{s}) \rangle, \qquad \langle E^2 \rangle = \left\langle |E_{\mathrm{loc}}(\boldsymbol{s})|^2 \right\rangle, \tag{3.8}$$

where E_{loc} is the local energy defined in Sec. 2.3.3. The exact estimator for $\langle E^3 \rangle$ involves the local estimator of $\hat{H}^2$:

$$\langle E^3 \rangle = \left\langle E^*_{\mathrm{loc}}(\boldsymbol{s}) \left(E^2\right)_{\mathrm{loc}}(\boldsymbol{s}) \right\rangle, \tag{3.9}$$

and therefore is substantially more expensive for local Hamiltonians due to nested sums over nonzero matrix elements:

$$\begin{aligned} \left(E^2\right)_{\mathrm{loc}}(\boldsymbol{s}) &= \sum_{\boldsymbol{s}'} \frac{\psi(\boldsymbol{s}')}{\psi(\boldsymbol{s})} \sum_{\boldsymbol{s}''} H_{\boldsymbol{s}\boldsymbol{s}''} H_{\boldsymbol{s}''\boldsymbol{s}'} \\ &= \frac{1}{\psi(\boldsymbol{s})} \sum_{\boldsymbol{s}''} H_{\boldsymbol{s}\boldsymbol{s}''} \sum_{\boldsymbol{s}'} \psi(\boldsymbol{s}') H_{\boldsymbol{s}''\boldsymbol{s}'}. \end{aligned} \tag{3.10}$$

This means that the computation of $\langle E^3 \rangle$ quickly becomes impractical with in-

creasing system size. However, we found that using the following approximation usually works well enough in practice:

$$\langle E^3 \rangle \approx \left\langle E_{\mathrm{loc}}(\boldsymbol{s};\bar{\boldsymbol{\theta}}) \left| E_{\mathrm{loc}}(\boldsymbol{s};\bar{\boldsymbol{\theta}}) \right|^2 \right\rangle , \tag{3.11}$$

which becomes exact when ψ is an eigenstate and is expected to be a good approximation when the energy variance is small. Using this approximation preserves the linear scaling in system size of all required moment estimates.

3.2.4 Relation to energy minimization

Immediately after the target has been constructed, i.e. for $\bar{\boldsymbol{\theta}} = \boldsymbol{\theta}$, the ratio between the target wave function in Eq. (3.1) and the current wave function is given by the local energy estimator:

$$\begin{aligned} \frac{\psi_T(\boldsymbol{s})}{\psi(\boldsymbol{s})} &= \frac{\psi(\boldsymbol{s}) - \Delta\tau \sum_{\boldsymbol{s}'} H_{\boldsymbol{s}\boldsymbol{s}'} \psi(\boldsymbol{s}')}{\psi(\boldsymbol{s})} \\ &= 1 - \Delta\tau \sum_{\boldsymbol{s}'} \frac{\psi(\boldsymbol{s}')}{\psi(\boldsymbol{s})} H_{\boldsymbol{s}\boldsymbol{s}'} \\ &= 1 - \Delta\tau E_{\mathrm{loc}}(\boldsymbol{s}) \,. \end{aligned} \tag{3.12}$$

By substituting this ratio into Eq. (3.4), we find that the overlap-loss gradient is proportional to the energy gradient:

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \theta_k} &= 2\,\mathrm{Re}\left\{ \left\langle O_k^*(\boldsymbol{s}) \right\rangle - \frac{\left\langle (1 - \Delta\tau E_{\mathrm{loc}}(\boldsymbol{s}))\, O_k^*(\boldsymbol{s}) \right\rangle}{\langle 1 - \Delta\tau E_{\mathrm{loc}}(\boldsymbol{s}) \rangle} \right\} \\ &= 2\,\mathrm{Re}\left\{ \frac{\Delta\tau \left\langle E_{\mathrm{loc}}(\boldsymbol{s}) O_k^*(\boldsymbol{s}) \right\rangle - \Delta\tau \langle E \rangle \left\langle O_k^*(\boldsymbol{s}) \right\rangle}{1 - \Delta\tau \langle E \rangle} \right\} \\ &= \frac{\Delta\tau}{1 - \Delta\tau \langle E \rangle} \frac{\partial E}{\partial \theta_k} \,. \end{aligned} \tag{3.13}$$

Thus, if the target $|\psi_T\rangle$ in Eq. (3.1) is recomputed after every optimizer step using the *current* parameters $\boldsymbol{\theta}$, SITE becomes equivalent to standard energy minimization up to a rescaling of the step size. However, SITE departs from this limit by holding the frozen parameters $\bar{\boldsymbol{\theta}}$ fixed over multiple optimization steps, i.e., by fitting a *fixed* target state $|\psi_T\rangle$ over an extended inner loop. This may explain why direct energy minimization is often unstable and why SITE performs better (as shown in Sec. 3.3.6): 1) the overlap between the fixed target and the current NQS can be small, so multiple gradient steps are needed before

meaningful progress is made—a regime where a fixed target provides a stable optimization signal; 2) a target that changes at every step can make stochastic-gradient optimization unstable. This interpretation is also qualitatively similar to the role of fixed targets in deep Q learning, where target stabilization is known to improve optimization robustness [107].

3.2.5 SITE algorithm

There are at least two simple ways to slow the target drift relative to the optimized network. One can use a duplicate network whose parameters follow a moving average of the main network, or one can use a duplicate network that is copied only intermittently and otherwise kept fixed for a number of optimizer steps. Both ideas are familiar from reinforcement-learning practice [108]. In SITE we adopt the second option. It matches the interpretation of the method as a sequence of discrete imaginary-time steps and, in contrast to a moving-average target, it allows the adaptive step size $\Delta\tau$ to be defined unambiguously for each epoch.

Training is organized into epochs, each corresponding to one discrete imaginary-time step $\Delta\tau$ (Fig. 3.2):

1. Estimate the moments $\langle E\rangle$, $\langle E^2\rangle$, $\langle E^3\rangle$ for the current NQS and compute $\Delta\tau$ via Eq. (3.6).

2. Freeze a copy $\bar{\boldsymbol{\theta}} \leftarrow \boldsymbol{\theta}$ and construct the target via Eq. (3.2).

3. Perform stochastic-gradient updates of $\boldsymbol{\theta}$ that minimize the overlap loss function in Eq. (3.3) (e.g. using Adam [83]) until a progress criterion is met.

In this work we use the following progress criterion: the epoch ends once the running energy estimate drops below $E_{\text{thr}} = \langle E\rangle - \sigma_E$, where σ_E is the standard error of the mean energy (the statistical uncertainty of $\langle E\rangle$, not to be confused with the variance σ^2 of Eq. (3.7)) computed at the start of the epoch using a large sample. During the epoch itself, however, recomputing the energy from such a large sample after every optimizer update would be unnecessarily expensive. Instead, we reuse the configurations already generated for gradient estimation, compute noisy batch energies from them, and smooth these values with a running average. This approximate online estimate is used only to decide whether the threshold has been crossed, whereas the threshold value itself is

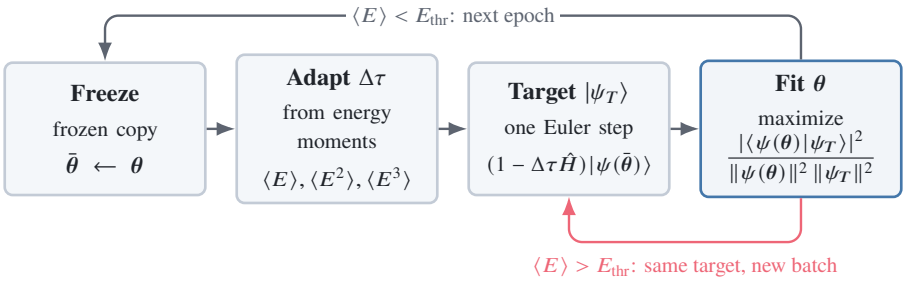

Figure 3.2: One SITE epoch, read from left to right: a specialization of the variational Monte Carlo iteration of Fig. 2.12 in which the optimizer stage is replaced by a supervised fit. No gradient flows through the three grey stages, since the frozen parameters $\bar{\boldsymbol{\theta}}$ and the step $\Delta\tau$ are held constant for the whole epoch; only the fit is trainable. The two loops are what distinguishes SITE from direct energy minimization. The inner one repeats the fit against the same target, on a fresh batch of configurations at every update, for as long as the running energy stays above E_{thr}; the outer one closes an epoch, re-freezing the state reached so that the next target is built from it.

defined from the more accurate large-sample estimate at the start of the epoch. A simplified pseudocode of the training algorithm is provided in Alg. 1.

3.3 Testing SITE with numerical experiments

The SITE update rule described above is largely agnostic to the particular variational family: it only requires the ability to evaluate the wave-function amplitudes $\psi(\boldsymbol{s}; \boldsymbol{\theta})$ and differentiate them with respect to the parameters. To validate the method numerically, we instantiate SITE with a generic MLP-based NQS and use it to approximate the ground state of the J_1–J_2 Heisenberg model on the square lattice (Sec. 2.1.2). This benchmark is intentionally demanding, since frustration induces a nontrivial sign/phase structure and a rugged optimization landscape, with the most challenging regime occurring near the putative phase boundary around $J_2/J_1 \approx 0.5$. Our goal is therefore not to engineer a model-specific architecture, but to test whether SITE can optimize large neural networks reliably without relying on problem-tailored symmetry constraints or SR-style preconditioning.

Algorithm 1 NQS training with SITE (overlap loss of Eq. (3.3))

```
NQS ← randomInitialization()
for N_epochs do
    ⟨E⟩, ⟨E^2⟩, ⟨E^3⟩, σ_E ← energyStatistics(NQS)
    Δτ ← optimalTimeStep(⟨E⟩, ⟨E^2⟩, ⟨E^3⟩)
    E_thr ← ⟨E⟩ − σ_E
    NQS_fixed ← copy(NQS)
    s_mcmc ← warmupMCMC(NQS)          ▷ initial batch of basis states for MCMC sampling
    while ⟨E⟩ > E_thr do
        s_batch, s_mcmc ← sampleMCMC(NQS, s_mcmc)
        ψ_target ← computeTarget(NQS_fixed, s_batch, Δτ)
        grad ← lossGradient(NQS, ψ_target, s_batch)
        NQS ← updateParameters(NQS, grad)
        ⟨E⟩ ← movingAverageEnergy(NQS, s_batch)
    end while
end for
```

3.3.1 Neural network architecture

The existing literature on NQS encompasses a wide range of neural network architectures, including convolutional networks, recurrent networks, and transformer-based autoregressive models (see e.g. the review [31] and references therein). In this work we employ MLPs (Sec. 2.2.1) because in preliminary experiments they offered a practical balance between performance, implementation simplicity, and computational cost. We emphasize that we do not claim MLPs to be optimal for NQS in general. Rather, the choice aligns with the methodological focus of this chapter, namely the optimization scheme.

Input encoding. For a spin-$\frac{1}{2}$ model on an $L \times L$ lattice, where L is the linear lattice size and $N = L^2$ the number of sites, a convenient representation of a basis configuration $\boldsymbol{s}$ is a matrix with entries in $\{-1, 1\}$ obtained from the local spin states, e.g. $|\uparrow\rangle \mapsto 1$ and $|\downarrow\rangle \mapsto -1$. A straightforward approach is to flatten this matrix and feed it directly to the first MLP layer. We found that a more effective encoding can be obtained by using a patch-based embedding inspired by vision-transformer preprocessing [109]. Specifically, the lattice is partitioned into non-overlapping patches of size $d_\mathrm{p} \times d_\mathrm{p}$. Each patch is flattened and mapped to an embedding vector in $\mathbb{R}^{d_\mathrm{enc}}$ by a trainable linear transformation. The resulting sequence of patch embeddings is concatenated and serves as the

input to the MLP. This representation preserves local spatial structure at the level of the embedding and can be seen as a lightweight alternative to explicit convolutional layers.

Output parameterization. As explained in Sec. 2.3, there are two common strategies for defining the output of an NQS. One can either predict the (unnormalized) complex amplitude $\psi(\boldsymbol{s})$ directly, or factorize the wave function into conditionals and generate it autoregressively. In this chapter we use the first approach. Concretely, the final layer produces two real outputs, which we interpret as a log-amplitude and a phase,

$$\psi(\boldsymbol{s};\boldsymbol{\theta}) = \exp\Big(\ell(\boldsymbol{s};\boldsymbol{\theta}) + \mathrm{i}\,\varphi(\boldsymbol{s};\boldsymbol{\theta})\Big). \tag{3.14}$$

Because physical observables depend only on relative amplitudes and phases, the log-amplitude ℓ can drift to very large magnitudes during training. To stabilize optimization we follow Ref. [100] and apply a saturating nonlinearity to the raw log-amplitude output,

$$\ell(\boldsymbol{s};\boldsymbol{\theta}) = a\,\tanh\left(\frac{x(\boldsymbol{s};\boldsymbol{\theta})}{a}\right), \tag{3.15}$$

with $a = 20$, which still allows amplitudes spanning many orders of magnitude. For the phase we apply no nonlinearity and allow values outside $[0, 2\pi)$. Because φ enters only through $\mathrm{e}^{\mathrm{i}\varphi}$, which is 2π-periodic, a value outside this interval carries no extra physical content: φ and $\varphi + 2\pi k$ give the same amplitude, so an unconstrained output simply represents each phase by any of its equivalent values. Forcing the output into a fixed interval of length 2π instead, e.g. by a bounded nonlinearity or by reduction modulo 2π, would introduce an artificial boundary: phases just below and just above the wrap point are physically almost identical, yet would have to be represented by network outputs differing by almost 2π, so that a smooth phase structure would require a discontinuity of the output across nearby configurations. Leaving φ unconstrained avoids this and lets the network adjust the phase continuously, with the periodicity of $\mathrm{e}^{\mathrm{i}\varphi}$ taking care of the wrapping. All variational parameters are real. The architecture is illustrated in Fig. 3.3.

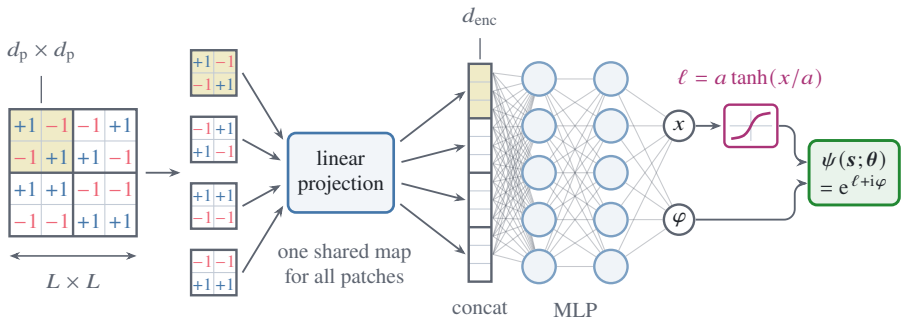

Figure 3.3: The NQS architecture used in the SITE numerical experiments. The configuration enters as an $L \times L$ array of ± 1 values in the spin colours used throughout (blue $+1$ for $|\uparrow\rangle$, red -1 for $|\downarrow\rangle$), here with zero total magnetization as in the fixed-magnetization sector of Sec. 3.3.2. The array is partitioned into $d_p \times d_p$ patches, every patch is flattened and mapped by the *same* trainable linear projection to a d_{enc}-dimensional embedding, and the embeddings are concatenated in patch order into the input of the MLP; the highlighted patch can be followed to its segment of the concatenated vector. The two real outputs are the raw log-amplitude x, which passes through the saturating nonlinearity $\ell = a \tanh(x/a)$, and the phase φ, which is left unconstrained; together they define the complex amplitude $\psi(\boldsymbol{s};\boldsymbol{\theta}) = \exp(\ell + i\varphi)$.

3.3.2 Sampling of basis states

SITE requires Monte Carlo averages with respect to $|\psi(\boldsymbol{s};\boldsymbol{\theta})|^2$ both for the loss gradient in Sec. 3.2.2 and for the energy-moment estimates used to select $\Delta\tau$ in Sec. 3.2.3. Since the NQS represents an unnormalized wave function in a Hilbert space of exponential dimension, direct sampling is infeasible. We therefore sample configurations using MCMC, specifically the Metropolis–Hastings algorithm (see Sec. 2.3.8).

For the spin models considered below we work in a sector of fixed total magnetization, i.e. a fixed number of up spins (Sec. 2.1.2), which the J_1–J_2 Hamiltonian conserves. We initialize each Markov chain at a uniformly random configuration $\boldsymbol{s}_0$ compatible with the magnetization constraint. At MCMC step m we propose a candidate $\boldsymbol{s}'_m$ by selecting a random nearest-neighbor bond and swapping the spins at its endpoints, which preserves the total magnetization. To reduce dependence on the initial state, we discard an initial warm-up segment of $N_{\text{warmup}} = 10\,L^2 = 10\,N$ proposed moves. To mitigate autocorrelations we keep only every $N_{\text{skip}} = 4$-th sample.

To increase throughput on graphics processing units we parallelize sampling by running many independent chains in parallel. In our implementation we choose the number of chains equal to the batch size M (the number of configurations used per stochastic-gradient step), so that one MCMC sweep produces exactly one configuration batch for the stochastic-gradient step. Moreover, during the inner optimization loop of each SITE epoch we reuse the final states of all chains as the starting point for the next sampling call. This persistence reduces equilibration overhead and makes the cost of sampling nearly independent of the number of optimizer steps taken within an epoch.

3.3.3 Implementation

The SITE algorithm is implemented in the in-house JAX-based NQS library described in Sec. 2.4. SR baselines are obtained with NetKet [95, 96], and the ED and DMRG reference results use QuSpin [97, 98] and TeNPy [99], respectively.

3.3.4 Symmetries of J_1–J_2 Heisenberg model

We consider the two-dimensional J_1–J_2 Heisenberg model on a square lattice (Sec. 2.1.2). Throughout the SITE experiments, the lattice is taken with periodic boundary conditions. The Hamiltonian has several symmetries (translations,

point-group symmetries, spin inversion, and global SU(2) spin rotation), which can be exploited to reduce the effective complexity of the optimization problem (Sec. 2.3.10). In the present chapter we intentionally do *not* enforce symmetry invariance in the NQS ansatz. This choice makes the optimization problem harder, but it also tests the central claim of SITE: that stable first-order optimization of large neural networks is possible without specialized architectures or explicit symmetry projections.

3.3.5 Hyperparameters and model selection

Unless stated otherwise, we use the hyperparameters summarized in Table 3.1. Throughout this section we optimize the NQS parameters using the Adam optimizer with an exponentially decaying learning-rate schedule, starting from η_0 and ending at η_f. In addition to the stochastic batches used for gradient updates, we periodically estimate energies from larger Monte Carlo samples to obtain reliable error bars and to evaluate the SITE progress criterion described in Sec. 3.2.5.

Hyperparameter	**symbol**	**value**
Input patch size	d_p	2
Patch encoding size	d_{enc}	8
Training batch size	M	256
Initial learning rate	η_0	10^{-3}
Final learning rate	η_f	10^{-5}
Number of samples for energy statistics during training	–	10^5
Number of samples for final energy estimation	–	10^6
Total number of optimizer steps	–	$5 \cdot 10^5$
MCMC warm-up steps	N_{warmup}	$10L^2$
MCMC sample skip interval	N_{skip}	4

Table 3.1: Hyperparameters used in the SITE numerical experiments.

Width and depth. To choose the network width (neurons per hidden layer) and depth (number of hidden layers), we performed a small grid search on a 6×6 lattice at $J_2/J_1 = 0.5$. We explored three widths (128, 256, 512) and depths from 1 to 5, and we measured performance by the energy error per site relative to ED, $(E - E_{ED})/N$. The results are shown in Fig. 3.4. As expected, increasing width generally improves accuracy, consistent with the idea that larger networks

can represent a larger effective variational subspace. The improvement does not clearly saturate at width 512, suggesting that even larger models could yield further gains, at the cost of additional computation. In contrast, increasing depth beyond 4 degrades performance in our setup, which is consistent with the general difficulty of training deep MLPs without additional techniques such as residual connections or carefully designed initializations. Based on these results, the subsequent experiments use depth 4 and width 512. This configuration has 893 994 variational parameters.

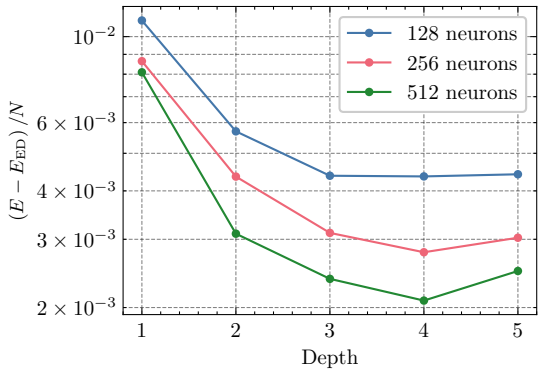

Figure 3.4: Energy error per site, $(E - E_{\mathrm{ED}})/N$, as a function of neural network width and depth for NQS trained with the SITE overlap loss. The lattice consists of 6×6 sites with $J_2/J_1 = 0.5$. Increasing the width consistently lowers the error, while increasing the depth helps up to about four hidden layers and degrades performance at depth five in this setup.

3.3.6 Comparison with direct energy minimization

We now compare SITE training to a more standard approach where the variational parameters are optimized by minimizing the energy directly. Specifically, we compare three methods on a 6×6 lattice: *(i)* direct energy minimization with Adam ("E loss"), *(ii)* SITE training with the overlap loss ("ITE loss"), and *(iii)* energy minimization with SR. Unless stated otherwise, all shared hyperparameters are kept fixed. For SR we employ vanilla stochastic gradient descent instead of Adam and increase the learning rate by a factor of 10. Because SR iterations are substantially more expensive (in our experiments by a factor of about 4–6 due to the iterative linear solver), we reduce the total number of optimizer steps for SR to 10^5. Energy errors are computed relative to ED.

Figure 3.5(a) shows representative energy-minimization curves at $J_2/J_1 = 0.5$ for three different random initializations of the NQS parameters. Direct energy minimization with a vanilla first-order optimizer is highly unstable: runs exhibit sudden jumps in energy and converge to noticeably different final values depending on initialization. In contrast, SITE training with the ITE loss produces smooth curves with essentially no catastrophic jumps, and the final energies show only weak dependence on initialization. Moreover, in this regime SITE consistently reaches lower energies than the E-loss baseline. This behavior aligns with the analysis in Sec. 3.2.4: using the energy as a loss corresponds to recomputing the SITE target after every update, i.e. optimizing against a target that changes at each step, which can destabilize stochastic-gradient optimization.

Figure 3.5(b) shows the energy error as a function of J_2/J_1. Because E-loss runs vary substantially, we performed 5 runs for each coupling ratio and report only the best result. SITE yields lower errors than direct energy minimization over a broad range of parameters, not only at the maximally frustrated point. Compared with SR, SITE is competitive: SR is sometimes slightly better and sometimes slightly worse, but the overall accuracy is comparable within the variability induced by sampling and finite optimization time. All methods exhibit a clear trend of larger errors near $J_2/J_1 = 0.5$, where frustration makes the variational landscape harder.

In terms of computational cost, SITE typically required about 30%–50% more wall-clock time than E-loss training with the same number of optimizer steps. The main overhead is the additional energy estimation used to decide when to update the frozen target within each epoch. Compared to SR, SITE iterations were on average about five times faster in our implementation, while achieving a similar final accuracy at comparable total compute budgets.

3.3.7 Scaling and benchmarking against DMRG

To go beyond system sizes accessible to ED, we compare SITE-trained NQS to DMRG by studying how the variational energy scales with lattice size at fixed frustration $J_2/J_1 = 0.5$. Since the performance of DMRG depends strongly on the bond dimension χ, we report results for $\chi = 128$ and $\chi = 1024$. For the NQS we also investigate the effect of longer optimization: in addition to the default training budget of $5 \cdot 10^5$ optimizer steps, we report results obtained with $5 \cdot 10^6$ steps. As shown in Fig. 3.6, extended training yields a small but systematic improvement in the final energy, and this improvement becomes more pronounced for larger lattices.

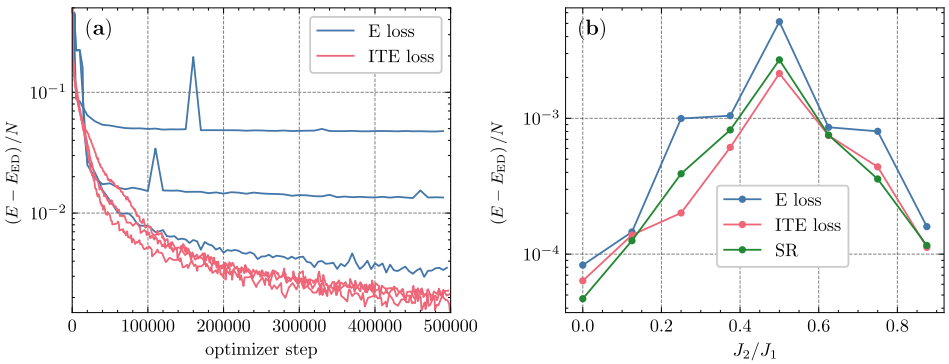

Figure 3.5: **(a)** Energy minimization curves, shown as the energy error per site $(E - E_{\mathrm{ED}})/N$, for NQS trained with direct energy minimization (E loss, blue) and SITE overlap loss (ITE loss, red). Each method is shown for three random initializations of the NQS parameters at $J_2/J_1 = 0.5$. The E-loss runs show large seed-to-seed variation and occasional upward jumps, whereas the SITE curves decrease more smoothly and end at lower errors. **(b)** Energy error per site, $(E - E_{\mathrm{ED}})/N$, as a function of J_2/J_1 for E loss (blue), ITE loss (red), and SR (green), on a 6×6 lattice. All three methods become least accurate near the maximally frustrated region, but SITE remains competitive with SR and improves over the E-loss baseline across most of the sweep.

For the considered sizes, the NQS clearly outperforms DMRG with $\chi = 128$. For $\chi = 1024$ the comparison is more nuanced. Both approaches reproduce the ED energy essentially exactly at 4×4. At 6×6 and 8×8, the NQS becomes competitive and can slightly outperform DMRG when trained longer. However, for lattice sizes larger than 8×8 DMRG with $\chi = 1024$ yields lower energies, and the gap grows with system size. One plausible explanation is the different scaling of variational capacity with system size. In the matrix-product-state ansatz underlying DMRG, the number of variational parameters grows with the system size (for fixed χ), whereas in the MLP-based NQS used here it remains almost constant: we kept the MLP width and depth fixed across lattice sizes, so only the input layer depends on L, and it constitutes a small fraction of the total parameter count. This suggests that, for the fixed architecture considered here, the NQS becomes effectively more constrained as the lattice grows, which can eventually limit accuracy. Of course, one could also scale the MLP architecture with system size, but that is not the regime studied in this benchmark.

It is difficult to directly compare wall-clock times because our DMRG calculations ran on CPU while NQS training ran on GPU. Nevertheless, for the largest system shown (12×12) the total compute time was of the same order for DMRG with $\chi = 1024$ and for NQS training with $5 \cdot 10^6$ optimizer steps (around 20 hours in our setup).

3.3.8 Energy accuracy limits and comparison to other NQS results

The maximally frustrated regime around $J_2/J_1 = 0.5$ is known to be particularly challenging for NQS optimization. For $J_2/J_1 = 0.5$ on a 6×6 lattice, Ref. [100] reported an apparent "error floor" for NQS energies, with multiple architectures and optimization strategies saturating at relative errors on the order of $2 \cdot 10^{-3}$; consistent results were also found in Refs. [110–112]. In Fig. 3.5(b) we observe a similar level of accuracy at this point with our default training budget. When increasing the training time to $5 \cdot 10^6$ steps we obtain a modest improvement (to about $1.4 \cdot 10^{-3}$), but the gain remains limited despite the large number of parameters in our MLP. This supports the interpretation that, in this regime, the optimization difficulty is dominated by the structure of the variational landscape and the sign/phase complexity of the ground state rather than by a lack of representational power.

Table 3.2 compares several reported energies per site for a 10×10 lattice at $J_2/J_1 = 0.5$. Most literature results exploit translation-invariant convolutional architectures and explicitly incorporate lattice symmetries. In contrast, our

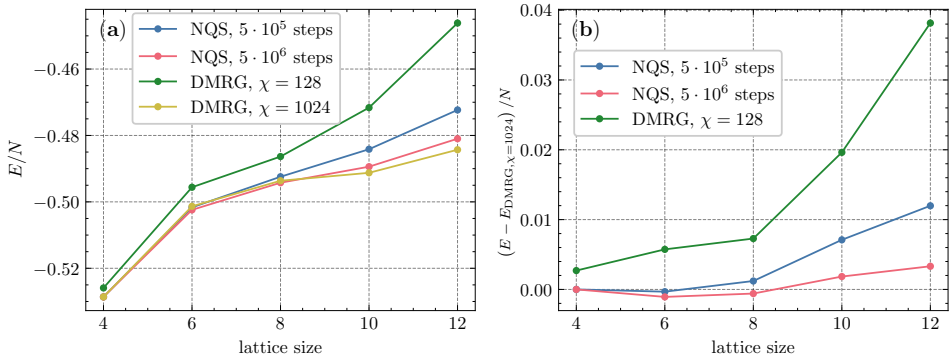

Figure 3.6: **(a)** Variational ground-state energy as a function of lattice size at $J_2/J_1 = 0.5$ for SITE-trained NQS and for DMRG. The blue and red curves correspond to NQS trained for $5 \cdot 10^5$ and $5 \cdot 10^6$ optimizer steps, respectively. The green and yellow curves show DMRG with bond dimensions $\chi = 128$ and $\chi = 1024$. Longer NQS training gives a systematic energy improvement and remains close to the high-bond-dimension DMRG curve up to the intermediate sizes. **(b)** Same data as (a), shown as energy differences per site, $(E - E_{1024})/N$, from the DMRG result E_{1024} at $\chi = 1024$. This representation makes the crossover clearer: for larger lattices DMRG with $\chi = 1024$ pulls ahead, while DMRG with $\chi = 128$ accumulates the largest error.

SITE run uses a generic MLP and flat-metric (Euclidean) gradient updates without symmetry enforcement, yet yields a competitive energy scale. While the best symmetry-aware NQS results remain lower in energy, this comparison highlights that SITE can reliably train large, generic NQS ansätze and reach energies in the same ballpark without extensive problem-specific engineering.

Method	E/N
CNN, REMD [113]	−0.4736
CNN, SR [110]	−0.4952
CNN, SR [114]	−0.4958
CNN, minSR [34]	−0.4976
MLP, SITE (ours)	−0.4894

Table 3.2: Comparison of energy estimates per site for a 10×10 lattice at $J_2/J_1 = 0.5$.

3.4 Conclusions

This chapter introduced SITE, the fixed-target NQS optimization scheme summarized in Statement 1, which is motivated directly by imaginary-time evolution. Instead of minimizing the variational energy directly, SITE constructs a fixed target wave function by applying a discretized imaginary-time step to a frozen copy of the current state, and then updates the variational parameters by maximizing the overlap with this target. An adaptive choice of the time step $\Delta\tau$ based on low-order energy moments accelerates convergence, while keeping the target fixed for multiple gradient steps stabilizes stochastic optimization.

By analyzing the relation between SITE and direct energy minimization, we also obtain a simple interpretation of why naive energy-loss training can be unstable: it corresponds to the limiting case of SITE where the target is recomputed after every optimizer step and therefore constantly shifts. In the numerical experiments on the J_1–J_2 model we found that SITE yields substantially smoother and more reproducible optimization trajectories than direct energy minimization with a first-order optimizer, and that it reaches lower energies for comparable training budgets. Compared with SR, SITE achieves competitive accuracy while avoiding the expensive solution of linear systems that dominates SR runtimes at large parameter counts.

We stress that these experiments were intentionally performed without

enforcing the symmetries of the Hamiltonian in the NQS ansatz. This design choice makes the benchmark more challenging, but it also highlights a key advantage of SITE: it enables stable training of large, generic neural networks using standard first-order optimization tools. In practical applications, SITE can be combined with symmetry-aware architectures and projections (Sec. 2.3.10), and exploring such combinations is a natural direction for further improving accuracy in highly frustrated regimes.

4 Universal performance gap in HBHM

This chapter is based on the results published in [P2].

4.1 Introduction

As discussed in Chapter 2, NQS provide a flexible variational representation of many-body wave functions and can be trained efficiently within the VMC framework. Despite the broad practical success of this approach, the accuracy achieved by a given NQS architecture can vary dramatically between physical models and, crucially, between parameter regimes of the *same* model. Understanding when and why NQS fail is therefore as important as demonstrating new successes: it identifies the limitations of standard architectures and optimizers, and it provides concrete targets for developing improved inductive biases and training strategies.

Previous work has reported sharp degradations of NQS performance in intrinsically complex settings, for example in the frustrated Heisenberg J_1–J_2 model near $J_2/J_1 \simeq 0.5$ [100, 115]. A natural hypothesis in such cases is that optimization becomes unreliable due to a rugged variational landscape, where stochastic-gradient methods can become trapped in plateaus, saddle points, or poor local minima. This perspective also motivates algorithmic developments that primarily aim to improve *trainability*, such as the optimization methods discussed in Chapter 3.

In this chapter we investigate a complementary failure mode: a regime in which standard, "unspecialized" NQS architectures exhibit a pronounced and seemingly universal *performance gap* that is not easily attributed to loss-landscape ruggedness. Figure 4.1 contrasts the two failure modes: in one, good variational states exist but a rugged landscape keeps the optimizer from finding them; in the other, optimization succeeds, yet the best state the ansatz can represent is itself a poor approximation. Our testbed is the Hofstadter–Bose–

Hubbard model (Sec. 2.1.3), and we focus on its hard-core limit with open boundary conditions. This study is complementary to recent work that applied recurrent neural-network wave functions to interacting Hofstadter models with finite on-site occupation and longer-range interactions [116], where a degradation of performance was likewise observed in selected benchmark observables. A ladder version of the Hofstadter–Bose–Hubbard model was also investigated in Ref. [117]. When the magnetic length becomes comparable to the lattice period, competing length scales can induce frustration and qualitatively change the structure of the ground state, making this model a stringent benchmark for variational wave functions.

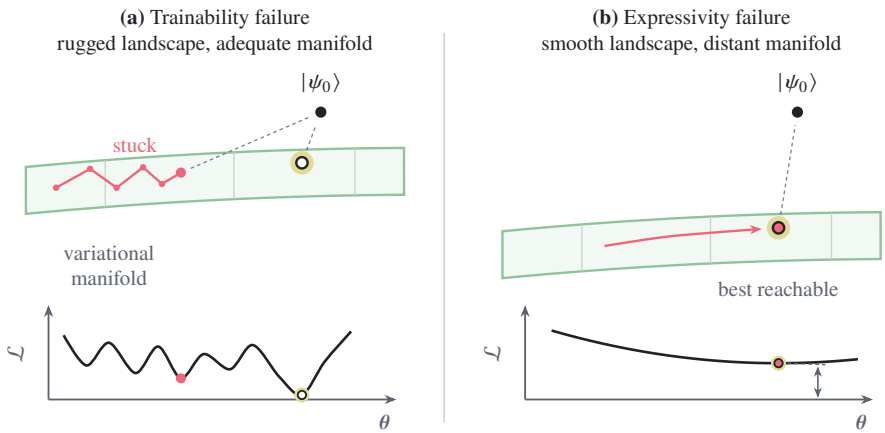

Figure 4.1: Two ways a variational calculation can fail. In both panels the green sheet is the variational manifold (cf. Fig. 2.13), the yellow marker is the best approximation to the ground state $|\psi_0\rangle$ that the manifold holds, the red trajectory is an optimization run ending in the red dot, and the dashed lines measure distances to $|\psi_0\rangle$; below each sheet, the corresponding loss $\mathcal{L}$ along a direction in parameter space. **(a)** Trainability failure: the manifold passes close to $|\psi_0\rangle$, but the landscape is rugged and the optimizer stalls in a poor local minimum, far from the good state it never finds; this is the failure mode reported for frustrated spin models [100, 115]. **(b)** Expressivity failure: the landscape is a single smooth basin and its minimum is found, but the manifold itself passes far from $|\psi_0\rangle$, so the loss of the best reachable state stays well above zero (double arrow). The diagnostics of Sec. 4.4.3 place the strong-flux HBHM in this second class.

We show that increasing the magnetic flux leads to a sharp drop in NQS accuracy: even for very small lattices, the variational energy error can differ by

orders of magnitude between low- and high-flux regimes. Moreover, at fixed flux we find a simple scaling relation between performance and the ratio of the number of variational parameters to the Hilbert-space dimension, which holds across multilayer perceptron and convolutional architectures and for different system sizes and particle densities. The universality of this trend suggests that the magnetic flux introduces features that are systematically difficult for these generic ansätze. To better characterize this difficulty, we perform a series of numerical tests of plausible explanations; strikingly, we find no evidence that the observed gap is primarily caused by ruggedness of the loss landscape. The HBHM thus provides a valuable test case for mapping out the limitations of current NQS methods and for guiding the development of architectures and optimization schemes tailored to magnetic and frustrated lattice systems.

This chapter is structured as follows. Sec. 4.2 specifies the hard-core Hofstadter–Bose–Hubbard benchmark and the flux convention used in the calculations. Sec. 4.3 describes the neural-network architectures and the three training protocols used to separate optimization effects from representational limitations. Sec. 4.4 presents the numerical evidence for the performance gap and analyzes several possible causes, including wave-function statistics, phase structure, entanglement, loss-landscape geometry, architectural choices, input encodings, curriculum learning, and numerical precision. Finally, Sec. 4.5 summarizes the main conclusions and discusses implications for future NQS benchmarks.

4.2 Hofstadter–Bose–Hubbard model

The Hofstadter model and its interacting Hofstadter–Bose–Hubbard extension were introduced in Sec. 2.1.3. Here we focus on the hard-core limit of the bosonic model on a finite $L_x \times L_y$ square lattice with open boundary conditions. Open boundaries allow us to vary the flux per plaquette α continuously, at the cost of explicitly breaking translational and rotational symmetries.

In the Landau gauge, the Hamiltonian reads

$$\hat{H} = -J \sum_{j=1}^{L_x-1} \sum_{k=1}^{L_y} \left(\hat{b}^\dagger_{j+1,k} \hat{b}_{j,k} + \text{h.c.} \right) - J \sum_{j=1}^{L_x} \sum_{k=1}^{L_y-1} \left(\mathrm{e}^{\mathrm{i}2\pi\alpha j}\, \hat{b}^\dagger_{j,k+1} \hat{b}_{j,k} + \text{h.c.} \right), \tag{4.1}$$

where the sites are labelled by integer coordinates (j, k) as in Eq. (2.20), $\hat{b}^\dagger_{j,k}$ ($\hat{b}_{j,k}$) are bosonic creation (annihilation) operators, and J sets the hopping

energy scale. The Peierls phase in the y-direction hopping implements a uniform magnetic flux through each plaquette; moving counter-clockwise around any elementary square yields a net phase $2\pi\alpha$ (Sec. 2.1.3). Because the spectrum is periodic under $\alpha \to \alpha + 1$ and symmetric under $\alpha \to -\alpha$, it is sufficient to consider $\alpha \in [0, 0.5]$.

Hard-core interactions are enforced as a constraint on the local Hilbert space, restricting the occupation numbers (the eigenvalues of $\hat{n}_{j,k}$) to $n_{j,k} \in \{0, 1\}$, corresponding to the $U \to \infty$ limit of Eq. (2.20). At fixed particle number N_b (we reserve N for the number of sites, here $N = L_x L_y$), the many-body Hilbert space is spanned by computational-basis configurations $|\boldsymbol{s}\rangle$ with exactly N_b occupied sites, for example

$$|\boldsymbol{s}\rangle = \hat{b}^\dagger_{j_1,k_1} \hat{b}^\dagger_{j_2,k_2} \cdots \hat{b}^\dagger_{j_{N_\mathrm{b}},k_{N_\mathrm{b}}} |\mathrm{vac}\rangle, \tag{4.2}$$

where the index pairs (j_ℓ, k_ℓ) are all distinct and ordered according to a fixed convention. The dimension of this hard-core Hilbert space is

$$\dim \mathcal{H} = \binom{L_x L_y}{N_\mathrm{b}}. \tag{4.3}$$

Throughout this chapter we concentrate on small lattices where exact diagonalization is feasible and where additional terms such as longer-range interactions can be neglected without obscuring the main phenomenon of interest: the sharp degradation of NQS performance at strong magnetic flux.

4.3 Methods

4.3.1 Neural-network architectures

We benchmark both MLP (Sec. 2.2.1) and CNN (Sec. 2.2.2) NQS architectures. Each configuration $\boldsymbol{s}$ is encoded by site occupations on the $L_x \times L_y$ lattice and fed either as a vector (MLP) or as a matrix (CNN), with occupancies symmetrized to ± 1 rather than $\{0, 1\}$. For real-valued network weights, the final layer outputs two real numbers interpreted as the real and imaginary parts of $\log \psi_{\boldsymbol{\theta}}(\boldsymbol{s})$. For complex-valued networks, the output is a single complex number representing $\log \psi_{\boldsymbol{\theta}}(\boldsymbol{s})$. In the CNN case, the final feature maps are flattened and passed to a final affine layer (instead of global pooling) in order to avoid imposing translational invariance, which is absent for the open-boundary HBHM.

Unless stated otherwise, each dense or convolutional layer is followed by layer normalization [76] and the GELU nonlinearity [65]. We experimented with a range of alternative encodings and architectural choices, but found that such details do not qualitatively affect the central observations.

Finally, we deliberately keep the number of variational parameters N_θ well below the Hilbert-space dimension $\dim \mathcal{H}$ even for the small lattices studied here. This reflects the intended use case of NQS as a compressed representation; once $N_\theta \gtrsim \dim \mathcal{H}$, one can in principle optimize the full state vector directly, making the neural network unnecessary.

4.3.2 Ground-state approximation with NQS

We use three complementary protocols to approximate or fit ground states, chosen to disentangle optimization effects from representational limitations.

Stochastic reconfiguration

We minimize the variational energy using SR, i.e. a natural-gradient update based on the quantum geometric tensor (Sec. 2.3.6). In practice we use the NetKet implementation of SR (Sec. 2.4) with the Adam optimizer [83], a learning rate $3 \cdot 10^{-4}$, and batch size 512. The SR linear system is stabilized by adding a small diagonal shift (0.01) to the metric tensor estimate.

Supervised imaginary time evolution

We also train the NQS using SITE, introduced in Chapter 3 (in particular Sec. 3.2). SITE replaces direct energy minimization by repeated supervised fitting of a fixed target state obtained from a discretized imaginary-time step. For SITE we use the in-house JAX-based implementation described in Sec. 2.4, with the same optimizer hyperparameters as above (Adam with learning rate $3 \cdot 10^{-4}$ and batch size 512).

Supervised learning of the exact solution

To probe the expressivity of the ansatz independently of the variational principle, we perform direct supervised learning of the exact-diagonalization ground state $|\psi_{\mathrm{ED}}\rangle$. Specifically, we minimize the negative log-overlap loss used in SITE,

but with the fixed target $|\psi_T\rangle$ set to $|\psi_{\mathrm{ED}}\rangle$:

$$\mathcal{L}(\boldsymbol{\theta}) = -\log\left[\frac{|\langle\psi(\boldsymbol{\theta})|\psi_{\mathrm{ED}}\rangle|^2}{\langle\psi(\boldsymbol{\theta})|\psi(\boldsymbol{\theta})\rangle\,\langle\psi_{\mathrm{ED}}|\psi_{\mathrm{ED}}\rangle}\right]. \tag{4.4}$$

This protocol is not meant as a practical ground-state solver (since ED is assumed available), but as a controlled diagnostic: if the ansatz cannot fit $|\psi_{\mathrm{ED}}\rangle$ well under supervised optimization, then variational methods cannot be expected to do better. In this setting we quantify performance by the overlap deviation

$$d(\psi_{\boldsymbol{\theta}}, \psi_{\mathrm{ED}}) = 1 - |\langle\psi(\boldsymbol{\theta})|\psi_{\mathrm{ED}}\rangle|, \tag{4.5}$$

where both $|\psi(\boldsymbol{\theta})\rangle$ and $|\psi_{\mathrm{ED}}\rangle$ are normalized to unity.

4.3.3 State sampling

All three protocols require estimates of expectation values with respect to $|\psi_{\boldsymbol{\theta}}(\boldsymbol{s})|^2$. For large systems this is typically done using Markov-chain Monte Carlo sampling or by employing autoregressive architectures with exact sampling (Sec. 2.3.8 and 2.3.9). Here we instead restrict to small lattices where the full wave function vector can be evaluated explicitly. This allows exact sampling from $|\psi_{\boldsymbol{\theta}}(\boldsymbol{s})|^2$ and, when needed, exact evaluation of losses and derivatives without stochastic noise. As a result, the performance trends reported below are not obscured by sampling artefacts and can be traced more directly to the structure of the target state and to the inductive bias of the chosen NQS architectures.

4.4 Results

4.4.1 Drop of NQS performance in the strong-flux regime

We start by probing how NQS accuracy changes with magnetic flux. As a baseline example we consider a 4×5 lattice with $N_{\mathrm{b}} = 4$ hard-core bosons and an MLP with two hidden layers of width 32. This network has $N_\theta = 1856$ variational parameters, compared with $\dim \mathcal{H} = \binom{20}{4} = 4845$. For each flux value α we optimize the NQS by both SR and SITE and compare the resulting energy with the ED ground-state energy E_{ED}, reporting the relative energy error $\Delta E = |E - E_{\mathrm{ED}}|/|E_{\mathrm{ED}}|$.

The left panel of Fig. 4.2 shows a rapid deterioration of accuracy with

increasing α. Over the range $0 < \alpha \lesssim 0.3$, the relative energy error increases by several orders of magnitude. Strikingly, SR and SITE yield essentially identical error curves, indicating that the degradation is not an artefact of a particular optimization scheme but is instead tied to the underlying variational representation in this regime.

For additional context, Fig. 4.2 also includes results from DMRG calculations performed with TeNPy [99]. To enable a like-for-like comparison, we first choose a small bond dimension ($\chi = 8$) so that the number of MPS parameters (2336) is close to that of the NQS. At low flux, the NQS reaches substantially lower energy errors than this constrained MPS. At higher flux the two approaches become comparable, and DMRG deteriorates much more gently than the NQS. Increasing the bond dimension to $\chi = 24$ strongly improves the DMRG results and clearly outperforms the NQS at large α. However, this improvement comes at the cost of an MPS parameter count (20832) that exceeds $\dim \mathcal{H}$ by roughly a factor of four. Even though the parameter count exceeds $\dim \mathcal{H}$, the fixed MPS structure limits the entanglement the ansatz can represent, reflecting the intrinsically two-dimensional structure of this system.

The sharp flux dependence persists when we remove variational optimization from the picture and directly fit ED wave functions by supervised learning (Sec. 4.3.2). The right panel of Fig. 4.2 aggregates an extensive sweep over model sizes and NQS architectures, and plots the resulting overlap deviation against the compression ratio $N_\theta/\dim \mathcal{H}$. For MLPs, this sweep covers all combinations of hidden-layer depths 2, 3, and 4; widths 32, 64, and 128; lattice sizes 4×4, 4×6, and 4×8 (four rows with 4, 6, or 8 columns); and particle numbers 4, 6, and 8. For CNNs, we repeat the same physical sweep while replacing the MLP width by channel counts 10, 20, and 44. Despite a substantial scatter (reflecting the non-universality of architectural details and physical parameters), the dominant trend is clear: for fixed $N_\theta/\dim \mathcal{H}$, the strong-flux cases ($\alpha = 0.3$) are separated from the zero-flux cases by a systematic gap of roughly three to four orders of magnitude in $d(\psi_{\boldsymbol{\theta}}, \psi_{\mathrm{ED}})$. This motivates the term *universal performance gap* used throughout this chapter.

The apparent universality also suggests that one can study the underlying mechanism efficiently in a controlled setting. Unless stated otherwise, the analyses below therefore use the 4×5 lattice with $N_{\mathrm{b}} = 4$ and the baseline MLP (two layers of width 32), primarily with direct supervised learning of the ED ground state to isolate representational effects and reduce computational overhead. Tests of flux dependence retain the sampled range of α, whereas

paired diagnostics compare the representative values $\alpha = 0$ and $\alpha = 0.3$.

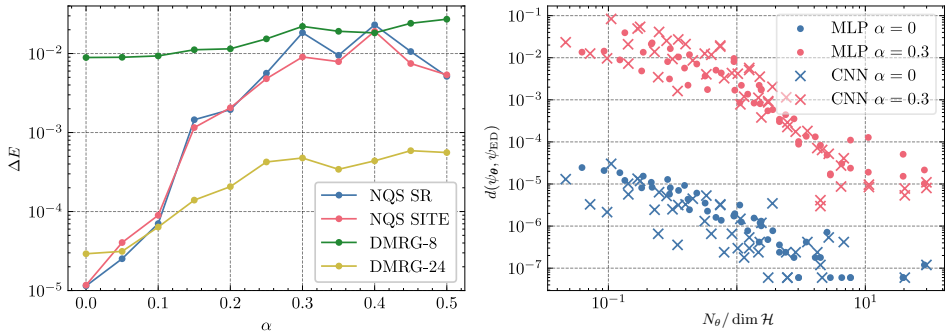

Figure 4.2: **Left:** Dependence of the relative energy error on magnetic flux for SR and SITE, compared with DMRG. For DMRG we show a constrained MPS with bond dimension $\chi = 8$ (green, comparable parameter count to the NQS) and a larger bond dimension $\chi = 24$ (yellow). The two NQS optimizers follow almost the same curve: they are accurate at low flux but deteriorate by several orders of magnitude once α is increased. **Right:** Overlap deviation $d(\psi_\theta, \psi_{\mathrm{ED}})$ versus the compression ratio $N_\theta/\dim\mathcal{H}$ for a sweep over lattice sizes, particle numbers, and MLP/CNN architectures. Blue and red symbols correspond to $\alpha = 0$ and $\alpha = 0.3$, respectively. At fixed compression ratio, the strong-flux points form a visibly separated band with much larger errors.

4.4.2 How many parameters are required?

There is no universal criterion for when an approximate ground state should be considered "good enough" for a given physical question. Nevertheless, it is informative to ask how the parameter count required to reach a fixed accuracy threshold scales with magnetic flux. We define a flux-dependent target error

$$\Delta E_{\mathrm{req}} = 0.1\,(E_{\mathrm{exc}} - E_{\mathrm{ED}}), \tag{4.6}$$

where E_{ED} and E_{exc} are the ground-state and first-excited-state energies obtained from ED; unlike the relative error ΔE, ΔE_{req} is an absolute energy. Reaching this threshold ensures that the variational energy lies well within the many-body gap and that the resulting state is clearly distinguishable from low-lying excitations.

Figure 4.3 shows the minimal number of parameters $N_{\theta,\mathrm{req}}$ needed to achieve $|E - E_{\mathrm{ED}}| \leq \Delta E_{\mathrm{req}}$, i.e. $\Delta E \leq \Delta E_{\mathrm{req}}/|E_{\mathrm{ED}}|$, as a function of α, obtained by

fixing the MLP depth (2, 3, or 4 hidden layers) and varying only the width. The values of $N_{\theta,\mathrm{req}}$ are extracted by linear interpolation between the smallest and largest widths bracketing the threshold. We find that $N_{\theta,\mathrm{req}}$ grows rapidly with α and exhibits pronounced jumps near $\alpha \simeq 0.1$ and $\alpha \simeq 0.25$, suggestive of flux values where the structure of the ground state changes in a way that is particularly unfavourable for the tested ansätze. For $\alpha \gtrsim 0.2$, achieving the target accuracy would require $N_{\theta,\mathrm{req}}$ well above $\dim \mathcal{H}$, i.e. beyond the regime where NQS provide compression at all.

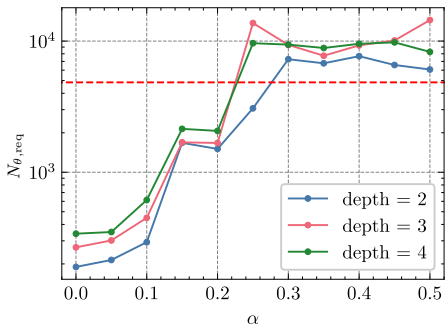

Figure 4.3: Minimal parameter count $N_{\theta,\mathrm{req}}$ needed to reach the target energy error ΔE_{req} as a function of magnetic flux. Blue, red, and green curves correspond to MLP depths of two, three, and four hidden layers. The required size rises sharply with α, with pronounced jumps near $\alpha \simeq 0.1$ and $\alpha \simeq 0.25$; at larger flux it approaches or exceeds the Hilbert-space dimension $\dim \mathcal{H}$, shown by the horizontal dashed line.

4.4.3 Assessing potential causes for the performance drop

The experiments above establish the existence and apparent universality of the performance gap, but they do not reveal what feature of the HBHM ground states makes the strong-flux regime difficult. In the following we report a sequence of numerical diagnostics designed to test several natural hypotheses.

Statistics of wave function elements

As a first probe we inspect the distribution of ground-state amplitudes in the computational basis. Figure 4.4 compares histograms of $\log_{10} |\psi_{\mathrm{ED}}(\boldsymbol{s})|$ for $\alpha = 0$ and $\alpha = 0.3$, as well as the corresponding histograms of $\log_{10} |\psi_{\boldsymbol{\theta}}(\boldsymbol{s})|$ after supervised training. The strong-flux ground state exhibits a noticeably broader

spread of amplitudes. While the NQS can reproduce a similarly wide dynamic range, such broad distributions can still make learning harder because the overlap objective strongly emphasizes the relatively small subset of high-probability configurations.

To test whether this effect could be relevant, we construct a modified target state at $\alpha = 0$ whose amplitudes are artificially broadened (by squaring the ED amplitudes and renormalizing) and repeat the supervised fitting procedure. The resulting overlap deviation becomes comparable to that of the original $\alpha = 0$ ground state, refuting the idea that an increasingly heavy-tailed amplitude distribution is the cause of the difficulty. At the same time, the histogram comparison for $\alpha = 0.3$ reveals a more specific mismatch: the trained NQS under-represents a distinct range of intermediate-amplitude configurations while over-populating the smallest-amplitude tail. We did not identify a simple explanation of this mismatch in terms of obvious basis-state features.

Element-wise errors provide a complementary viewpoint. Figure 4.5 shows the error in norm and phase of individual amplitudes as a function of the exact amplitude norm. Since physically equivalent wave functions may differ by a global phase, we align the NQS output by the overlap phase $\Delta\varphi = \arg\langle\psi(\boldsymbol{\theta})|\psi_{\mathrm{ED}}\rangle$ before evaluating phase errors. At $\alpha = 0.3$ we observe a clear anticorrelation between phase error and amplitude norm, consistent with the fact that high-norm configurations are emphasized more strongly by the overlap loss. Nevertheless, neither the norm- nor phase-error patterns directly point to a single dominant failure mechanism.

Phase structure

Magnetic flux renders the Hamiltonian non-stoquastic, so the ground state cannot be chosen everywhere real and nonnegative. A natural concern is therefore that the difficulty is primarily tied to learning a nontrivial complex phase pattern. To isolate the role of phases, we repeat the supervised-learning experiment while forcing the NQS to fit only the norms or only the phases of the ED amplitudes, supplying the complementary component directly from ED.

The results are summarized in Fig. 4.6. The overlap deviation worsens with α at nearly the same rate even when the NQS fits only the norms, i.e., when it fits a real, nonnegative function as one would for a stoquastic Hamiltonian. This indicates that the complex phase structure is not the main bottleneck. At the same time, fitting only the phases also becomes more difficult at strong flux, so phases may still contribute secondarily.

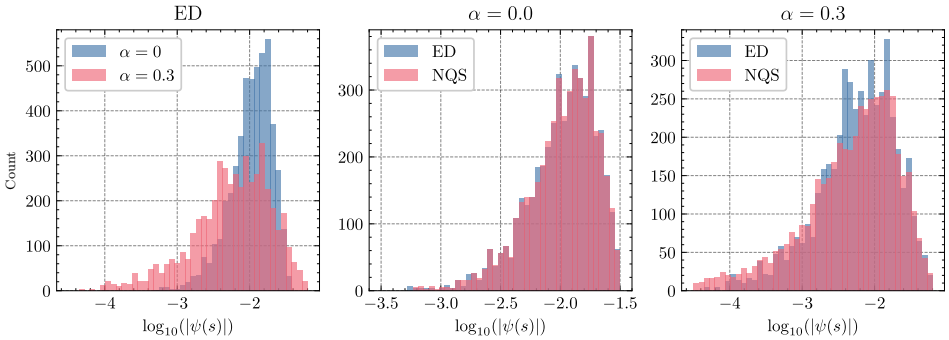

Figure 4.4: Histograms of wave-function element values. **Left:** Comparison of ED ground-state amplitudes without (blue) and with (red) magnetic flux. The finite-flux state has a broader low-amplitude tail. **Middle:** Comparison of trained NQS (red) and ED (blue) amplitudes at $\alpha = 0$, where the two histograms nearly overlap. **Right:** Same comparison at $\alpha = 0.3$; the NQS captures the broad distribution only approximately, with visible deviations in the intermediate- and low-amplitude regions.

Quantum entanglement

Another plausible correlate of difficulty is entanglement. We quantify bipartite entanglement by partitioning the lattice into two halves, performing a Schmidt decomposition, and computing the von Neumann entropy

$$S_{\mathrm{vN}} = -\sum_k \sigma_k^2 \log_2 \sigma_k^2, \tag{4.7}$$

where $\{\sigma_k\}$ are the Schmidt coefficients across the cut, i.e. the singular values of the bipartite coefficient matrix, so that $\{\sigma_k^2\}$ are the eigenvalues of the reduced density matrix. For the baseline 4×5 system, the ED ground state has $S_{\mathrm{vN}} \simeq 1.83$ at $\alpha = 0$ and $S_{\mathrm{vN}} \simeq 2.5$ at $\alpha = 0.3$.

To test whether this increase alone explains the performance gap, we construct families of modified $\alpha = 0.3$ states with reduced entanglement by truncating the Schmidt spectrum. We do this either by truncating a single bipartition in the middle of a one-dimensional site ordering, or by truncating repeatedly along all bonds of that ordering. Figure 4.7 shows that reducing S_{vN} does improve the achievable overlap. However, even when the entropy is reduced to values comparable to the $\alpha = 0$ case, the overlap deviation remains much larger than at $\alpha = 0$. This suggests that increased entanglement is not the primary source of

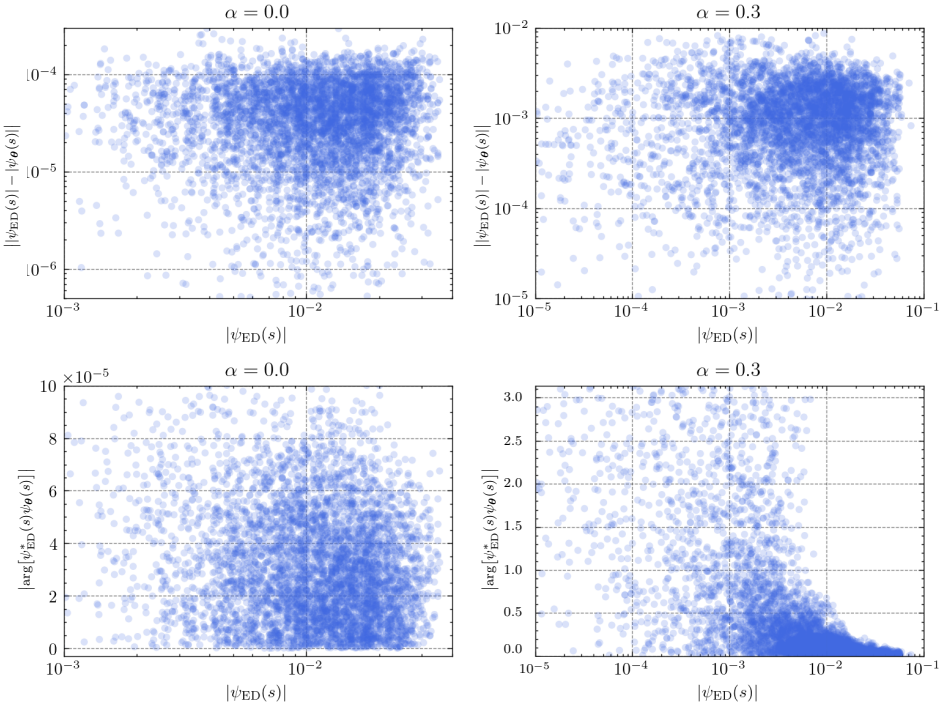

Figure 4.5: **Top:** Norm errors of NQS amplitudes versus the corresponding ED amplitude norms. **Bottom:** Phase errors versus ED amplitude norms, after aligning the global phase. In the axis labels, $\psi_\theta(s)$ and $\psi_{\mathrm{ED}}(s)$ denote the NQS and exact amplitudes of the basis configuration $\boldsymbol{s}$. Left and right columns show $\alpha = 0$ and $\alpha = 0.3$, respectively. At finite flux the exact amplitudes span a much wider range, and the phase-error cloud becomes strongly amplitude dependent: large-amplitude configurations are fitted most accurately, while small-amplitude configurations carry much larger phase errors.

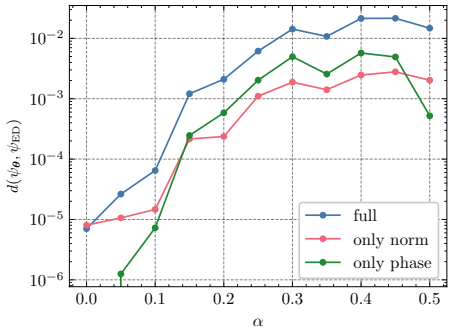

Figure 4.6: Overlap deviation versus magnetic flux when the NQS fits the full complex wave function (blue), only its phase (green), or only its norm (red), with the remaining component taken from ED. All three curves worsen as α increases, and the norm-only curve follows the same overall trend as the full fit, indicating that the performance drop is not solely a failure to learn the complex phase.

the observed difficulty, although it may contribute to it.

Fractional quantum Hall physics

In the continuum, strong magnetic fields and interactions can stabilize fractional quantum Hall states. In lattice Hofstadter systems this physics is expected to manifest as fractional Chern insulators (FCIs) in regimes where a partially filled topological band is effectively isolated [59, 60, 118, 119]. If the strong-flux ground states in our finite systems were approaching such a fractionalized phase, this could plausibly challenge generic NQS architectures.

To test this idea we analyze performance as a function of the filling factor

$$\nu = \frac{N_{\mathrm{b}}}{\alpha(L_x - 1)(L_y - 1)}, \tag{4.8}$$

where $(L_x - 1)(L_y - 1)$ is the number of plaquettes for open boundaries [61]. For bosons, an FCI analogue of the Laughlin state is often expected near $\nu \simeq 1/2$. For fixed N_{b} and lattice geometry, ν is only a reciprocal reparameterization of α and therefore is not an independent control parameter. Its purpose here is instead to compare different geometries on a physically relevant scale: if the loss of accuracy were tied to a state emerging at a particular filling, features occurring at different values of α because of the different plaquette counts should align at a common ν. Figure 4.8 shows the overlap deviation versus α

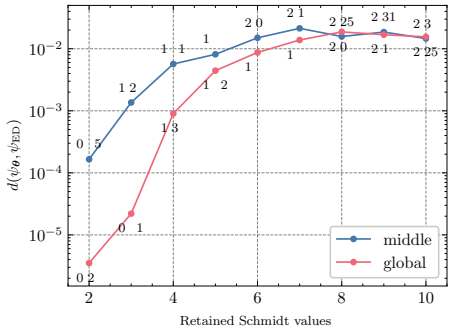

Figure 4.7: Overlap deviation versus the number of kept Schmidt singular values when truncating once at the middle cut of a 1D site ordering (blue, "middle") or repeatedly at every bond (red, "global"). Numbers next to points indicate the resulting von Neumann entropy S_{vN}. More severe truncation lowers both entropy and fitting error, especially for the repeated truncation, but the error remains well above the zero-flux baseline even when the entropy is reduced to comparable values.

(left) and versus ν (right) for several lattice shapes. The dominant growth of the overlap deviation occurs at filling factors well above unity, while the behavior for $\nu < 1$ is irregular and does not display a robust feature pinned to a particular ν. We therefore find no evidence that the performance gap is directly associated with the stabilization of an FCI-like regime in the parameter ranges and system sizes considered here.

Variational loss landscape

In other models, degraded NQS performance has been linked to difficult optimization landscapes, where training becomes trapped in plateaus or poor local minima [100, 115]. To assess whether a similar mechanism is at work here, we visualize the overlap-loss landscape around the trained solution using the 2D projection technique of Ref. [120]. Given optimized parameters $\boldsymbol{\theta}_*$ and two random unit vectors $\boldsymbol{e}_1, \boldsymbol{e}_2$ in parameter space, we consider the surface

$$f(\delta_1, \delta_2) = \mathcal{L}(\boldsymbol{\theta}_* + \delta_1 \boldsymbol{e}_1 + \delta_2 \boldsymbol{e}_2), \tag{4.9}$$

where $\mathcal{L}$ is the exact overlap loss in Eq. (4.4) evaluated using the full wave function vectors.

Figure 4.9 shows representative landscapes for $\alpha = 0$ and $\alpha = 0.3$. In contrast

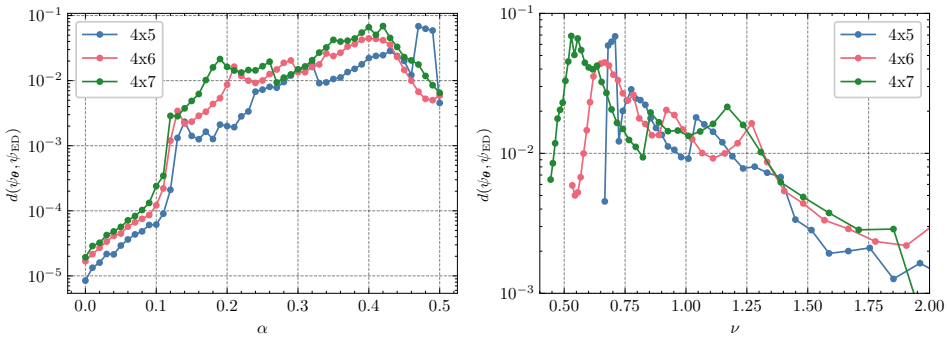

Figure 4.8: Dependence of overlap deviation on magnetic flux **(left)** and on filling factor ν **(right)** for three lattice shapes: 4×5 (blue), 4×6 (red), and 4×7 (green). For each geometry, ν is a reciprocal reparameterization of α; the right panel instead tests whether features from different geometries align at a common, physically relevant filling. No such alignment is observed: the largest degradation occurs over a broad range rather than at a single filling factor.

to the highly rugged surfaces reported in Refs. [100, 115], the HBHM landscapes appear smooth and strongly convex near the optimum. We verified that this qualitative picture is robust across multiple random choices of $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$ and for different values of α.

To quantify the local curvature more directly, we compute the Hessian of $\mathcal{L}$ with respect to the NQS parameters at $\boldsymbol{\theta}_*$. For our baseline MLP, $N_\theta = 1856$ is small enough to make this feasible. The resulting eigenvalue spectra are shown in Fig. 4.10. All eigenvalues are positive, confirming that the trained solution is a (local) minimum and that the local loss landscape is convex for all considered flux values. At the same time, the curvature scale changes dramatically with α: the median Hessian eigenvalue grows approximately exponentially from $\alpha = 0$ to $\alpha = 0.3$.

A very sharp minimum could in principle hinder convergence of first-order optimizers and artificially inflate the final overlap deviation. To rule this out, we refine $\boldsymbol{\theta}_*$ using Newton's method, employing exact gradients and Hessians computed from the full wave function. Despite rapid convergence in parameter space, the improvement in $d(\psi_{\boldsymbol{\theta}}, \psi_{\mathrm{ED}})$ is only on the order of 10^{-4}, far too small to explain the three-to-four order-of-magnitude performance gap. We therefore conclude that, for the HBHM and system sizes studied here, neither ruggedness nor local curvature of the loss landscape is the dominant source of the observed performance degradation.

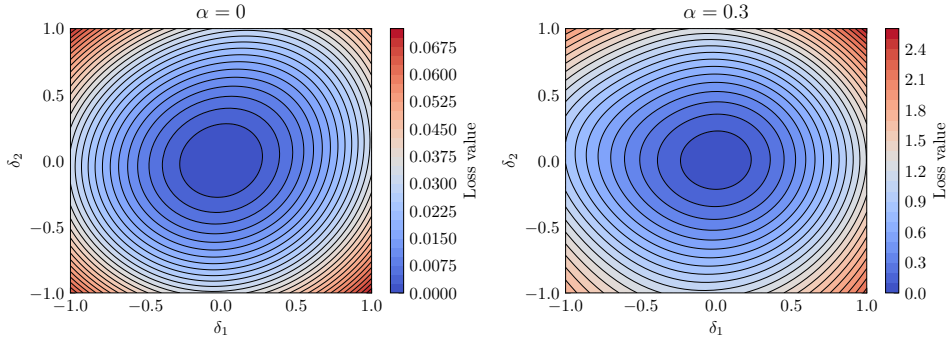

Figure 4.9: Two-dimensional loss-landscape projections around the trained NQS parameters at $\alpha = 0$ **(left)** and $\alpha = 0.3$ **(right)**. Both projections show a smooth basin centered near the optimized parameters, rather than a visibly rugged landscape. The strong-flux basin has a much larger loss scale, consistent with a sharper local minimum but not with trapping in many nearby poor minima.

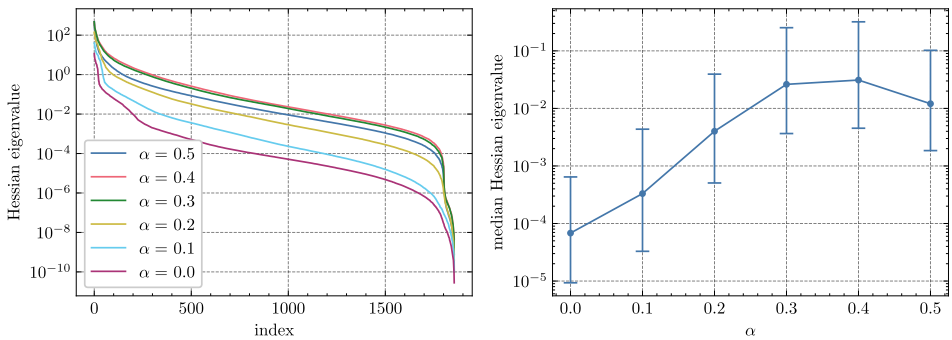

Figure 4.10: **Left:** Hessian eigenvalue spectra (sorted by magnitude) of the overlap loss at the trained minima for several flux values. The spectra remain positive but shift upward as the flux increases, showing that the local curvature grows over much of parameter space. **Right:** Median Hessian eigenvalue versus α, with error bars indicating the 25th and 75th percentiles. The median curvature rises by several orders of magnitude between $\alpha = 0$ and $\alpha \simeq 0.3$ before saturating within the plotted range.

4.4.4 Experiments with NQS modifications

The universal nature of the gap suggests that it is not tied to a fragile implementation detail. Nevertheless, NQS training involves many design choices, and it is natural to ask whether modest modifications can mitigate the strong-flux degradation. In this section we summarize a set of targeted experiments on the baseline problem (4×5 lattice, $N_b = 4$, MLP with two hidden layers of width 32) comparing $\alpha = 0$ and $\alpha = 0.3$. Results are collected in Table 4.1.

Complex weights

Since non-stoquastic ground states are inherently complex, one might expect complex-valued network parameters to represent them more efficiently. We therefore implement an NQS with complex weights and activations applied separately to real and imaginary parts. As Table 4.1 shows, complex parameters yield only a modest improvement at $\alpha = 0.3$. Given that complex weights effectively double the number of degrees of freedom for fixed layer sizes, this small gain does not suggest a qualitatively better inductive bias for the strong-flux regime.

Mean-squared error loss

We also test replacing the overlap loss by a mean-squared error objective on the complex amplitudes. We consider both uniform sampling over basis states and importance sampling according to $|\psi_{\boldsymbol{\theta}}(\boldsymbol{s})|^2$, finding that the latter performs substantially better. Even so, the mean-squared-error (MSE) loss leads to markedly worse overlaps than the baseline overlap-loss training.

Input encoding

To test whether the difficulty stems from an unfortunate representation of configurations, we experiment with alternative input encodings. These include: (i) applying a two-dimensional Fourier transform to the occupation pattern before feeding it to the network; (ii) a learnable "patch" embedding that linearly encodes each 2×2 block before concatenation [121]; and (iii) learnable token embeddings for site position and occupation number, inspired by NLP models. As summarized in Table 4.1, none of these encodings substantially improves the strong-flux performance, although the patch encoding performs best among them.

Modification	$d(\psi_{\boldsymbol{\theta}}, \psi_{\mathrm{ED}})$ **at** $\alpha = 0$	$d(\psi_{\boldsymbol{\theta}}, \psi_{\mathrm{ED}})$ **at** $\alpha = 0.3$
Baseline	$6.95 \cdot 10^{-6}$	$1.23 \cdot 10^{-2}$
Complex parameters	$5.84 \cdot 10^{-6}$	$6.51 \cdot 10^{-3}$
MSE loss	$2.72 \cdot 10^{-5}$	$6.28 \cdot 10^{-2}$
Fourier transformed input	$1.22 \cdot 10^{-5}$	$1.66 \cdot 10^{-2}$
Patches encoding	$7.80 \cdot 10^{-6}$	$1.37 \cdot 10^{-2}$
Learnable embeddings	$1.92 \cdot 10^{-5}$	$1.85 \cdot 10^{-2}$
Float64	$7.01 \cdot 10^{-6}$	$1.15 \cdot 10^{-2}$
Curriculum learning	—	$2.14 \cdot 10^{-2}$

Table 4.1: Overlap deviations achieved with various NQS modifications at $\alpha = 0$ and $\alpha = 0.3$. The baseline uses the overlap loss in Eq. (4.4) and the architecture described in Sec. 4.3.1.

Curriculum learning

Curriculum learning can help in problems where a good solution for a simpler task provides a useful initialization for a harder one [122]. We attempt such a curriculum by starting from $\alpha = 0$ and increasing α gradually up to 0.3 during training. In our setting this does not improve the final overlap and in fact slightly worsens it. Empirically, the ground state changes abruptly at certain flux values, so that previously learned representations become less relevant; in addition, continuing training across such changes may trap the optimizer in worse basins than training from scratch at the final α.

Numerical precision

Finally, we test whether limited numerical precision could be responsible for the observed gap by repeating the baseline experiments using 64-bit floating point numbers throughout. The results are essentially unchanged, indicating that the strong-flux difficulty is not a simple consequence of floating-point roundoff in our implementation.

4.5 Conclusions

The central finding of this chapter, summarized in Statement 2, is that the hard-core HBHM provides a particularly stark example of an NQS failure regime. As the magnetic flux per plaquette increases, the accuracy of generic MLP and CNN ansätze degrades sharply: relative energy errors obtained by SR

and SITE grow by orders of magnitude, and even direct supervised fitting of ED ground states exhibits a systematic three-to-four order-of-magnitude overlap gap between $\alpha = 0$ and $\alpha = 0.3$ at fixed compression ratio $N_\theta/\dim\mathcal{H}$ [P2]. Moreover, achieving a fixed energy-accuracy threshold at strong flux requires parameter counts that can exceed the Hilbert-space dimension, undermining the premise of NQS as a compressed representation.

We also investigated several natural explanations for this performance gap; the outcome of this process of elimination is summarized in Fig. 4.11. The difficulty is not primarily tied to learning a complex phase pattern: fitting only the amplitude norms already becomes hard at strong flux. Increased entanglement correlates with flux, but artificially reducing entanglement of the $\alpha = 0.3$ ground state does not restore the easy-to-learn behavior of $\alpha = 0$. We find no signature that the degradation is linked to fractional Chern insulating physics in the considered finite-size regimes. Perhaps most strikingly, the overlap-loss landscape in the HBHM appears smooth and locally convex near trained solutions, with Hessian spectra showing well-defined minima rather than rugged plateaus. Local second-order refinement by Newton's method yields negligible improvements, ruling out the possibility that poor performance is simply due to failure to converge within a rugged landscape.

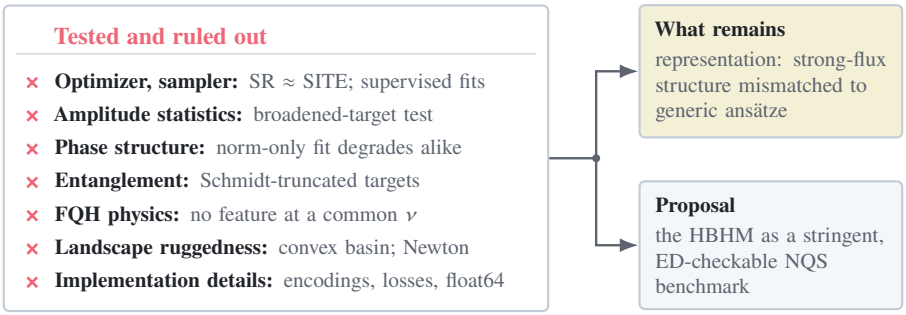

Figure 4.11: The elimination argument of this chapter in summary. Each candidate explanation is ruled out as the dominant cause of the performance gap by the test listed next to it (Secs. 4.4.1, 4.4.3, and 4.4.4); entanglement and phase structure may still contribute secondarily. What survives the elimination is a representational limitation of the tested ansätze, in the sense of Fig. 4.1(b), and the practical outcome is the HBHM itself as a benchmark: the gap is already pronounced at system sizes where every diagnostic can be checked against exact diagonalization.

Taken together, these results suggest that the strong-flux HBHM ground states

possess structure that is systematically hard for unspecialized NQS architectures to exploit, even when optimization and sampling issues are largely eliminated. This distinguishes the HBHM from previously studied failure cases where rugged energy landscapes appear to be the dominant obstacle. At present, however, it is difficult to quantify this obstruction more sharply without a correspondingly robust theory of NQS expressivity. It therefore remains unclear whether the observed gap can ultimately be closed by better inductive biases and optimization strategies, or whether it reflects a deeper limitation of generic NQS representations in magnetic lattice systems. At the same time, the HBHM setting remains computationally convenient: the performance gap is already pronounced at small sizes, enabling rapid controlled experiments and detailed diagnostics. We therefore propose the HBHM as a complementary benchmark for stress-testing NQS methods and for motivating new architectures and training strategies tailored to magnetic and frustrated lattice systems. Its difficulty can also be increased in a controlled way, for example by relaxing the hard-core constraint or by introducing longer-range interactions, making it a flexible platform for future benchmark studies.

5 Neural importance resampling

This chapter is based on the results published in [P3].

5.1 Introduction

As discussed in previous chapters, NQS have become one of the state-of-the-art tools for approximating ground states and other properties of many-body quantum systems [30–32, 123]. In the VMC setting, the central technical ingredient is a reliable sampler for the Born distribution defined by the variational wave function (Sec. 2.3.2, Sec. 2.3.3). When sampling fails, observable estimates become biased and optimization can become unstable, irrespective of how expressive the underlying neural network ansatz is.

The default sampling strategy in the NQS literature is MCMC, most commonly the Metropolis–Hastings algorithm (Sec. 2.3.8). While asymptotically correct, MCMC can be impractical in regimes where $|\psi(\boldsymbol{s})|^2$ is sharply peaked or strongly multi-modal: mixing becomes slow, autocorrelations reduce the effective batch size, equilibration is difficult to diagnose, and local proposals can become trapped. More sophisticated *learned* proposals can alleviate some of these issues, but they also introduce additional constraints. In particular, adaptive MCMC schemes must preserve stationarity (or detailed balance) and satisfy conditions such as diminishing adaptation and bounded convergence [124, 125], which complicates both implementation and hyperparameter tuning.

A conceptually different way to avoid Markov chains is to use *autoregressive* NQS that can be sampled exactly by ancestral generation (Sec. 2.3.9). However, the required autoregressive factorization constrains the NQS architecture and can make global constraints and symmetries harder to enforce [114]. Moreover, in the multi-state VMC formulation (Sec. 2.3.11), the target distribution is defined by a determinant of multiple single-state NQS. Even if each constituent network were autoregressive, it is not clear how to sample *exactly* from the determinant distribution.

In this chapter we present *neural importance resampling* (NIR), an alternative

sampling approach that decouples the NQS ansatz from the sampler. NIR replaces MCMC with importance resampling [126], using proposals generated by a separate autoregressive *sampling neural network* (SNN). The SNN is trained to approximate the target NQS distribution, making importance resampling practical even in the high-dimensional configuration spaces typical of lattice many-body problems. Compared to MCMC, NIR produces independent proposals and avoids the need for proposal design and convergence diagnostics. Compared to fully autoregressive NQS, it leaves the variational ansatz unconstrained and supports sampling of more complex constructions such as multi-state determinant NQS.

This chapter is structured as follows. In Sec. 5.2 we introduce the sampling targets of interest, with emphasis on the multi-state determinant NQS construction. Sec. 5.3 describes the NIR algorithm: importance resampling, the autoregressive proposal network, the objective used to train it, and an adaptive retraining scheme that stabilizes sampling throughout optimization. In Sec. 5.4 we present numerical experiments on the TFIM and the frustrated J_1–J_2 Heisenberg model. Finally, Sec. 5.5 summarizes the main conclusions and discusses practical implications.

5.2 Sampling targets in VMC

This chapter builds on the NQS and VMC framework introduced in Chapter 2, so we only recall the notation needed to specify the sampling targets. In standard (single-state) VMC one samples basis configurations $\boldsymbol{s}$ from the Born distribution of the variational wave function,

$$p(\boldsymbol{s}) \propto |\psi_{\boldsymbol{\theta}}(\boldsymbol{s})|^2, \tag{5.1}$$

which enables unbiased Monte Carlo estimates of local observables and energy gradients (Sec. 2.3.2, Sec. 2.3.4). NIR is agnostic to the particular VMC objective: it only assumes access to an unnormalized target density proportional to p.

Multi-state determinant target. To target the lowest K eigenstates simultaneously we adopt the multi-state NES-VMC formulation of Sec. 2.3.11. It introduces an extended configuration $\boldsymbol{S} = (\boldsymbol{s}_1, \ldots, \boldsymbol{s}_K)$ and a determinant wave function $\Psi_{\text{tot}}(\boldsymbol{S})$ (Eq. (2.111)). The corresponding sampling target is the Born

distribution on the extended system,

$$p(\boldsymbol{S}) \propto \left|\Psi_{\text{tot},\boldsymbol{\theta}}(\boldsymbol{S})\right|^2 . \tag{5.2}$$

Since permuting the copies changes the determinant sign but not its magnitude, $p(\boldsymbol{S})$ is permutation-invariant under reordering of $(\boldsymbol{s}_1, \ldots, \boldsymbol{s}_K)$. The variational objective is the trace of the energy matrix estimated from samples $\boldsymbol{S} \sim |\Psi_{\text{tot}}|^2$ (Eq. (2.116)).

This multi-state construction sharpens the sampling challenge: the sampler must now act on K coupled copies at once, so the number of sampled variables grows by a factor K and the cardinality of the configuration space grows exponentially, as $|\mathcal{S}|^K$ for a single-copy space $\mathcal{S}$. Moreover, exact autoregressive sampling is generally unavailable for determinant targets, motivating the sampling strategy developed next.

5.3 Neural importance resampling

5.3.1 Importance resampling

Importance resampling (IR), also known as sampling-importance resampling [126], draws samples from a target distribution $p(\boldsymbol{s})$ using proposals from a different distribution $q(\boldsymbol{s})$. Given N_q independent proposals $\boldsymbol{s}^{(j)} \sim q$, one computes importance weights

$$w_j = \frac{p(\boldsymbol{s}^{(j)})}{q(\boldsymbol{s}^{(j)})}, \tag{5.3}$$

and then draws N_p resampled configurations by selecting elements of the proposal batch with probabilities proportional to w_j. In the NQS setting, p is the Born distribution defined by the variational wave function (where $\boldsymbol{s}$ may be a multi-copy tuple in the multi-state formulation), and it is sufficient to use *unnormalized* weights proportional to $|\Psi_{\boldsymbol{\theta}}(\boldsymbol{s})|^2/q(\boldsymbol{s})$ because resampling depends only on relative weights; here and in Fig. 5.2, $\Psi_{\boldsymbol{\theta}}$ stands for the target amplitude, i.e. $\psi_{\boldsymbol{\theta}}(\boldsymbol{s})$ in single-state VMC and $\Psi_{\text{tot}}(\boldsymbol{S})$ in the multi-state case.

If the only goal were to compute expectation values under p, one could also use standard importance sampling: estimate expectations using the weighted proposal batch without resampling. In this work we use resampling primarily as an engineering choice: it allows NIR to serve as a drop-in replacement for an MCMC sampler in an existing VMC pipeline without modifying downstream

code that expects samples distributed approximately as p.

The main limitation of IR is that it requires substantial overlap between p and q. In high-dimensional problems, constructing such a proposal by hand is usually infeasible, which is why IR is often considered a low-dimensional technique. NIR addresses this by *learning* q with a neural network.

5.3.2 Autoregressive proposal network

We parameterize the proposal distribution q with an autoregressive neural network (Sec. 2.3.9), which can be sampled exactly and evaluated efficiently. Concretely, we model a joint distribution over discrete variables $\boldsymbol{s} = (s_1, \ldots, s_N)$ as

$$q(\boldsymbol{s}) = \prod_{k=1}^{N} q(s_k \mid \boldsymbol{s}_{<k}), \qquad \boldsymbol{s}_{<k} = (s_1, \ldots, s_{k-1}), \tag{5.4}$$

and obtain independent proposal samples by ancestral generation. In the multi-state case we apply the same idea to tuples by fixing an ordering of the K configurations and an ordering of sites within each configuration; the tuple is then treated as a single sequence of discrete tokens.

In our implementation q is represented by a Transformer [74] with reordered layer normalization [75], motivated by the success of Transformers as general-purpose generative models [127]. Each site value is embedded into a learnable vector of dimension d_{model} (the token embedding dimension of Sec. 2.2.3), learnable positional embeddings are added, and the resulting sequence is processed by standard Transformer blocks with a causal attention mask. Each output token is linearly projected to logits for the corresponding conditional distribution. To ensure that the prediction for s_k depends only on $\boldsymbol{s}_{<k}$, the input is shifted by prepending a learnable start token and discarding the final token. We refer to this autoregressive model as the *sampling neural network*.

5.3.3 Training objective: forward vs. backward Kullback–Leibler divergence

To make IR effective, the proposal distribution should match the target distribution closely and, crucially, it should not assign negligible probability mass to configurations that are likely under the target. A natural training objective is therefore the Kullback–Leibler divergence (KLD) between the target p and the

proposal q. Two common variants are the forward (mean-seeking) KLD,

$$\mathcal{L}_{\mathrm{f}} = \sum_{\boldsymbol{s}} p(\boldsymbol{s}) \log \frac{p(\boldsymbol{s})}{q(\boldsymbol{s})}, \tag{5.5}$$

and the backward (mode-seeking) KLD,

$$\mathcal{L}_{\mathrm{b}} = \sum_{\boldsymbol{s}} q(\boldsymbol{s}) \log \frac{q(\boldsymbol{s})}{p(\boldsymbol{s})}. \tag{5.6}$$

Their gradients can be written as

$$\nabla_{\boldsymbol{\phi}} \mathcal{L}_{\mathrm{f}} = -\left\langle \nabla_{\boldsymbol{\phi}} \log q(\boldsymbol{s}) \right\rangle_{\boldsymbol{s} \sim p}, \tag{5.7}$$

$$\nabla_{\boldsymbol{\phi}} \mathcal{L}_{\mathrm{b}} = \left\langle (\log q(\boldsymbol{s}) - \log p(\boldsymbol{s})) \nabla_{\boldsymbol{\phi}} \log q(\boldsymbol{s}) \right\rangle_{\boldsymbol{s} \sim q}, \tag{5.8}$$

where $\boldsymbol{\phi}$ denotes the learnable weights of the SNN and $\nabla_{\boldsymbol{\phi}}$ the gradient with respect to them. In the backward-KLD gradient, the additional score-function term $\left\langle \nabla_{\boldsymbol{\phi}} \log q(\boldsymbol{s}) \right\rangle_{\boldsymbol{s} \sim q}$ has been dropped because $\nabla_{\boldsymbol{\phi}} \sum_{\boldsymbol{s}} q(\boldsymbol{s}) = 0$.

The backward KLD is attractive because it only requires samples from q, which are available exactly and without bias. However, for importance resampling the forward KLD is better aligned with the requirement that $q(\boldsymbol{s}) > 0$ whenever $p(\boldsymbol{s}) > 0$. Empirically we found both losses to be viable, but forward-KLD training yields more robust VMC optimization with reduced sensitivity to hyperparameters. Backward-KLD training can suffer from mode collapse: the SNN may focus on a subset of high-probability modes, leaving other relevant configurations with extremely small $q(\boldsymbol{s})$ that are rarely proposed. In that regime, IR occasionally produces large, destabilizing importance weights when such configurations are sampled, which manifests as sudden collapses of the effective sample size discussed below. Figure 5.1 contrasts the two regimes: a mass-covering proposal keeps all importance weights comparable, whereas a mode-collapsed proposal turns every visit to a neglected mode into a dominating weight. To further enforce nonzero support we add a small floor p_{floor} to single-site probabilities in q, ensuring $q(\boldsymbol{s}) > 0$ for all configurations.

For the multi-state determinant NQS of Sec. 2.3.11, the target distribution is permutation-invariant under reordering of the K configurations. Since the autoregressive proposal depends on an ordering, in these calculations we encourage approximate permutation invariance by randomly permuting the order of the K configurations before evaluating the SNN during training.

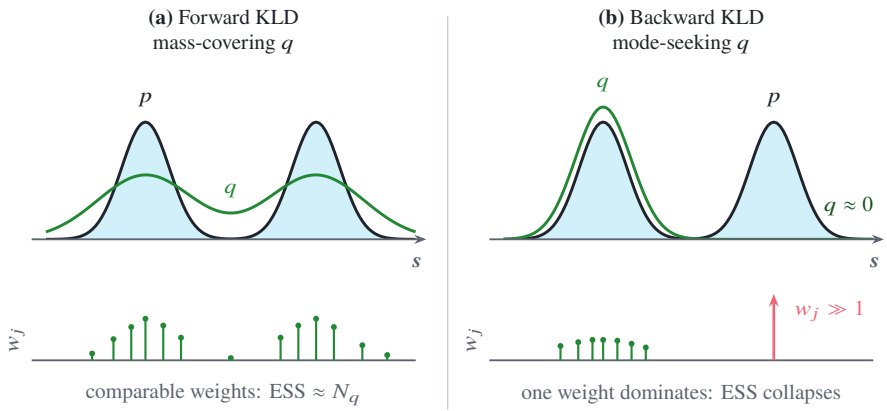

Figure 5.1: Why the proposal must cover the target. Both panels show the same bimodal Born distribution p (cf. Fig. 2.14) together with a learned proposal q and, below on the same horizontal coordinate, the importance weights $w_j = p/q$ of a batch of proposals drawn from q. **(a)** The mass-covering proposal favoured by the forward KLD is nonzero wherever p is appreciable: both modes are proposed, the weights remain comparable, and the ESS stays close to the number of proposals N_q. **(b)** A mode-collapsed proposal, the failure mode of backward-KLD training: configurations in the covered mode are proposed with modest weights, but the rare proposal that lands in the neglected mode (kept possible at all by the floor p_{floor}) receives an enormous weight that dominates the batch, and the ESS collapses. The sampling-efficiency criterion of Sec. 5.3.4 detects this regime and triggers retraining of the SNN.

5.3.4 Adaptive proposal network retraining

During VMC optimization the target distribution p changes as NQS parameters are updated. If the SNN is not kept sufficiently close to the current target, the overlap between p and q can deteriorate and the IR step can become inefficient or unstable. NIR addresses this by explicitly monitoring the *effective sample size*

$$\text{ESS} = \frac{\left(\sum_j w_j\right)^2}{\sum_j w_j^2}, \tag{5.9}$$

computed from the weights of the current proposal batch. Intuitively, ESS estimates how many proposals meaningfully contribute after reweighting: when the weights are uniform, ESS $\approx N_q$, while when a few weights dominate, ESS becomes small.

We generate proposals until ESS exceeds $\alpha_{\text{ESS}} N_p$, where α_{ESS} is a dimensionless multiplier of the resampled batch size N_p, and then perform importance resampling to obtain a batch approximating the target distribution. If the overlap is poor, more proposals are needed to reach the threshold, which automatically allocates more sampling work to difficult regimes. However, in extreme mismatch cases this process can become very slow. To prevent divergence between p and q we introduce a second criterion: we only update the NQS using a resampled batch if the *sampling efficiency*,

$$\text{eff} = \frac{\text{ESS}}{n_{\text{proposed}}}, \tag{5.10}$$

exceeds a threshold α_{eff}, where n_{proposed} is the total number of proposals drawn to reach the ESS target. If the efficiency is below threshold, we update the SNN (using the forward-KLD loss on the resampled batch) and repeat the sampling procedure before taking an NQS optimization step. This yields a simple adaptive retraining mechanism that keeps the proposal aligned with the target throughout VMC. Figure 5.2 shows a schematic overview and algorithm 2 shows the pseudocode for the NQS optimization with NIR sampling.

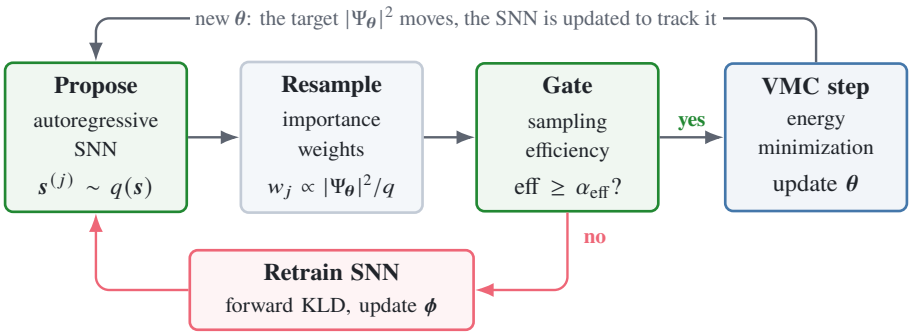

Figure 5.2: One iteration of NQS training with NIR (cf. the generic VMC loop of Fig. 2.12, whose sampler stage NIR replaces). Proposals are generated by the sampling network until the ESS threshold $\alpha_{\text{ESS}} N_p$ is met, then resampled with the importance weights w_j to approximate the NQS Born distribution. The efficiency gate separates the two nested loops: if the sampling efficiency falls below α_{eff}, the inner loop (red) retrains the SNN on the forward KLD and redraws proposals before any NQS update; otherwise the batch is used for an NQS energy-minimization step, and the outer loop repeats with the SNN updated once more to track the target moved by the new parameters.

Algorithm 2 NQS training with neural importance resampling

```
NQS (θ) ← randomInitialization
SNN (ϕ) ← randomInitialization
for t = 1, ..., N_iter do
    ok ← false
    while ¬ok do
        B_q ← propose(q)                    ▷ draw proposals until ESS ≥ α_ESS N_p
        {w_j} ← weights(B_q, |Ψ_θ|^2/q)
        B_p ← resample(B_q, {w_j}, N_p)
        eff ← ESS/n_proposed
        if eff ≥ α_eff then
            ok ← true
        else
            ϕ ← updateSNN(ϕ, B_p)           ▷ forward-KLD update
        end if
    end while
    θ ← updateNQS(θ, B_p)                   ▷ VMC step (e.g. minSR)
    ϕ ← updateSNN(ϕ, B_p)                   ▷ track the updated target
end for
```

5.3.5 Related work

Recent work across machine learning and computational science communities has shown that importance resampling becomes practical in high-dimensional spaces once the proposal distribution is learned by a neural network. Neural Adaptive Sequential Monte Carlo introduced gradient-based adaptation of recurrent proposal networks, minimizing the forward-KLD to the target and repeatedly resampling particles to maintain diversity [128]. Building on this idea, Auto-Encoding SMC integrates proposal learning with variational inference, treating resampling weights as part of an evidence lower bound optimized jointly with generative and proposal networks [129]. In computer graphics, neural importance sampling employs normalizing flows to learn globally coherent proposals that reduce Monte Carlo variance when integrating high-dimensional light-transport integrands [130]. Flow-based proposals have also been used for event generators in particle physics, where exhaustive neural importance sampling yields unbiased estimates with competitive efficiency [131].

Generative neural samplers have also been applied in statistical physics. [132] propose a framework for asymptotically unbiased estimation of observables using generative neural samplers, including settings where the partition function

is unknown. Autoregressive neural networks have been used as adaptive proposals in MCMC for spin-glass models [133], and adaptive MCMC methods augmented with normalizing-flow transitions can learn nonlocal proposal kernels and accelerate sampling in high-dimensional spaces [134]. For the NQS setting considered here, NIR is preferable to a flow-augmented MCMC kernel because it replaces the Markov chain by explicitly reweighted independent proposals, avoiding autocorrelation and burn-in diagnostics while leaving the NQS ansatz unchanged. Recent work further explores the interface between flow-based proposals and energy-based modeling, including balanced learning dynamics between proposal and target models [135] and systematic benchmarks clarifying when machine-learning-assisted Monte Carlo methods outperform conventional algorithms [136].

Collectively, these studies motivate NIR: by training a dedicated neural proposal model with KL-based objectives, one can achieve the overlap required for efficient importance resampling in high-dimensional configuration spaces.

5.4 Numerical experiments

5.4.1 Transverse-field Ising model

We first test NIR on the two-dimensional TFIM introduced in Sec. 2.1 (Eq. (2.5)). This model provides a stringent sampler benchmark because in the ferromagnetic regime ($g \ll 1$) the ground-state distribution is sharply peaked and effectively bimodal, with the two dominant configurations (all spins up and all spins down in the σ^z basis) separated by an extensive Hamming distance. On the infinite square lattice the quantum phase transition occurs at $g \approx 3.0444$ [40].

The largest lattice considered is 8×8. While this size may appear modest, targeting multiple low-energy states using the multi-state formulation of Sec. 2.3.11 increases the effective problem size: for $K = 3$ states, the expanded system contains $3 \times 8 \times 8 = 192$ spin variables.

5.4.2 Multi-state NQS and implementation details

We search for the three lowest-energy states ($K = 3$) using the determinant ansatz of Sec. 2.3.11, which requires three independent single-state NQS networks. Each single-state network is an MLP with $n_{\text{layer}} = 4$ hidden layers and width $d_{\text{mlp}} = 128$. Every linear projection is followed by layer normalization [76] and a GELU activation [65]. The SNN architecture is the Transformer proposal

network. We use the Adam optimizer [83] for both the NQS and the SNN. For the NQS we employ the minSR method [34] to improve convergence (Sec. 2.3.7). NIR is implemented in the in-house JAX-based NQS library described in Sec. 2.4.

The additional cost of NIR comes from autoregressive proposal generation, evaluating proposal probabilities, and SNN updates. In the experiments reported here, these SNN-related operations accounted for roughly 40–60% of the wall-clock time per VMC update, with the precise fraction depending on the current overlap between p and q and therefore on how often the adaptive retraining loop was triggered. This is a substantial overhead relative to an NQS update alone, but it remained practical because the proposal network is smaller than the NQS ensemble and because improved overlap reduces the number of rejected or poorly mixed sampling steps that would otherwise be spent in MCMC. The main hyperparameters used for the TFIM experiments are summarized in Table 5.1.

Hyperparameter	**Symbol**	**Value**
Number of MLP hidden layers	n_{layer}	4
MLP hidden-layer width	d_{mlp}	128
SNN embedding dimension	d_{model}	32
SNN attention heads	–	4
SNN Transformer layers	–	4
NQS learning rate	–	10^{-3}
SNN learning rate	–	10^{-3}
NQS training steps	–	$10,000$
Resampled batch size (NQS and SNN updates)	N_p	512
ESS threshold multiplier	α_{ESS}	2.0
Sampling-efficiency threshold	α_{eff}	0.1
MCMC warmup steps	–	640
MCMC thinning interval	–	10

Table 5.1: Main hyperparameters used in the TFIM experiments.

5.4.3 Impact of the sampler: NIR vs. MCMC on the TFIM

As previously noted in [71] and [137], TFIM poses challenges for VMC with an MCMC sampler in the ferromagnetic phase. To isolate the impact of sampling, we perform pairs of runs that differ only by the sampler (NIR vs. MCMC), keeping all other hyperparameters fixed. Figure 5.3 (left) compares the final variational energy (the trace of the energy matrix in Eq. (2.116)) across

transverse-field strengths and lattice sizes. Each data point is an average over three random seeds. On the 4×4 lattice both samplers perform similarly, but a clear performance gap opens in the ferromagnetic regime already on 6×6 and becomes pronounced on 8×8. With MCMC, training often becomes unstable and the NQS can become trapped in an excited state with significantly higher total energy. In contrast, NIR maintains stable optimization. Figure 5.3 (right) shows a representative training trajectory for 8×8 and $g = 0.1$, plotted as the error of E_{tot}.

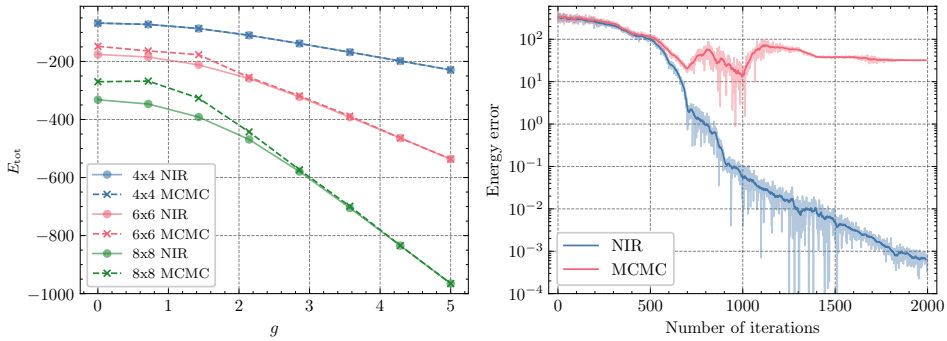

Figure 5.3: **Left:** Dependence of the trace of the final variational energy matrix, E_{tot} of Eq. (2.116), on transverse field strength g for three lattice sizes (4×4, 6×6, 8×8). For every combination, two data points are shown, differing only in the sampler used. The two samplers agree on the smallest lattice, but on larger lattices the MCMC points sit at substantially higher energies in the ferromagnetic regime. **Right:** Example of the evolution of the error in the sum of the three lowest energies, i.e. in E_{tot}, during a training run using NIR (blue) and MCMC (red) for TFIM on an 8×8 lattice and $g = 0.1$. The transparent curves show per-iteration estimates from the optimization batch, and the solid lines represent moving averages. NIR continues to decrease over the full run, while MCMC stalls at an error many orders of magnitude larger.

To further support the interpretation that the observed gap is driven by sampling failure, we compare both samplers to exact sampling on a small 3×2 lattice with $g = 0.01$. In the $K = 3$ multi-state formulation this corresponds to 18 spins, which is close to the practical limit for exact evaluation over all configurations. We sample 25600 configurations using either NIR or MCMC, estimate the empirical configuration probabilities $p_{\text{emp}}(\boldsymbol{s})$, and compute their Jensen–Shannon divergence (JSD) from the exact probabilities $p(\boldsymbol{s})$ obtained

by enumerating the NQS over all configurations,

$$\mathrm{JSD}(p_{\mathrm{emp}} \,\|\, p) = \frac{1}{2} \sum_{\boldsymbol{s}} \left[p_{\mathrm{emp}}(\boldsymbol{s}) \log \frac{p_{\mathrm{emp}}(\boldsymbol{s})}{m(\boldsymbol{s})} + p(\boldsymbol{s}) \log \frac{p(\boldsymbol{s})}{m(\boldsymbol{s})} \right], \tag{5.11}$$

where $m = \frac{1}{2}(p_{\mathrm{emp}} + p)$ is the mixture of the two distributions; the JSD is a symmetrized and bounded variant of the KLD that stays finite when $p_{\mathrm{emp}}(\boldsymbol{s}) = 0$ for some configurations with $p(\boldsymbol{s}) > 0$, as is unavoidable for a finite sample. Figure 5.4 shows the JSD evolution during VMC. Early in training the divergences are comparable, but as the NQS approaches low-energy states, the JSD for MCMC increases while the JSD for NIR decreases as the SNN learns the target distribution.

In this experiment we disabled the adaptive retraining scheme of Sec. 5.3.4 to simplify the dynamics: we used a fixed number of proposal samples and performed one SNN update per NQS update. For such a small system the overlap between the MCMC sample and the true distribution remains sufficient for successful VMC optimization, but the divergence trends already foreshadow the breakdown observed at larger lattices.

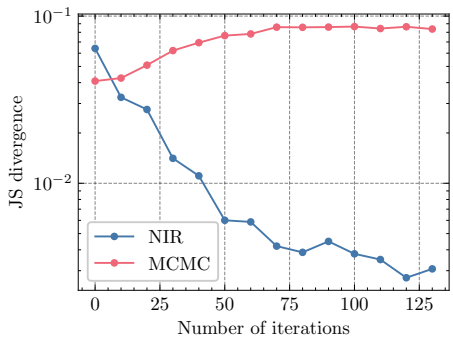

Figure 5.4: Evolution of the JSD between empirical configuration probabilities from the MCMC and NIR samplers and the exact probabilities during NQS optimization for a 3×2 TFIM lattice with $g = 0.01$. As training proceeds, the NIR divergence decreases by more than an order of magnitude, whereas the MCMC divergence grows and then plateaus, showing the onset of the sampling mismatch even on this small system.

5.4.4 Comparing NIR with DMRG

To benchmark the accuracy of the multi-state NQS in the critical regime, we compute the three lowest energy levels on the 8×8 lattice for $g \in [0, 6]$ and compare to density matrix renormalization group calculations [14, 138] with virtual bond dimension 100, performed with TeNPy (Sec. 2.4). The left panel of Fig. 5.5 shows the resulting energy gaps (with NQS energies shifted by the NQS ground-state estimate), capturing the closing and reopening of the gap near $g \in (2, 3)$.

For a more precise benchmark, we compute DMRG energies for bond dimensions 25, 50, 100, and 200 and extrapolate to infinite bond dimension by fitting $E(\chi) = a + b(\chi + c)^d$, where a, b, c, and d are fit parameters. The right panel of Fig. 5.5 shows the NQS error relative to the extrapolated values. The agreement is at the level of $\sim 10^{-7}$ for small and large g and degrades to $\sim 10^{-5}$ near criticality, consistent with the increased difficulty of converging both methods in a gapless regime.

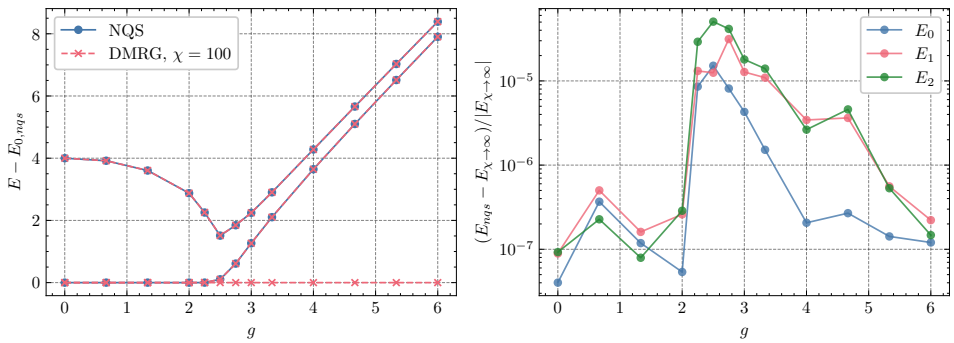

Figure 5.5: **Left:** Dependence of the three lowest energy levels (relative to the ground state found with NQS) on g for TFIM on an 8×8 lattice, computed using NQS (blue) and DMRG (red). The two methods are visually indistinguishable on this scale and show the expected closing and reopening of the low-energy gap near the critical region. **Right:** Relative energy differences between the NQS levels and the DMRG values extrapolated to infinite bond dimension, $E_{\chi\to\infty}$, as a function of g. The discrepancies are smallest away from criticality and peak around the gap minimum, where the low-energy spectrum is hardest to resolve.

5.4.5 Benchmarking on the J_1–J_2 Heisenberg model

As a second testbed we apply NIR to the frustrated J_1–J_2 Heisenberg model at the point of maximum frustration $J_2 = J_1/2$, introduced in Sec. 2.1.2. We search for the ground state on the 10×10 lattice with periodic boundary conditions. Unlike the TFIM experiments above, this is a single-state calculation with $K = 1$, so the sampling target is the ordinary Born distribution of one NQS rather than the determinant distribution over K configurations. This benchmark is widely used in the NQS literature and is challenging because frustration induces a rugged optimization landscape and nontrivial correlations.

For this experiment, we use a residual convolutional NQS architecture [72]. The input is first mapped to 16 channels by a single convolutional layer, followed by four residual blocks, each comprised of two convolutional layers with an identity skip connection. After global average pooling, a final linear layer outputs two values interpreted as the real and imaginary parts of $\log \psi(\boldsymbol{s})$. All convolutions use 3×3 kernels with circular padding, preserve spatial dimensions, and are followed by layer normalization [76] and GELU activations [65].

Due to circular padding and global pooling, the network is translationally invariant. We enforce spin-inversion symmetry by flipping all spins if the first spin is down. Rotation and reflection symmetries are enforced by applying the corresponding transformations to sampled configurations and averaging the network outputs. We also restrict the Hilbert space to the zero-magnetization sector ($S^z_{\text{tot}} = 0$, equal numbers of up and down spins) by subtracting 30 from the real part of $\log \psi(\boldsymbol{s})$ when a configuration has non-zero magnetization. The precise value is not special; it only needs to be large enough to give magnetization-violating configurations negligible Born weight relative to the legal sector. This effectively suppresses sampling of invalid states even though the SNN itself does not enforce the constraint.

The SNN architecture and adaptive NIR scheme are the same as in the TFIM experiments. We do not explicitly enforce symmetries in the SNN. In principle one could symmetrize by applying random symmetry-group transformations to sampled configurations, but we found this unnecessary for stable optimization and omitted it for simplicity.

We perform multiple runs varying the SNN size. Table 5.2 summarizes the embedding dimension d_{model}, the number of Transformer layers n_{layer}, the parameter count N_ϕ of the SNN, the final variational energy per site, and the average sampling efficiency. The smallest two SNNs were not computationally practical: their sampling efficiency was too low, indicating insufficient overlap

with the NQS distribution. Among the remaining runs, the final energies are nearly identical even though the SNN approximation quality differs, because the adaptive scheme compensates for reduced overlap by proposing more samples.

d_{model}	n_{layer}	N_ϕ	E/N	$\text{ESS}/n_{\text{proposed}}$
16	2	7938	-	0.01
16	3	11154	-	0.03
16	4	14370	-0.4969	0.10
32	2	30210	-0.4966	0.15
32	3	42786	-0.4968	0.21
32	4	55362	-0.4969	0.28

Table 5.2: Final variational energies and sampling efficiencies for different sizes of the SNN proposal network on the 10×10 J_1–J_2 model at $J_2/J_1 = 1/2$ (d_{model}: embedding dimension, n_{layer}: number of Transformer layers, N_ϕ: number of SNN parameters).

For reference, some of the lowest energies per site reported for the 10×10 J_1–J_2 model at maximum frustration are -0.497627 [34], -0.497629 [33], and -0.497634 [35] (see Table 1 in [35] for a broader comparison). Our results are slightly higher. The goal of this experiment is not to set a new state of the art, but to demonstrate that NIR remains practical for a highly frustrated model where sampling can be challenging and where the NQS itself incorporates nontrivial symmetry constraints.

5.5 Conclusions

This chapter introduced neural importance resampling, the ansatz-independent sampling strategy summarized in Statement 3, which combines importance resampling with a learned autoregressive proposal network. By outsourcing sampling to a dedicated sampling neural network, NIR avoids Markov-chain autocorrelation and serial dependence between successive samples, as well as the tuning burden of MCMC, while leaving the NQS architecture unconstrained, in contrast to fully autoregressive NQS approaches. The method is particularly useful in settings where exact autoregressive sampling is unavailable, such as the determinant multi-state NQS construction of [89].

Through numerical experiments we showed that NIR enables stable VMC optimization in TFIM regimes where MCMC sampling becomes unreliable, and that it yields energies competitive with DMRG across a broad parameter

range. On the frustrated J_1–J_2 benchmark, adaptive retraining allows NIR to remain viable even when the proposal network is relatively small, trading off proposal count against overlap without breaking the optimization loop. Its practical failure mode is nevertheless clear: if no SNN architecture that is cheap enough to train and sample from can maintain sufficient overlap with the NQS distribution, the required number of proposals becomes too large and NIR loses its computational advantage. Overall, these results suggest that NIR is a practical alternative sampler for modern NQS pipelines and a promising ingredient for multi-state variational methods.

6 The toric code as a case study for neural quantum states

6.1 Introduction

The toric code was introduced in Sec. 2.1.2 as an exactly solvable model with topology-dependent ground-state degeneracy, anyonic excitations, and a direct interpretation as a quantum error-correcting code. These properties make it an interesting case study for NQS. At the Hamiltonian level the model is local and sparse, so local-energy evaluation is straightforward. At the wave-function level, however, the relevant structure is highly nonlocal: ground-state configurations obey closed-loop constraints, the degenerate sectors are distinguished by winding parities around the torus, and local observables do not identify the topological sector. This combination tests whether an NQS workflow can exploit the simple stabilizer structure of the model rather than merely treating it as another generic spin Hamiltonian.

The physical motivation for studying this model is the robustness of topological order under perturbations. For the ideal toric code, the four ground states on a torus are exactly degenerate and separated from anyonic excitations by a finite gap. Perturbations such as a uniform magnetic field break the commuting-projector structure and deform the low-energy spectrum. Topological order is expected to persist for sufficiently weak local perturbations, but it is eventually destroyed by a phase transition once the perturbation becomes large enough. Recent analytical work has studied this problem for the toric code on honeycomb and triangular lattices in a parallel field and found field-induced continuous transitions out of the topological phase [139].

In the present context, however, the main goal is not a complete numerical determination of that phase boundary. Instead, the toric code is used as a diagnostic example for NQS methods. It isolates several issues that are also relevant more broadly: how to represent degenerate low-energy subspaces, how to encode nonlocal constraints, and how to sample from distributions with disconnected or weakly connected sectors. The robustness problem supplies

a useful physical target, namely tracking the gap between the topological low-energy manifold and the first excited state, but the central lesson of this chapter concerns the inductive biases needed for NQS to handle topological stabilizer structure.

6.2 Honeycomb toric code in a uniform field

The model considered here is a toric-code Hamiltonian on a honeycomb lattice with periodic boundary conditions. The spin-$\frac{1}{2}$ degrees of freedom live on edges, as in the square-lattice construction described in Sec. 2.1.2, but the geometry of the stabilizers changes. Each vertex stabilizer involves the three edges meeting at a vertex, while each plaquette stabilizer involves the six edges surrounding a hexagonal plaquette. The Hamiltonian is

$$\hat{H} = -\frac{1}{2}\sum_{v}\hat{A}_v - \frac{1}{2}\sum_{p}\hat{B}_p - \sum_{j}\boldsymbol{h}\cdot\hat{\boldsymbol{\sigma}}_j, \tag{6.1}$$

where

$$\hat{A}_v = \prod_{j\in v}\hat{\sigma}_j^z, \qquad \hat{B}_p = \prod_{j\in p}\hat{\sigma}_j^x. \tag{6.2}$$

Here v denotes the set of edges incident on vertex v, p denotes the set of edges on the boundary of plaquette p, $\hat{\boldsymbol{\sigma}}_j = (\hat{\sigma}_j^x, \hat{\sigma}_j^y, \hat{\sigma}_j^z)$, and $\boldsymbol{h}$ is a uniform magnetic field in spin space; the stabilizer energy costs of Eq. (2.11) are set to $\Delta_v = \Delta_p = 1$, which fixes the unit of energy. At $\boldsymbol{h} = 0$, all stabilizers commute and the model is exactly solvable. The ground states satisfy $\hat{A}_v = +1$ and $\hat{B}_p = +1$ for all vertices and plaquettes, and on the torus the ground-state degeneracy is again fourfold. For $\boldsymbol{h} \neq 0$, the field terms no longer commute with all stabilizers. The model is then no longer exactly solvable by stabilizer diagonalization, and the low-energy spectrum must be obtained perturbatively or numerically.

In the numerical experiments below the field is taken along the spin z axis, $\boldsymbol{h} = (0, 0, h_z)$. This longitudinal choice is made primarily for benchmarking: a σ^z field commutes with every vertex stabilizer $\hat{A}_v$, though not with the plaquette stabilizers $\hat{B}_p$, so the $\hat{A}_v$ remain exactly conserved and the low-energy spectrum stays within the $\hat{A}_v = +1$ sector. As detailed in Sec. 6.6, this restriction shrinks the relevant Hilbert space enough to be reachable by exact diagonalization even on the 4×4 cluster, providing an exact reference against which the NQS results can be checked. A field with components transverse to z would break this

conservation and remove the exact benchmark, and is not considered here.

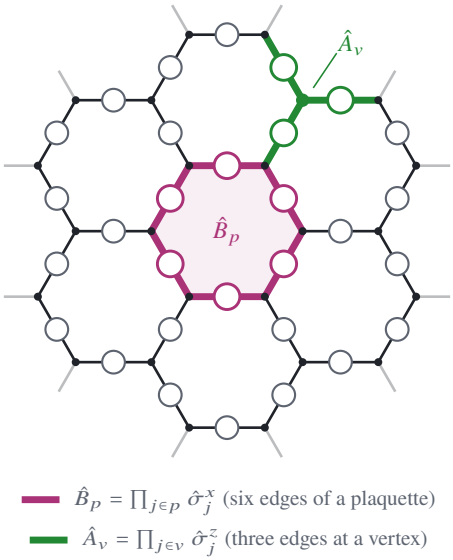

Figure 6.1: Finite honeycomb toric-code geometry used in the numerical experiments, drawn as a window onto the periodic (torus) lattice (faded dangling bonds). Spins are placed on lattice edges (circles), and one stabilizer of each type in Eq. (6.2) is highlighted in the colours of the square-lattice construction of Fig. 2.3: the plaquette operator $\hat{B}_p$ acts on the six edges surrounding a hexagon, and the vertex operator $\hat{A}_v$ on the three edges meeting at a vertex.

For NQS, this model differs from the spin and bosonic benchmarks in earlier chapters in an important way. The hard part is not simply representing short-range correlations or a complicated sign structure. Already at zero field, the exact ground states are equal-amplitude superpositions over constrained loop configurations, with the four sectors distinguished by noncontractible loop parities. These parities are global binary quantities. A neural network receiving only raw edge spins can in principle compute them, but doing so from local input coordinates is an unnecessarily indirect task, especially when the training signal comes from noisy variational estimates. The magnetic field makes the model physically richer but does not remove this basic representational issue.

6.3 Multi-state variational formulation

The topological degeneracy makes this model a natural application of the determinant multi-state NQS formulation summarized in Sec. 2.3.11. For K

target states, one introduces K single-state NQS amplitudes $\psi_a(\boldsymbol{s})$ and evaluates them on K configurations $\boldsymbol{S} = (\boldsymbol{s}_1, \ldots, \boldsymbol{s}_K)$. The total wave function on this extended system is the determinant

$$\Psi_{\text{tot}}(\boldsymbol{S}) = \det[\psi_a(\boldsymbol{s}_\ell)]_{\ell,a=1}^{K} \,. \tag{6.3}$$

The determinant suppresses collapse of different ansätze onto the same state, because linearly dependent columns make Ψ_{tot} vanish. The optimized objective is the trace of the multi-state energy matrix, and after convergence the individual energy estimates are obtained by diagonalizing that matrix.

For a toric code on a torus, the natural low-energy target has $K = 5$: four states are needed to represent the topological ground-state manifold, and the fifth state gives access to the first excitation above it. This gives the gap diagnostic

$$\Delta_{\text{TC}}(\boldsymbol{h}) = E_4(\boldsymbol{h}) - E_3(\boldsymbol{h}), \tag{6.4}$$

where the five energies are labelled in increasing order, $E_0 \leq E_1 \leq \cdots \leq E_4$, with E_0 the ground-state energy (the labelling used in Fig. 6.3), so that E_0 to E_3 span the topological manifold and E_4 is the first excitation above it. In a robustness study, this gap would be tracked as a function of field strength. In this chapter, the same setup is used mainly to probe whether the NQS can represent the full low-energy subspace without collapsing onto only part of the topological manifold.

The formulation is attractive because it avoids choosing a particular basis within the degenerate ground-state subspace. This follows directly from the trace objective: it is invariant under invertible linear mixing of the single-state ansätze (Sec. 2.3.11), so the optimization fixes only the low-energy subspace and not a basis within it. That feature is especially important here: within an exactly or nearly degenerate subspace, individual eigenvectors are not unique, whereas the subspace itself is physically meaningful. However, the same formulation substantially increases the sampling burden. The extended configuration $\boldsymbol{S}$ contains K copies of the lattice, and the sampling distribution is proportional to $|\Psi_{\text{tot}}(\boldsymbol{S})|^2$ rather than to the Born probability of a single wave function. For $K = 5$, even a modest toric-code cluster becomes a high-dimensional, strongly constrained, multi-modal sampling problem.

6.4 Practical obstacles for NQS

This case study exposes three practical obstacles. The first is the nonlocal structure of the ground-state manifold. In the σ^z basis, the low-energy sector is organized by closed loops, and the topological sectors are distinguished by loop parities around noncontractible cycles. Local neural-network features do not expose this information directly. As a result, the network must learn both a local constraint, namely that open strings with violated vertex stabilizers are energetically costly, and a global classification, namely the winding sector. These two requirements are simple to state in stabilizer language but awkward for an unstructured NQS input representation.

The second obstacle is sampling. Local Metropolis proposals can move efficiently within some regions of configuration space, but they struggle with global changes of loop structure. Changing a winding parity requires a noncontractible string operation, while local moves that violate the closed-loop constraint must pass through high-energy configurations. For the determinant multi-state ansatz this problem is amplified because one samples several coupled configurations at once. Poor mixing then manifests as unstable energy estimates and unreliable optimization.

The third obstacle is degeneracy. For ordinary ground-state VMC, small errors in the variational state can often be judged directly by the energy. Here the objective must resolve a low-energy subspace with four nearly degenerate states and a nearby excited state. If the ansatz does not represent all sectors comparably well, the determinant formulation may still lower the total energy while giving a distorted estimate of the gap in Eq. (6.4). This is why the calculation is sensitive not only to the expressive capacity of the networks, but also to the way topological-sector information enters the ansatz and sampler.

6.5 Inductive biases and sampling improvements

Four targeted modifications, each acting on a different part of the variational pipeline, proved effective for training on the small clusters studied.

The first is a soft penalty applied to the output log-amplitude. For each configuration, the number of violated vertex stabilizers $\hat{A}_v$ is computed, and $\log \psi$ is shifted by a large negative constant whenever this number exceeds a small threshold. Because the shift depends only on the configuration and not on the network parameters, it leaves the parameter gradient of $\log \psi$ unchanged,

but it reweights the variational distribution so as to suppress, without strictly forbidding, configurations far from the stabilizer-satisfying subspace. This does not solve the full topological problem, but it injects the most basic physical constraint: low-energy states of the toric code concentrate on configurations close to that subspace. For the longitudinal field studied here the vertex stabilizers are exactly conserved (Sec. 6.2), so the low-energy states lie strictly in the $\hat{A}_v = +1$ sector. The constraint being targeted is thus exact rather than heuristic, although the soft penalty imposes it only approximately: a finite log-amplitude shift strongly suppresses violating configurations without removing them entirely, so strict confinement to the $\hat{A}_v = +1$ sector would require masking those configurations outright. The network is then free to learn the residual structure inside a physically restricted region of Hilbert space rather than to rediscover the stabilizer penalties from scratch.

The second is feature augmentation. Instead of feeding only raw edge-spin values into the neural network, additional inputs are constructed from known toric-code quantities: the eigenvalues ± 1 of every vertex stabilizer, together with the parities of the two noncontractible Wilson loops wrapping the torus. The loop parities are exactly the global $\mathbb{Z}_2 \times \mathbb{Z}_2$ labels that distinguish the four topological sectors at zero field, while the vertex eigenvalues expose where a configuration contains open-string endpoints. Each of these quantities is a parity function of a subset of the raw spins, ranging from three edges meeting at a vertex to all $O(L)$ edges along a noncontractible cycle, where L is the linear size of the cluster. Parity functions are notoriously hard for shallow feedforward networks to approximate, and the difficulty grows with the number of inputs, so the loop parities in particular would be very awkward to learn from scratch. Precomputing them turns those products into features that subsequent layers can read off linearly.

The third is architectural. The augmented inputs are processed by a branched MLP: one branch handles the local stabilizer-eigenvalue features, a second branch handles the two loop parities, and a final MLP merges the two representations into $\log \psi$. Separating the branches keeps the global topological labels from being diluted by the much larger set of local stabilizer features at the input layer. The choice of nonlinearity also matters: using $\sin(x)$ in place of a monotone activation gave more stable results in some early experiments, plausibly because parity and modulo-like functions, which are central to the loop constraints, are easier to approximate with a periodic activation; this should be regarded as an empirical observation rather than a general principle.

The fourth concerns sampling. The slow mixing of local Metropolis updates identified in Sec. 6.4 (where moving between loop sectors requires noncontractible strings of flips and the intermediate moves pass through high-energy configurations) is addressed by replacing the Markov chain with the autoregressive proposal model of Chapter 5. The sampling neural network generates independent proposals that are then importance-resampled against the NQS target distribution. The motivation is the same as in the NIR setting: independent proposals from a learned distribution can place weight directly in each loop sector, rather than relying on local moves to diffuse between weakly connected regions.

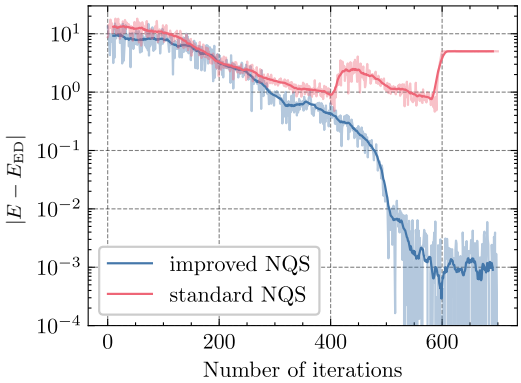

Figure 6.2: Representative training behaviour on the 4×4 honeycomb toric-code cluster, showing the absolute energy error $|E - E_{\mathrm{ED}}|$ with respect to the exact ground-state energy E_{ED} at $\boldsymbol{h} = 0$ as a function of optimization iteration. For each setup the light trace is the noisy energy estimate from the individual training batch at every iteration, while the bold trace is its running mean, shown to make the trend easier to read. The standard NQS setup (red), with raw spin inputs and local MCMC sampling, makes only limited initial progress, plateaus around an energy error of order unity, and eventually drifts upward as the optimization destabilizes; in repeated runs it usually fails to escape this regime. The improved setup (blue), which combines the soft penalty on stabilizer-violating configurations, the loop-parity and stabilizer feature augmentation, the branched architecture, and the nonlocal sampler, decreases steadily over several hundred iterations and reaches an energy error several orders of magnitude lower.

6.6 Results and interpretation

6.6.1 Standard versus improved training

The main empirical outcome is qualitative but clear. A standard NQS pipeline, using raw spin configurations and local MCMC sampling, performs poorly even for the unperturbed toric code: as illustrated by the red curve in Fig. 6.2, the energy error decreases for a few hundred iterations, becomes stuck at an error of order unity (comparable to the energy cost of creating a stabilizer defect) and is then prone to drift upward as the optimization destabilizes. This failure is notable because the zero-field Hamiltonian is exactly solvable and its ground states have an extremely simple description in stabilizer language. The difficulty therefore does not come from conventional many-body complexity alone. Rather, it comes from a mismatch between the natural variables of the problem, which are loops and stabilizers, and the generic variables supplied to the neural network and sampler.

With the improvements described above, the multi-state NQS calculation becomes reliable on the small systems tested, specifically on 2×2 and 4×4 periodic clusters (12 and 48 edge spins, respectively) in the implementation's lattice convention. The improved training curve in Fig. 6.2 shows several orders of magnitude lower energy error than the standard setup. This demonstrates that NQS can represent and optimize toric-code low-energy states when the most important topological and stabilizer information is made accessible. At the same time, scaling beyond these small systems remained difficult: on larger clusters the optimization tended to destabilize even with all four improvements active, drifting upward much as the unimproved setup does on the small clusters (Fig. 6.2). In these runs the energy decreased initially but then developed growing fluctuations and drifted upward rather than settling onto a plateau, plausibly because both the variance of the matrix-valued local energy and the difficulty of sampling the K-copy determinant distribution grow with system size and leave the stochastic optimization increasingly noisy; we did not, however, isolate a single mechanism. The dominant obstacle at larger sizes therefore appears to be training stability rather than the representational gaps that the inductive biases address. The result is therefore best viewed as a case study in necessary inductive bias rather than as a completed simulation of the field-driven phase transition.

6.6.2 Benchmark against exact diagonalization

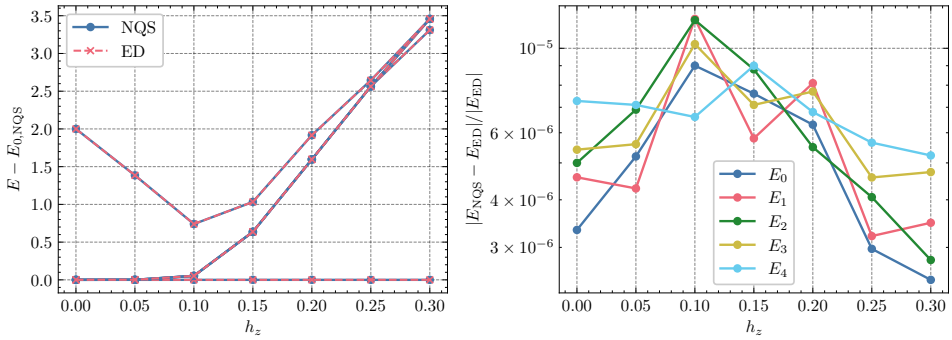

Figure 6.3: Benchmark of the converged $K = 5$ determinant multi-state NQS against ED on the 4×4 honeycomb toric-code cluster, as a function of a uniform field $\boldsymbol{h} = (0, 0, h_z)$ along the z axis in Eq. (6.1). *Left:* the five recovered energies measured from the NQS ground-state energy; the NQS (solid) and ED (dashed) curves overlap on this scale. Four states stay nearly degenerate over the whole range and form the topological manifold, while the fifth lies a finite gap above it; this gap $\Delta_{\mathrm{TC}}(\boldsymbol{h})$ of Eq. (6.4) is non-monotonic, passing through a minimum near $h_z \approx 0.1$. *Right:* the per-level relative error $|E_{n,\mathrm{NQS}} - E_{n,\mathrm{ED}}|/|E_{n,\mathrm{ED}}|$ on a logarithmic scale. It stays between roughly 10^{-6} and 2×10^{-5} across the sweep and is largest near $h_z \approx 0.1$, where the levels are closest together and hardest to separate.

Although a complete determination of the field-driven phase boundary is out of reach here, the field dependence of the low-energy spectrum can still be benchmarked against ED on the 4×4 cluster. With 48 edge spins this cluster is far too large for a brute-force diagonalization of the full 2^{48}-dimensional Hilbert space, but its low-energy spectrum is still accessible exactly by exploiting the structure of the longitudinal field $\boldsymbol{h} = (0, 0, h_z)$. As anticipated in Sec. 6.2, this field commutes with every vertex stabilizer $\hat{A}_v = \prod_{j \in v} \hat{\sigma}_j^z$, though not with the plaquette stabilizers $\hat{B}_p$, so the Hamiltonian is block-diagonal in the $\hat{A}_v$ eigenvalues and the low-energy states lie in the $\hat{A}_v = +1$ sector. In the σ^z basis this sector is the linear $\mathbb{Z}_2$ solution space of the vertex parity constraints; it is spanned by $N_{\mathrm{plaq}} + 1$ binary coordinates, where N_{plaq} is the number of plaquettes, and therefore contains only $2^{N_{\mathrm{plaq}}+1} = 2^{17} \approx 1.3 \times 10^5$ configurations. Within it the field term is diagonal while each plaquette operator $\hat{B}_p$ acts as a fixed bit flip, so the lowest few eigenpairs are obtained by a matrix-free Lanczos

iteration, and the zero-field endpoint of the sweep is fixed analytically from the commuting-stabilizer spectrum. Figure 6.3 compares the converged $K = 5$ multi-state NQS with ED over a sweep $h_z \in [0, 0.3]$. Across the whole range the NQS energies reproduce the exact ones to a relative accuracy between roughly 10^{-6} and 2×10^{-5} per level, and not only for the ground state: the ansatz resolves the four nearly degenerate states of the topological manifold and the first excited state simultaneously. The error is largest near $h_z \approx 0.1$, where the gap $\Delta_{\mathrm{TC}}(\boldsymbol{h})$ of Eq. (6.4) passes through a minimum and the near-degeneracy of the lowest levels makes them hardest to resolve. The zero-field end of this sweep is the configuration examined in more detail below through the Wilson-loop sector diagnostic. Across the whole sweep the gap stays open and the four lowest states remain nearly degenerate, so on this cluster the system does not leave the topological regime over $h_z \in [0, 0.3]$. The non-monotonic behaviour of the gap should therefore not be over-interpreted on a cluster of this size: the shallow minimum near $h_z \approx 0.1$ is most likely a finite-size level rearrangement rather than a sharp signature of the field-driven transition.

6.6.3 Ablation of the four improvements

To disentangle the contribution of each improvement, the four modifications of Sec. 6.5 are treated as binary switches and the meaningful combinations are run on both clusters. The four switches are the stabilizer-violation soft penalty (P), the stabilizer-eigenvalue and loop-parity feature augmentation (F), the branched architecture (A), and the nonlocal sampler (S), i.e. the autoregressive NIR proposal model of Chapter 5 used in place of local Metropolis updates. Although this would naively give a full $2^4 = 16$ factorial design, the architecture switch A is only defined when the augmented features F are present, since its separate branches act on the stabilizer-eigenvalue and loop-parity inputs. The tables therefore omit the four combinations with A active and F inactive. Each cell is evaluated over $N_{\mathrm{seed}} = 8$ independent seeds that differ only in the network initialization. A run is classified as *converged* if the smoothed training energy is non-increasing over the second half of the optimization and the final energy lies below a fixed threshold chosen from the baseline runs; this criterion is applied uniformly across cells and is fixed before inspecting the results. Tables 6.1 and 6.2 report the number of converged seeds together with the median and best converged energy error, quoted as the *absolute* difference $|E - E_{\mathrm{ED}}|$ from the exact ground-state energy at $\boldsymbol{h} = 0$ rather than the normalized error $|E_{\mathrm{NQS}} - E_{\mathrm{ED}}|/|E_{\mathrm{ED}}|$ plotted in Fig. 6.3. For cells in which no seed converges,

P	F	A	S	Converged	Median $\lvert E - E_{\mathrm{ED}}\rvert$	Best $\lvert E - E_{\mathrm{ED}}\rvert$
				0/8	—	—
✓				3/8	3.3×10^{-2}	5.8×10^{-3}
	✓			0/8	—	—
			✓	2/8	1.4×10^{-2}	7.0×10^{-3}
✓	✓			6/8	8.2×10^{-4}	2.5×10^{-4}
✓			✓	4/8	4.1×10^{-3}	9.0×10^{-4}
	✓	✓		0/8	—	—
	✓		✓	1/8	1.8×10^{-2}	1.1×10^{-2}
✓	✓	✓		5/8	2.9×10^{-4}	1.8×10^{-4}
✓	✓		✓	8/8	3.2×10^{-4}	1.9×10^{-4}
	✓	✓	✓	2/8	1.3×10^{-3}	6.3×10^{-4}
✓	✓	✓	✓	8/8	2.9×10^{-5}	1.4×10^{-5}

Table 6.1: Ablation of the four improvements on the 2×2 honeycomb toric-code cluster. The columns P, F, A, S indicate whether the soft penalty, feature augmentation, branched architecture, and nonlocal (autoregressive NIR) sampler are active; ✓ means active. "Converged" is the number of seeds (out of eight) that pass the prespecified stability criterion described in the text. Median and best energy errors are computed only over the converged seeds and are reported as the absolute difference $|E - E_{\mathrm{ED}}|$, where E_{ED} is the exact ground-state energy at $\boldsymbol{h} = 0$; this is not the normalized error $|E_{\mathrm{NQS}} - E_{\mathrm{ED}}|/|E_{\mathrm{ED}}|$ plotted in Fig. 6.3.

the energy columns are left blank, so that occasional outliers are not mistaken for a working configuration. This restricted grid, rather than a single staircase that adds one improvement at a time, is what makes interactions between the interventions visible: a row with two switches on can succeed even when both corresponding single-switch rows fail, which is the most direct signature of a synergistic combination.

Read together, the two tables tell a consistent story. On the 2×2 cluster the problem is forgiving: several minimal configurations already converge, namely the soft penalty alone (P), the nonlocal sampler alone (S), and the penalty combined with feature augmentation (P+F). The accuracy improves as further switches are added, reaching a median error of 2.9×10^{-5} for the full combination. On the 4×4 cluster the requirements sharpen markedly: no single switch converges any seed, and every configuration that does converge contains the nonlocal sampler S, so at the larger size S behaves as a necessary ingredient

P	F	A	S	Converged	Median $\lvert E - E_{\mathrm{ED}}\rvert$	Best $\lvert E - E_{\mathrm{ED}}\rvert$
				0/8	—	—
✓				0/8	—	—
	✓			0/8	—	—
			✓	0/8	—	—
✓	✓			0/8	—	—
✓			✓	3/8	5.5×10^{-2}	1.9×10^{-2}
	✓	✓		0/8	—	—
	✓		✓	0/8	—	—
✓	✓	✓		0/8	—	—
✓	✓		✓	8/8	1.2×10^{-2}	9.0×10^{-3}
	✓	✓	✓	5/8	2.6×10^{-2}	9.1×10^{-3}
✓	✓	✓	✓	8/8	8.7×10^{-4}	6.0×10^{-4}

Table 6.2: Ablation of the four improvements on the 4×4 honeycomb toric-code cluster, in the same format as Table 6.1. Convergence is more demanding on this cluster, so fewer single- and two-switch combinations pass the stability criterion.

rather than one improvement among several. Among the configurations that converge there, the branched architecture A then supplies the final order of magnitude in accuracy, lowering the median error from 1.2×10^{-2} (P+F+S) to 8.7×10^{-4} (P+F+A+S). The grid also exposes concrete synergies on the 4×4 cluster, where every single-switch row fails to converge a single seed: feature augmentation, branched architecture, and sampler together (F+A+S) converge with no penalty, and the penalty with the sampler (P+S) converges with no feature augmentation.

These observations are consistent with the broader message of this thesis. NQS are flexible function approximators, but their success depends strongly on whether the architecture, optimization procedure, and sampler align with the structure of the target state. In the SITE chapter, the main issue was the stability of the optimization update. In the HBHM chapter, the evidence pointed instead to a representational mismatch: strong-flux ground states remained hard for generic ansätze to approximate even when they were trained directly on exact states. In the NIR chapter, the bottleneck was sampling from sharply structured or determinant-defined distributions. The toric-code case combines aspects of all three: the target is a degenerate subspace, the relevant features are nonlocal, and the sampling distribution is difficult for local Markov chains.

6.7 Conclusions

This chapter used the honeycomb toric code in a uniform field as a diagnostic case for NQS in a topologically ordered Hilbert space, where the degrees of freedom that distinguish the low-energy sectors are nonlocal stabilizer and loop variables rather than raw spin configurations. The central finding, summarized in Statement 4, is that a standard pipeline with raw spin inputs and local Markov-chain sampling fails even at zero field, where the ground states are exactly solvable, whereas augmenting the ansatz and sampler with stabilizer-eigenvalue and loop-parity features, soft stabilizer-violation penalties, a branched architecture, and a nonlocal NIR sampler recovers the low-energy manifold on small clusters. The ablation singled out the nonlocal sampler as the ingredient that becomes necessary on the larger cluster; beyond these sizes the apparent binding constraint shifted from representation and sampling to the stability of the optimization.

The methodological lesson is that the natural variables of topological stabilizer models should be built into both the ansatz and the sampler from the beginning: stabilizer eigenvalues, loop parities, and nonlocal updates should be paired with more robust optimization if the approach is to reach beyond the small clusters studied here. Generic NQS methods are thus not immediately reliable for the toric code, but with the right physical features and sampling its topological low-energy manifold becomes accessible at the sizes reached here, which makes the chapter a technical starting point for NQS approaches to topologically ordered models.

7 Conclusions

This thesis studied NQS as variational tools for lattice many-body quantum systems. They replace an exponentially large table of wave-function amplitudes by a flexible neural-network parameterization optimized with VMC, making them attractive where established methods face limitations in frustrated, magnetic, two-dimensional, or topologically structured regimes. The results nevertheless show that flexibility alone is insufficient: success depends on whether the ansatz, optimizer, sampler, and input representation expose the relevant physical structure.

The work was organized around three practical bottlenecks: trainability, representational bias, and sampling. These bottlenecks are coupled in practice, but separating them clarifies NQS behaviour. SITE addressed trainability, the HBHM study isolated representational bias, and NIR focused on sampling; the toric-code case study combined all three in a setting where the natural variables are nonlocal stabilizer and loop degrees of freedom.

7.1 Main findings

The first contribution was SITE, an optimization method for NQS based on fitting a fixed imaginary-time-evolved target state (Chapter 3; [P1]). SITE applies an approximate imaginary-time step to a frozen copy of the current wave function and maximizes the overlap of the trainable NQS with this target. This preserves the physical direction of imaginary-time evolution while allowing standard first-order optimization in the inner loop. The adaptive time-step rule based on low-order energy moments further reduces the need for manual tuning.

The SITE experiments on the frustrated J_1–J_2 Heisenberg model showed that a fixed-target formulation can stabilize training relative to naive direct energy minimization. On the 6×6 lattice SITE was competitive with stochastic reconfiguration while keeping each update roughly five times cheaper in our implementation, since it avoids the iterative solution of the linear system that dominates SR runtimes; on larger lattices its variational energies tracked high-

bond-dimension DMRG up to 8×8 before DMRG pulled ahead. The analysis also gives a simple explanation for why direct energy-loss training is often unstable: it corresponds to the limiting case in which the target is updated after every optimizer step, making the learning signal itself drift continuously. SITE does not eliminate all difficulty in frustrated systems, and it does not replace the advantages of symmetry-aware ansätze or geometry-aware methods such as stochastic reconfiguration. Its value is more specific: alongside scalable SR variants such as minSR [34, 35], it provides a first-order route to training large generic NQS without an explicit metric-tensor inversion at every update.

The second contribution was the study of a universal performance gap for the tested generic NQS architectures in the hard-core Hofstadter–Bose–Hubbard model (Chapter 4; [P2]). This model revealed a qualitatively different failure mode. As the magnetic flux increased, generic MLP and CNN wave functions lost accuracy by several orders of magnitude. The degradation appeared not only in variational optimization with SR or SITE, but also when the NQS was trained directly against exact-diagonalization ground states, producing a systematic gap at fixed compression ratio $N_\theta/\dim\mathcal{H}$. By removing stochastic sampling and much of the optimizer dependence, these supervised experiments showed that the strong-flux regime is not merely hard to optimize: it is hard for the tested unspecialized architectures to represent compactly. At strong flux, reaching even a modest energy-accuracy threshold required more variational parameters than the Hilbert-space dimension itself, removing the very compression that motivates NQS.

Several simple explanations were tested and found insufficient. The difficulty was not primarily caused by learning a complex phase, since fitting the amplitude profile alone already became hard. It was not resolved by reducing entanglement in the strong-flux target, by changing basic input encodings, by using complex weights, by switching to a mean-squared-error loss, or by applying curriculum learning in flux. The loss landscape near trained solutions appeared locally well behaved rather than obviously rugged. The main conclusion is therefore cautious but important: strong-flux magnetic lattice systems can contain wave-function structure that is systematically mismatched to generic neural-network ansätze, even at small sizes where exact diagnostics are possible. The HBHM should consequently be treated as a useful stress test for future NQS architectures, not only as another benchmark where lower energy is the sole target.

The third contribution was NIR, a sampling strategy that decouples the NQS ansatz from the sampler (Chapter 5; [P3]). Standard VMC usually relies on

MCMC, but local Markov chains can mix poorly when the Born distribution is sharply peaked, multi-modal, constrained, or defined on an extended configuration space. NIR instead trains a separate autoregressive network to propose independent configurations and uses importance resampling to correct its mismatch with the target. The variational wave function remains unconstrained while the sampler retains exact generation and probability evaluation.

The NIR experiments showed that this approach can stabilize VMC in regimes where MCMC becomes unreliable: on the ferromagnetic 8×8 TFIM it agreed with the three lowest DMRG levels to relative differences between 10^{-7} and 10^{-5} while MCMC stalled. They also showed that NIR is naturally suited to multi-state determinant NQS, where it is not clear how to sample exactly from the full determinant distribution with an autoregressive factorization. The adaptive retraining criterion based on effective sample size provides a direct diagnostic of sampler-target mismatch and trades additional proposal generation or proposal retraining against optimization stability. The limitation is also clear: NIR is useful only if a proposal network with manageable cost can maintain sufficient overlap with the target distribution. When that overlap becomes too poor, the number of required proposals grows and the computational advantage disappears.

The toric-code case study gave a concrete example of why physical inductive bias matters (Chapter 6). At zero field, the toric code is exactly solvable and its ground states have a simple stabilizer description. Nevertheless, a standard NQS pipeline using raw spin inputs and local MCMC proposals performs poorly because the relevant structure is nonlocal: closed-loop constraints, loop parities, and topological sectors are awkward to infer from local spin coordinates alone. Feature augmentation with stabilizer eigenvalues and loop parities, soft penalties for stabilizer violations, a branched architecture, and nonlocal NIR-style sampling all improved training on small clusters, where the converged $K = 5$ ansatz reproduced the exact low-energy spectrum of the 4×4 cluster to between 10^{-6} and 2×10^{-5} per level and resolved the four near-degenerate topological states together with the first excited state. A controlled ablation singled out the nonlocal sampler as the ingredient that becomes necessary on the larger cluster. Beyond these sizes the optimization became unstable even with all four ingredients in place, so the toric code remains an open challenge for future NQS studies, with trainability becoming the next apparent bottleneck after the main representational obstacles have been addressed. The calculation did not become a complete finite-size study of the field-driven transition, but

it demonstrated the main methodological point: for topological models, the natural variables of the problem should be built into the ansatz and sampler from the beginning.

7.2 Broader lessons

Read across the four settings, the results give a concrete answer to the diagnostic question posed at the outset, namely which part of the NQS pipeline limits accuracy in a given regime. For the frustrated J_1–J_2 Heisenberg model the evidence points to trainability as the binding constraint for the generic ansatz studied here: the architecture could reach competitive energies, but the stochastic optimization had to be stabilized. For the strong-flux HBHM it was representation, since the gap survived even when exact ground states were fitted directly. For the multi-state and ferromagnetic-TFIM calculations it was sampling, where a learned, ansatz-independent sampler succeeded as local Markov chains failed. For the toric code, representation and sampling were coupled initial limitations: nonlocal features and a nonlocal sampler relieved them on small clusters, after which trainability became the apparent limitation at larger sizes. No single bottleneck dominates everywhere; the useful question, as formulated in Statement 5, is which one binds in a given regime. Figure 7.1 summarizes this reading and groups the four studies by the kind of contribution they make: methods that relieve a bottleneck, and benchmarks that expose one.

Taken together, these results support a balanced view of NQS. They combine compact wave-function parameterization, automatic differentiation, flexible architectures, and Monte Carlo estimation, and can be adapted to ground states, excited states, dynamics, and sampling tasks. At the same time, they should not be treated as architecture-independent universal solvers. Their performance depends strongly on the interaction between physical structure and machine-learning design choices. The DMRG comparisons in the contribution chapters give the same balanced picture: NQS can be competitive in some regimes, but they do not uniformly dominate tensor-network baselines, especially in the strong-flux HBHM and the largest J_1–J_2 benchmarks studied here.

One lesson is that optimization and expressivity failures must be distinguished: poor variational energy can reflect either a rugged or noisy optimization problem or an ansatz that represents the target inefficiently. SITE targets the first issue, while the HBHM results show that the second issue can persist even after optimization and sampling have been largely controlled. This distinction matters

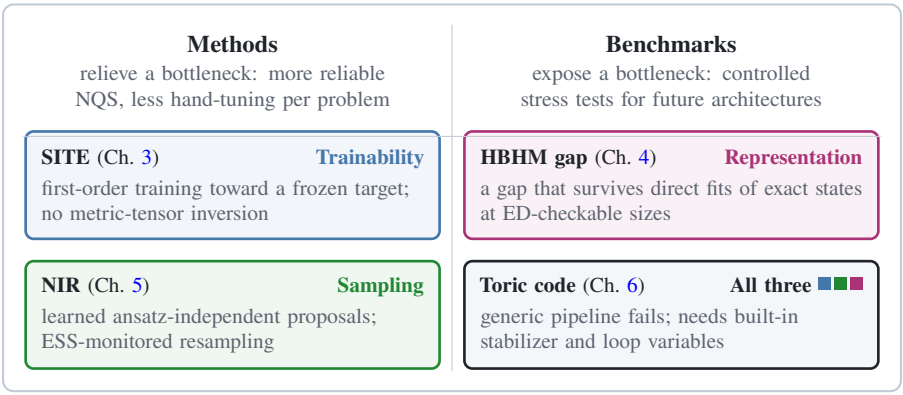

Figure 7.1: The four studies grouped by their contribution. SITE and NIR improve stages of the VMC loop, whereas the HBHM gap and toric code expose limitations of the generic pipeline. The colours follow Fig. 2.12: trainability enters through the optimizer (blue), sampling through the sampler (green), and representation through the ansatz (purple); the toric code couples all three.

for future method development: a stronger optimizer will not fix an ansatz whose inductive bias is poorly matched to the state.

A second lesson is that sampling is not a peripheral detail: it determines the estimates of energies, gradients, overlaps, and effective metrics, and poor mixing distorts the information received by the optimizer. NIR shows one way to make sampling an explicit learnable component of the pipeline, with effective sample size providing a useful measure of whether the proposal still overlaps the target. This perspective is especially relevant for excited-state and multi-state formulations, where the target distribution may not be the Born distribution of a single simple wave function.

A third lesson is that generic neural-network flexibility must be complemented by physical variables, a point made most directly by the toric-code study but present throughout the thesis. Symmetries, conservation laws, sign conventions, phase structures, stabilizer constraints, and topological sectors should not be left for a network to rediscover if they are already known. The most reliable NQS methods are likely to be hybrid: flexible enough to learn correlations that are not known analytically, but constrained or informed enough to respect the structure that is known.

7.3 Outlook

Several directions follow naturally from this work. For SITE, the most immediate extension is to combine the fixed-target imaginary-time training loop with symmetry-projected or equivariant NQS architectures. The experiments in this thesis deliberately used relatively generic ansätze in order to test trainability, but practical frustrated-spin calculations should benefit from enforcing lattice symmetries, spin quantum numbers, or sign structures directly. It would also be useful to compare SITE systematically with recent scalable variants of SR and natural-gradient optimization at matched parameter counts and computational budgets.

For the HBHM, the main open problem is representational. The observed performance gap should be probed with architectures that explicitly encode magnetic translation structure, gauge covariance, complex phase patterns, or Landau-level-inspired features. The small-system diagnostics developed here make such tests feasible because exact ground states are available and sampling noise can be removed. A useful future benchmark would not only report final energies, but also separate amplitude, phase, entanglement, and compression errors across flux and density.

For NIR, the important next step is to improve proposal models and adaptation criteria. Autoregressive transformers are flexible, but they are not always the cheapest or most physically natural proposal distributions. Normalizing-flow-inspired discrete samplers, symmetry-aware autoregressive models, or proposal networks trained with explicit coverage objectives may reduce the proposal count needed in difficult regimes. For multi-state NQS, NIR also opens a path toward larger excited-state calculations, provided that the cost of evaluating determinant targets and training the proposal network remains controlled.

For topological models, the natural direction is to move beyond the qualitative toric-code demonstration toward systematic finite-size studies. This would require architectures and samplers that treat stabilizer constraints and topological sectors as first-class objects, followed by careful comparison with perturbative, tensor-network, or series-expansion benchmarks. The broader goal is not merely to make an NQS fit the toric code, but to identify design principles for neural wave functions in phases where the important degrees of freedom are extended and nonlocal.

Recent literature already reflects these directions. SITE is discussed as fidelity-based imaginary-time training, in which an Euler step is applied to the

current state rather than projected onto the variational manifold as in SR [31]. Its 9×10^5-parameter MLP is the largest network in the compilations of Refs. [31, 35] for the 10×10 J_1–J_2 model; newer accelerated imaginary-time schemes may also transfer to this formulation [140, 141]. The HBHM gap has motivated both a physics-informed flux-attachment architecture [142] and lower-variance Monte Carlo estimators of the phase gradient, with the latter work arguing that estimator variance may explain part of the difficulty otherwise attributed to ansatz expressivity [143]. NIR is included among sampling approaches for peaked or multimodal Born distributions [144].

The overall conclusion is therefore twofold. First, NQS provide a powerful and adaptable framework for many-body quantum simulation, and the methods developed here improve the optimization and sampling parts of that framework. Second, the limitations found here are not incidental failures of implementation; they reveal where NQS methods still need more physics. Progress will likely come from treating neural networks not as black-box replacements for many-body insight, but as flexible components inside variational algorithms whose geometry, constraints, and sampling procedures are designed around the structure of the quantum state. Used in that way, with each bottleneck diagnosed rather than hidden behind a single variational energy, neural quantum states can become more reliable scientific tools for the quantum many-body problem.

Bibliography

[1] P. A. Lee, N. Nagaosa, and X.-G. Wen. "Doping a Mott insulator: Physics of high-temperature superconductivity". *Reviews of Modern Physics* 78 (2006), pp. 17–85. DOI: 10.1103/RevModPhys.78.17.

[2] H. T. Diep et al. *Frustrated spin systems*. World scientific, 2013. DOI: 10.1142/8676.

[3] L. Balents. "Spin liquids in frustrated magnets". *Nature* 464 (2010), pp. 199–208. DOI: 10.1038/nature08917.

[4] X.-G. Wen. "Colloquium: Zoo of quantum-topological phases of matter". *Reviews of Modern Physics* 89 (2017), p. 041004. DOI: 10.1103/RevModPhys.89.041004.

[5] C. Nayak, S. H. Simon, A. Stern, M. Freedman, and S. Das Sarma. "Non-Abelian anyons and topological quantum computation". *Reviews of Modern Physics* 80 (2008), pp. 1083–1159. DOI: 10.1103/RevModPhys.80.1083.

[6] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland. "Surface codes: Towards practical large-scale quantum computation". *Physical Review A* 86 (2012), p. 032324. DOI: 10.1103/PhysRevA.86.032324.

[7] C. Gross and I. Bloch. "Quantum simulations with ultracold atoms in optical lattices". *Science* 357 (2017), pp. 995–1001. DOI: 10.1126/science.aal3837.

[8] A. Browaeys and T. Lahaye. "Many-body physics with individually controlled Rydberg atoms". *Nature Physics* 16 (2020), pp. 132–142. DOI: 10.1038/s41567-019-0733-z.

[9] J. Preskill. "Quantum Computing in the NISQ era and beyond". *Quantum* 2 (2018), p. 79. DOI: 10.22331/q-2018-08-06-79.

[10] A. Auerbach. *Interacting Electrons and Quantum Magnetism*. Graduate Texts in Contemporary Physics. Springer New York, 2012. DOI: 10.1007/978-1-4612-0869-3.

[11] E. Y. Loh, J. E. Gubernatis, R. T. Scalettar, S. R. White, D. J. Scalapino, and R. L. Sugar. "Sign problem in the numerical simulation of many-electron systems". *Physical Review B* 41 (1990), pp. 9301–9307. DOI: 10.1103/PhysRevB.41.9301.

[12] S. R. White. "Density matrix formulation for quantum renormalization groups". *Physical Review Letters* 69 (1992), pp. 2863–2866. DOI: 10.1103/PhysRevLett.69.2863.

[13] P. Silvi et al. "The Tensor Networks Anthology: Simulation techniques for many-body quantum lattice systems". *SciPost Physics Lecture Notes* (2019), p. 8. DOI: 10.21468/SciPostPhysLectNotes.8.

[14] U. Schollwöck. "The density-matrix renormalization group in the age of matrix product states". *Annals of Physics* 326 (2011), pp. 96–192. DOI: 10.1016/j.aop.2010.09.012.

[15] J. Eisert, M. Cramer, and M. B. Plenio. "Colloquium: Area laws for the entanglement entropy". *Reviews of Modern Physics* 82 (2010), pp. 277–306. DOI: 10.1103/RevModPhys.82.277.

[16] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. "Learning representations by back-propagating errors". *Nature* 323 (1986), pp. 533–536. DOI: 10.1038/323533a0.

[17] G. E. Hinton, S. Osindero, and Y.-W. Teh. "A Fast Learning Algorithm for Deep Belief Nets". *Neural Computation* 18 (2006), pp. 1527–1554. DOI: 10.1162/neco.2006.18.7.1527.

[18] X. Glorot and Y. Bengio. "Understanding the difficulty of training deep feedforward neural networks". In: *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics* (May 13–15, 2010). Ed. by Y. W. Teh and M. Titterington. Vol. 9. Proceedings of Machine Learning Research. Chia Laguna Resort, Sardinia, Italy: PMLR, 2010, pp. 249–256.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton. "ImageNet Classification with Deep Convolutional Neural Networks". In: *Advances in Neural Information Processing Systems*. Ed. by F. Pereira, C. Burges, L. Bottou, and K. Weinberger. Vol. 25. Curran Associates, Inc., 2012.

[20] J. Jumper et al. "Highly accurate protein structure prediction with AlphaFold". *Nature* 596 (2021), pp. 583–589. DOI: 10.1038/s41586-021-03819-2.

[21] G. Carleo and M. Troyer. "Solving the quantum many-body problem with artificial neural networks". *Science* 355 (2017), pp. 602–606. DOI: 10.1126/science.aag2302.

[22] D. H. Ackley, G. E. Hinton, and T. J. Sejnowski. "A learning algorithm for boltzmann machines". *Cognitive Science* 9 (1985), pp. 147–169. DOI: 10.1016/S0364-0213(85)80012-4.

[23] P. Smolensky. "Information Processing in Dynamical Systems: Foundations of Harmony Theory". In: *Parallel Distributed Processing, Volume 1: Explorations in the Microstructure of Cognition: Foundations*. The MIT Press, 1986, pp. 194–281. DOI: 10.7551/mitpress/5236.003.0009.

[24] W. M. C. Foulkes, L. Mitas, R. J. Needs, and G. Rajagopal. "Quantum Monte Carlo simulations of solids". *Reviews of Modern Physics* 73 (2001), pp. 33–83. DOI: 10.1103/RevModPhys.73.33.

[25] F. Becca and S. Sorella. *Quantum Monte Carlo Approaches for Correlated Systems*. Cambridge University Press, 2017. DOI: 10.1017/9781316417041.

[26] I. Lagaris, A. Likas, and D. Fotiadis. "Artificial neural network methods in quantum mechanics". *Computer Physics Communications* 104 (1997), pp. 1–14. DOI: 10.1016/S0010-4655(97)00054-4.

[27] S. Sorella. "Green Function Monte Carlo with Stochastic Reconfiguration". *Physical Review Letters* 80 (1998), pp. 4558–4561. DOI: 10.1103/PhysRevLett.80.4558.

[28] S. Sorella. "Generalized Lanczos algorithm for variational quantum Monte Carlo". *Physical Review B* 64 (2001), p. 024512. DOI: 10.1103/PhysRevB.64.024512.

[29] S. Sorella. "Wave function optimization in the variational Monte Carlo method". *Physical Review B* 71 (2005), p. 241103. DOI: 10.1103/PhysRevB.71.241103.

[30] J. Carrasquilla. "Machine learning for quantum matter". *Advances in Physics: X* 5 (2020), p. 1797528. DOI: 10.1080/23746149.2020.1797528.

[31] H. Lange, A. V. de Walle, A. Abedinnia, and A. Bohrdt. "From architectures to applications: a review of neural quantum states". *Quantum Science and Technology* 9 (2024), p. 040501. DOI: 10.1088/2058-9565/ad7168.

[32] M. Medvidović and J. Robledo Moreno. "Neural-network quantum states for many-body physics". *The European Physical Journal Plus* 139 (2024), p. 631. DOI: 10.1140/epjp/s13360-024-05311-y.

[33] Y. Nomura and M. Imada. "Dirac-Type Nodal Spin Liquid Revealed by Refined Quantum Many-Body Solver Using Neural-Network Wave Function, Correlation Ratio, and Level Spectroscopy". *Physical Review X* 11 (2021), p. 031034. DOI: 10.1103/PhysRevX.11.031034.

[34] A. Chen and M. Heyl. "Empowering deep neural quantum states through efficient optimization". *Nature Physics* 20 (2024), pp. 1476–1481. DOI: 10.1038/s41567-024-02566-1.

[35] R. Rende, L. L. Viteritti, L. Bardone, F. Becca, and S. Goldt. "A simple linear algebra identity to optimize large-scale neural network quantum states". *Communications Physics* 7 (2024), p. 260. DOI: 10.1038/s42005-024-01732-4.

[36] S. Bravyi, D. P. Divincenzo, R. Oliveira, and B. M. Terhal. "The complexity of stoquastic local Hamiltonian problems". *Quantum Information and Computation* 8 (2008), pp. 361–385. DOI: 10.26421/QIC8.5-1.

[37] M. B. Hastings. "An area law for one-dimensional quantum systems". *Journal of Statistical Mechanics: Theory and Experiment* 2007 (2007), P08024. DOI: 10.1088/1742-5468/2007/08/P08024.

[38] S. Sachdev. *Quantum Phase Transitions*. 2nd ed. Cambridge University Press, 2011. DOI: 10.1017/CBO9780511973765.

[39] X. Li, H. Yu, F. Lou, J. Feng, M.-H. Whangbo, and H. Xiang. "Spin Hamiltonians in Magnets: Theories and Computations". *Molecules* 26 (2021), p. 803. DOI: 10.3390/molecules26040803.

[40] H. W. J. Blöte and Y. Deng. "Cluster Monte Carlo simulation of the transverse Ising model". *Physical Review E* 66 (2002), p. 066110. DOI: 10.1103/PhysRevE.66.066110.

[41] P. Chandra, P. Coleman, and A. I. Larkin. "Ising transition in frustrated Heisenberg models". *Physical Review Letters* 64 (1990), pp. 88–91. DOI: 10.1103/PhysRevLett.64.88.

[42] R. Darradi, O. Derzhko, R. Zinke, J. Schulenburg, S. E. Krüger, and J. Richter. "Ground state phases of the spin-1/2 J1-J2 Heisenberg antiferromagnet on the square lattice: A high-order coupled cluster treatment". *Physical Review B* 78 (2008), p. 214415. DOI: 10.1103/PhysRevB.78.214415.

[43] J. Richter and J. Schulenburg. "The spin-1/2 J1–J2 Heisenberg antiferromagnet on the square lattice: Exact diagonalization for N=40 spins". *The European Physical Journal B* 73 (2009), pp. 117–124. DOI: 10.1140/epjb/e2009-00400-4.

[44] S.-S. Gong, W. Zhu, D. N. Sheng, O. I. Motrunich, and M. P. A. Fisher. "Plaquette Ordered Phase and Quantum Phase Diagram in the Spin-$\frac{1}{2}$ J_1–J_2 Square Heisenberg Model". *Physical Review Letters* 113 (2014), p. 027201. DOI: 10.1103/PhysRevLett.113.027201.

[45] S. Morita, R. Kaneko, and M. Imada. "Quantum Spin Liquid in Spin 1/2 J1-J2 Heisenberg Model on Square Lattice: Many-Variable Variational Monte Carlo Study Combined with Quantum-Number Projections". *Journal of the Physical Society of Japan* 84 (2014), p. 024720. DOI: 10.7566/JPSJ.84.024720.

[46] X. Qian and M. Qin. "Absence of spin liquid phase in the $J_1 - J_2$ Heisenberg model on the square lattice". *Physical Review B* 109 (2024), p. L161103. DOI: 10.1103/PhysRevB.109.L161103.

[47] W. Marshall. "Antiferromagnetism". *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences* 232 (1955), pp. 48–68. DOI: 10.1098/rspa.1955.0200.

[48] A. Kitaev. "Fault-tolerant quantum computation by anyons". *Annals of Physics* 303 (2003), pp. 2–30. DOI: 10.1016/S0003-4916(02)00018-0.

[49] S. H. Simon. *Topological quantum*. Oxford University Press, 2023. DOI: 10.1093/oso/9780198886723.001.0001.

[50] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill. "Topological quantum memory". *Journal of Mathematical Physics* 43 (2002), pp. 4452–4505. DOI: 10.1063/1.1499754.

[51] A. Altland and B. D. Simons. *Condensed Matter Field Theory*. 2nd ed. Cambridge University Press, 2010. DOI: 10.1017/CBO9780511789984.

[52] S. H. Simon. *The Oxford solid state basics*. OUP Oxford, 2013.

[53] J. Hubbard. "Electron correlations in narrow energy bands". *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences* 276 (1963), pp. 238–257. DOI: 10.1098/rspa.1963.0204.

[54] H. A. Gersch and G. C. Knollman. "Quantum Cell Model for Bosons". *Physical Review* 129 (1963), pp. 959–967. DOI: 10.1103/PhysRev.129.959.

[55] M. Fisher, P. B. Weichman, G. Grinstein, and D. S. Fisher. "Boson localization and the superfluid-insulator transition". *Physical Review B* 40 (1989), pp. 546–570. DOI: 10.1103/PhysRevB.40.546.

[56] I. Bloch, J. Dalibard, and W. Zwerger. "Many-body physics with ultracold gases". *Reviews of Modern Physics* 80 (2008), pp. 885–964. DOI: 10.1103/RevModPhys.80.885.

[57] R. Peierls. "Zur theorie des diamagnetismus von leitungselektronen". *Zeitschrift für Physik* 80 (1933), pp. 763–791. DOI: 10.1007/BF01342591.

[58] D. R. Hofstadter. "Energy levels and wave functions of Bloch electrons in rational and irrational magnetic fields". *Physical Review B* 14 (1976), pp. 2239–2249. DOI: 10.1103/PhysRevB.14.2239.

[59] E. J. Bergholtz and Z. Liu. "Topological flat band model and fraction Chern insulators". *International Journal of Modern Physics B* 27 (2013), p. 1330017. DOI: 10.1142/s021797921330017x.

[60] S. A. Parameswaran, R. Roy, and S. L. Sondhi. "Fractional quantum Hall physics in topological flat bands". *Comptes Rendus Physique* 14 (2013), pp. 816–839. DOI: 10.1016/j.crhy.2013.04.003.

[61] B. Wang, X.-Y. Dong, and A. Eckardt. "Measurable signatures of bosonic fractional Chern insulator states and their fractional excitations in a quantum-gas microscope". *SciPost Physics* 12 (2022), p. 095. DOI: 10.21468/SciPostPhys.12.3.095.

[62] M. Aidelsburger, M. Atala, M. Lohse, J. T. Barreiro, B. Paredes, and I. Bloch. "Realization of the Hofstadter Hamiltonian with Ultracold Atoms in Optical Lattices". *Physical Review Letters* 111 (2013), p. 185301. DOI: 10.1103/PhysRevLett.111.185301.

[63] Y. Bengio, I. Goodfellow, A. Courville, et al. *Deep learning*. Vol. 1. MIT press Cambridge, MA, USA, 2017.

[64] A. F. Agarap. *Deep Learning using Rectified Linear Units (ReLU)*. 2019. DOI: 10.48550/arXiv.1803.08375.

[65] D. Hendrycks and K. Gimpel. *Gaussian Error Linear Units (GELUs)*. 2023. DOI: 10.48550/arXiv.1606.08415.

[66] P. Ramachandran, B. Zoph, and Q. V. Le. *Searching for Activation Functions*. 2017. DOI: 10.48550/arXiv.1710.05941.

[67] G. Cybenko. "Approximation by superpositions of a sigmoidal function". *Mathematics of Control, Signals, and Systems* 2 (1989), pp. 303–314. DOI: 10.1007/BF02551274.

[68] K. Hornik. "Approximation capabilities of multilayer feedforward networks". *Neural Networks* 4 (1991), pp. 251–257. DOI: 10.1016/0893-6080(91)90009-T.

[69] J. Bradbury et al. *JAX: composable transformations of Python+NumPy programs*. Version 0.3.13. 2018.

[70] K. He, X. Zhang, S. Ren, and J. Sun. "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification". In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2015, pp. 1026–1034. DOI: 10.1109/ICCV.2015.123.

[71] O. Sharir, Y. Levine, N. Wies, G. Carleo, and A. Shashua. "Deep Autoregressive Models for the Efficient Variational Simulation of Many-Body Quantum Systems". *Physical Review Letters* 124 (2020), p. 020503. DOI: 10.1103/PhysRevLett.124.020503.

[72] K. He, X. Zhang, S. Ren, and J. Sun. "Deep Residual Learning for Image Recognition". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90.

[73] T. Cohen and M. Welling. "Group equivariant convolutional networks". In: *International Conference on Machine Learning*. PMLR. 2016, pp. 2990–2999. DOI: 10.48550/arXiv.1602.07576.

[74] A. Vaswani et al. "Attention is all you need". In: *Advances in Neural Information Processing Systems*. Vol. 30. Curran Associates, Inc., 2017. DOI: 10.48550/arXiv.1706.03762.

[75] R. Xiong et al. "On layer normalization in the transformer architecture". In: *International Conference on Machine Learning*. PMLR. 2020, pp. 10524–10533. DOI: 10.48550/arXiv.2002.04745.

[76] J. L. Ba, J. R. Kiros, and G. E. Hinton. *Layer Normalization*. 2016. DOI: 10.48550/arXiv.1607.06450.

[77] P. Shaw, J. Uszkoreit, and A. Vaswani. "Self-Attention with Relative Position Representations". In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*. Association for Computational Linguistics, 2018, pp. 464–468. DOI: 10.18653/v1/N18-2074.

[78] A. Katharopoulos, A. Vyas, N. Pappas, and F. Fleuret. *Transformers are RNNs: Fast Autoregressive Transformers with Linear Attention*. 2020. DOI: 10.48550/arXiv.2006.16236.

[79] J. L. Elman. "Finding structure in time". *Cognitive Science* 14 (1990), pp. 179–211. DOI: 10.1016/0364-0213(90)90002-E.

[80] M. Hibat-Allah, M. Ganahl, L. E. Hayward, R. G. Melko, and J. Carrasquilla. "Recurrent neural network wave functions". *Physical Review Research* 2 (2020), p. 023358. DOI: 10.1103/PhysRevResearch.2.023358.

[81] S. Hochreiter and J. Schmidhuber. "Long Short-Term Memory". *Neural Computation* 9 (1997), pp. 1735–1780. DOI: 10.1162/neco.1997.9.8.1735.

[82] I. Sutskever, J. Martens, G. Dahl, and G. Hinton. "On the importance of initialization and momentum in deep learning". In: *International Conference on Machine Learning*. PMLR. 2013, pp. 1139–1147.

[83] D. P. Kingma and J. Ba. *Adam: A Method for Stochastic Optimization*. 2017. DOI: 10.48550/arXiv.1412.6980.

[84] S. L. Braunstein and C. M. Caves. "Statistical distance and the geometry of quantum states". *Physical Review Letters* 72 (1994), pp. 3439–3443. DOI: 10.1103/PhysRevLett.72.3439.

[85] S.-i. Amari. "Natural Gradient Works Efficiently in Learning". *Neural Computation* 10 (1998), pp. 251–276. DOI: 10.1162/089976698300017746.

[86] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. "Equation of State Calculations by Fast Computing Machines". *The Journal of Chemical Physics* 21 (1953), pp. 1087–1092. DOI: 10.1063/1.1699114.

[87] W. K. Hastings. "Monte Carlo Sampling Methods Using Markov Chains and Their Applications". *Biometrika* 57 (1970), pp. 97–109. DOI: 10.1093/biomet/57.1.97.

[88] A. Malyshev, J. M. Arrazola, and A. I. Lvovsky. *Autoregressive Neural Quantum States with Quantum Number Symmetries*. 2023. DOI: 10.48550/arXiv.2310.04166.

[89] D. Pfau, S. Axelrod, H. Sutterud, I. von Glehn, and J. S. Spencer. "Accurate computation of quantum excited states with neural networks". *Science* 385 (2024), eadn0137. DOI: 10.1126/science.adn0137.

[90] D. Hendry, A. Sinibaldi, and G. Carleo. "Grassmann variational Monte Carlo with neural wave functions". *Physical Review B* (2026). arXiv:2507.10287. DOI: 10.1103/l8mr-567g.

[91] G. Van Rossum and F. L. Drake. *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace, 2009. DOI: 10.5555/1593511.

[92] I. Babuschkin et al. *The DeepMind JAX Ecosystem*. http://github.com/deepmind. 2020.

[93] J. Heek et al. *Flax: A neural network library and ecosystem for JAX*. http://github.com/google/flax. Version 0.6.6. 2023.

[94] P. Kidger and C. Garcia. *Equinox: neural networks in JAX via callable PyTrees and filtered transformations*. 2021. DOI: 10.48550/arXiv.2111.00254.

[95] G. Carleo et al. "NetKet: A machine learning toolkit for many-body quantum systems". *SoftwareX* 10 (2019), p. 100311. DOI: 10.1016/j.softx.2019.100311.

[96] F. Vicentini et al. "NetKet 3: Machine learning toolbox for many-body quantum systems". *SciPost Physics Codebases* (2022), p. 7. DOI: 10.21468/SciPostPhysCodeb.7.

[97] P. Weinberg and M. Bukov. "QuSpin: a Python package for dynamics and exact diagonalisation of quantum many body systems. Part I: spin chains". *SciPost Physics* 2 (2017), p. 003. DOI: 10.21468/SciPostPhys.2.1.003.

[98] P. Weinberg and M. Bukov. "QuSpin: a Python package for dynamics and exact diagonalisation of quantum many body systems. Part II: bosons, fermions and higher spins". *SciPost Physics* 7 (2019), p. 020. DOI: 10.21468/SciPostPhys.7.2.020.

[99] J. Hauschild and F. Pollmann. "Efficient numerical simulations with Tensor Networks: Tensor Network Python (TeNPy)". *SciPost Physics Lecture Notes* (2018). Code available from https://github.com/tenpy/tenpy, p. 5. DOI: 10.21468/SciPostPhysLectNotes.5.

[100] M. Bukov, M. Schmitt, and M. Dupont. "Learning the ground state of a non-stoquastic quantum Hamiltonian in a rugged neural network landscape". *SciPost Physics* 10 (2021), p. 147. DOI: 10.21468/SciPostPhys.10.6.147.

[101] D. Kochkov and B. K. Clark. *Variational optimization in the AI era: Computational Graph States and Supervised Wave-function Optimization*. 2018. DOI: 10.48550/arXiv.1811.12423.

[102] C. Giuliani, F. Vicentini, R. Rossi, and G. Carleo. "Learning ground states of gapped quantum Hamiltonians with Kernel Methods". *Quantum* 7 (2023), p. 1096. DOI: 10.22331/q-2023-08-29-1096.

[103] H. Atanasova, L. Bernheimer, and G. Cohen. "Stochastic representation of many-body quantum states". *Nature Communications* 14 (2023), p. 3601. DOI: 10.1038/s41467-023-39244-4.

[104] B. Jónsson, B. Bauer, and G. Carleo. *Neural-network states for the classical simulation of quantum computing*. 2018. DOI: 10.48550/arXiv.1808.05232.

[105] M. Medvidović and G. Carleo. "Classical variational simulation of the Quantum Approximate Optimization Algorithm". *npj Quantum Information* 7 (2021), p. 101. DOI: 10.1038/s41534-021-00440-z.

[106] A. Sinibaldi, C. Giuliani, G. Carleo, and F. Vicentini. "Unbiasing time-dependent Variational Monte Carlo by projected quantum evolution". *Quantum* 7 (2023), p. 1131. DOI: 10.22331/q-2023-10-10-1131.

[107] V. Mnih et al. "Human-level control through deep reinforcement learning". *Nature* 518 (2015), pp. 529–533. DOI: 10.1038/nature14236.

[108] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath. "Deep Reinforcement Learning: A Brief Survey". *IEEE Signal Processing Magazine* 34 (2017), pp. 26–38. DOI: 10.1109/MSP.2017.2743240.

[109] A. Dosovitskiy et al. *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale*. 2020. DOI: 10.48550/arXiv.2010.11929.

[110] K. Choo, T. Neupert, and G. Carleo. "Two-dimensional frustrated J_1-J_2 model studied with neural network quantum states". *Physical Review B* 100 (2019), p. 125124. DOI: 10.1103/PhysRevB.100.125124.

[111] F. Ferrari, F. Becca, and J. Carrasquilla. "Neural Gutzwiller-projected variational wave functions". *Physical Review B* 100 (2019), p. 125131. DOI: 10.1103/PhysRevB.100.125131.

[112] A. Szabó and C. Castelnovo. "Neural network wave functions and the sign problem". *Physical Review Research* 2 (2020), p. 033075. DOI: 10.1103/PhysRevResearch.2.033075.

[113] X. Liang, W.-Y. Liu, P.-Z. Lin, G.-C. Guo, Y.-S. Zhang, and L. He. "Solving frustrated quantum many-particle models with convolutional neural networks". *Physical Review B* 98 (2018), p. 104426. DOI: 10.1103/PhysRevB.98.104426.

[114] M. Reh, M. Schmitt, and M. Gärttner. "Optimizing design choices for neural quantum states". *Physical Review B* 107 (2023), p. 195115. DOI: 10.1103/PhysRevB.107.195115.

[115] E. M. Inack, S. Morawetz, and R. G. Melko. "Neural annealing and visualization of autoregressive neural networks in the Newman–Moore model". *Condensed Matter* 7 (2022), p. 38. DOI: 10.3390/condmat7020038.

[116] F. Döschl, F. A. Palm, H. Lange, F. Grusdt, and A. Bohrdt. "Neural network quantum states for the interacting Hofstadter model with higher local occupations and long-range interactions". *Physical Review B* 111 (2025), p. 045408. DOI: 10.1103/PhysRevB.111.045408.

[117] K. Çeven, M. Ö. Oktel, and A. Kele ş. "Neural-network quantum states for a two-leg Bose-Hubbard ladder under magnetic flux". *Physical Review A* 106 (2022), p. 063320. DOI: 10.1103/PhysRevA.106.063320.

[118] B. Andrews and G. Möller. "Stability of fractional Chern insulators in the effective continuum limit of Harper-Hofstadter bands with Chern number| C|> 1". *Physical Review B* 97 (2018), p. 035159. DOI: 10.1103/PhysRevB.97.035159.

[119] B. Andrews, M. Mohan, and T. Neupert. "Abelian topological order of ν= 2/5 and 3/7 fractional quantum Hall states in lattice models". *Physical Review B* 103 (2021), p. 075132. DOI: 10.1103/PhysRevB.103.075132.

[120] H. Li, Z. Xu, G. Taylor, C. Studer, and T. Goldstein. *Visualizing the Loss Landscape of Neural Nets*. 2017. DOI: 10.48550/arXiv.1712.09913.

[121] A. Trockman and J. Zico Kolter. *Patches Are All You Need?* 2022. DOI: 10.48550/arXiv.2201.09792.

[122] P. Soviany, R. T. Ionescu, P. Rota, and N. Sebe. "Curriculum Learning: A Survey". *International Journal of Computer Vision* 130 (2022), pp. 1526–1565. DOI: 10.1007/s11263-022-01611-x.

[123] D. R. Vivas, J. Madroñero, V. Bucheli, L. O. Gómez, and J. H. Reina. *Neural-Network Quantum States: A Systematic Review*. 2022. DOI: 10.48550/arXiv.2204.12966.

[124] G. O. Roberts and J. S. Rosenthal. "Coupling and ergodicity of adaptive Markov chain Monte Carlo algorithms". *Journal of Applied Probability* 44 (2007), pp. 458–475. DOI: 10.1239/jap/1183667414.

[125] C. Andrieu and J. Thoms. "A tutorial on adaptive MCMC". *Statistics and Computing* 18 (2008), pp. 343–373. DOI: 10.1007/s11222-008-9110-y.

[126] D. B. Rubin. "The calculation of posterior distributions by data augmentation: Comment: A noniterative sampling/importance resampling alternative to the data augmentation algorithm for creating a few imputations when fractions of missing information are modest: The SIR algorithm". *Journal of the American Statistical Association* 82 (1987), pp. 543–546. DOI: 10.2307/2289460.

[127] M. A. K. Raiaan et al. "A Review on Large Language Models: Architectures, Applications, Taxonomies, Open Issues and Challenges". *IEEE Access* 12 (2024), pp. 26839–26874. DOI: 10.1109/ACCESS.2024.3365742.

[128] S. S. Gu, Z. Ghahramani, and R. E. Turner. "Neural adaptive sequential Monte Carlo". In: *Advances in Neural Information Processing Systems*. Vol. 28. Curran Associates, Inc., 2015. DOI: 10.48550/arXiv.1506.03338.

[129] T. A. Le, M. Igl, T. Rainforth, T. Jin, and F. Wood. *Auto-encoding sequential monte carlo*. 2017. DOI: 10.48550/arXiv.1705.10306.

[130] T. Müller, B. McWilliams, F. Rousselle, M. Gross, and J. Novák. "Neural importance sampling". *ACM Transactions on Graphics* 38 (2019), pp. 1–19. DOI: 10.1145/3341156.

[131] S. Pina-Otey, F. Sánchez, T. Lux, and V. Gaitan. "Exhaustive neural importance sampling applied to Monte Carlo event generation". *Physical Review D* 102 (2020), p. 013003. DOI: 10.1103/PhysRevD.102.013003.

[132] K. A. Nicoli, S. Nakajima, N. Strodthoff, W. Samek, K.-R. Müller, and P. Kessel. "Asymptotically unbiased estimation of physical observables with neural samplers". *Physical Review E* 101 (2020), p. 023304. DOI: 10.1103/PhysRevE.101.023304.

[133] B. McNaughton, M. V. Milo ševi ć, A. Perali, and S. Pilati. "Boosting Monte Carlo simulations of spin glasses using autoregressive neural networks". *Physical Review E* 101 (2020), p. 053312. DOI: 10.1103/PhysRevE.101.053312.

[134] M. Gabrié, G. M. Rotskoff, and E. Vanden-Eijnden. "Adaptive Monte Carlo augmented with normalizing flows". *Proceedings of the National Academy of Sciences* 119 (2022), e2109420119. DOI: 10.1073/pnas.2109420119.

[135] L. Grenioux, É. Moulines, and M. Gabrié. *Balanced Training of Energy-Based Models with Adaptive Flow Sampling*. 2023. DOI: 10.48550/arXiv.2306.00684.

[136] L. M. Del Bono, F. Ricci-Tersenghi, and F. Zamponi. "Performance of machine-learning-assisted Monte Carlo in sampling from simple statistical physics models". *Physical Review E* 112 (2025), p. 045307. DOI: 10.1103/s1rm-29zx.

[137] M. Schmitt, M. M. Rams, J. Dziarmaga, M. Heyl, and W. H. Zurek. "Quantum phase transition dynamics in the two-dimensional transverse-field Ising model". *Science Advances* 8 (2022), eabl6850. DOI: 10.1126/sciadv.abl6850.

[138] R. Orús. "Tensor networks for complex quantum systems". *Nature Reviews Physics* 1 (2019), pp. 538–550. DOI: 10.1038/s42254-019-0086-7.

[139] V. Kott, M. Mühlhauser, J. A. Koziol, and K. P. Schmidt. "Quantum robustness of the toric code in a parallel field on the honeycomb and triangular lattice". *SciPost Physics* 17 (2024), p. 053. DOI: 10.21468/SciPostPhys.17.2.053.

[140] J. P. Zima, E. M. Stoudenmire, S. R. White, O. Parcollet, and J. Kaye. *Fast Tensor Network Imaginary Time Evolution by Implicit Stepping on Logarithmic Grids*. 2026. DOI: 10.48550/arXiv.2606.02930.

[141] C. L. Benavides-Riveros, P. Sharma, and F. Šimkovic IV. *Augmenting Imaginary-Time Evolution with Local Geometric Information*. 2026. DOI: 10.48550/arXiv.2606.23934.

[142] R. Badalyan, K. G. Nazaryan, and T. A. Sedrakyan. *Neural Flux Attachment: From Bose Condensates to Chiral Topological Matter*. 2026. DOI: 10.48550/arXiv.2608.06168.

[143] Y.-R. Xue, R. Wang, B. Wang, and C. Wei. *Low-variance estimators overcome the phase-gradient bottleneck in complex-valued neural quantum states*. 2026. DOI: 10.48550/arXiv.2606.13912.

[144] S. Zhou, Y. Huang, X. Yuan, and X. Cai. *Enhanced Neural Quantum State via Annealed Gradient Descent*. 2026. DOI: 10.48550/arXiv.2607.18865.

This page is intentionally left blank.

Santrauka lietuvių kalba

Įvadas

Daugelis svarbiausių neišspręstų kondensuotosios medžiagos fizikos ir kvantinės informacijos problemų yra susijusios su stipriai koreliuotomis kvantinėmis daugiadalelėmis sistemomis. Tokių reiškinių kaip aukštatemperatūris superlaidumas, magnetizmas sistemose su konkuruojančiomis sąveikomis, kvantiniai sukinių skysčiai, trupmeninis kvantinis Holo efektas ar kitos topologinės tvarkos formos negalima paaiškinti remiantis vien pavienių dalelių savybėmis [1–6]. Šiuos reiškinius lemia kartu veikiančios kvantinė koherencija, dalelių sąveikos ir simetrijos. Gardeliniai modeliai leidžia juos tirti kontroliuojamomis sąlygomis: modelyje išlaikomi svarbiausi lokalūs žemos energijos laisvės laipsniai, o aukštos energijos detalės aprašomos tik netiesiogiai arba atmetamos. Tai nėra vien teorinis supaprastinimas. Artimus gardelinius hamiltonianus galima įgyvendinti ultrašaltų atomų sistemose, programuojamuose Rydbergo atomų masyvuose ir kvantinės informacijos įrenginiuose, todėl jų savybes galima tirti ir eksperimentiškai [7–10]. Šiame darbe nagrinėjamos kvantinės sukinių sistemos, Bose'o–Hubbardo tipo modeliai ir torinis kodas yra pakankamai paprasti, kad juos būtų galima tiksliai apibrėžti ir modeliuoti, bet kartu pakankamai turtingi, kad atskleistų esminius kvantinių daugiadalelių sistemų sunkumus.

Šioje disertacijoje neuroninės kvantinės būsenos (angl. *neural quantum states*, NQS) nagrinėjamos kaip variacinių banginių funkcijų šeima kvantinėms daugiadalelėms sistemoms aprašyti. Šių sistemų skaitinį sudėtingumą pirmiausia lemia eksponentiškai auganti Hilberto erdvės dimensija. Jei gardeliniame modelyje kiekvieno iš N lokalių laisvės laipsnių Hilberto erdvės dimensija yra d, bendrajai būsenai aprašyti reikia d^N kompleksinių amplitudžių. Todėl tikslusis diagonalizavimas tampa nepraktiškas jau palyginti mažose sistemose, ir tenka remtis apytikriais skaitiniais metodais.

Vis dėlto skaičiavimo sunkumą lemia ne vien Hilberto erdvės dydis, bet ir pačios banginės funkcijos sandara. Stipriai koreliuotose sistemose, pavyzdžiui, kvantiniuose magnetuose su konkuruojančiomis sąveikomis, magnetinio srauto

veikiamose gardelėse ar topologinės tvarkos fazėse, svarbų vaidmenį atlieka kolektyvinės koreliacijos, ženklų ir kompleksinių fazių sandara bei nelokalūs apribojimai. Kvantiniai Monte Karlo metodai labai veiksmingi, jei pavyksta rasti bazę, kurioje svoriai yra neneigiami, tačiau konkuruojančios sąveikos, fermioninė statistika ar Peierlso fazės gali sukelti ženklo arba fazės problemą [11]. Tenzorinių tinklų metodai, ypač matricinės sandaugos būsenos (angl. *matrix product states*, MPS) ir tankio matricos renormalizavimo grupė (angl. *density matrix renormalization group*, DMRG), itin sėkmingi vienmatėse ir kvazivienmatėse sistemose, tačiau jų skaičiavimo sąnaudos sparčiai auga, kai reikia išlaikyti dvimatėms sistemoms būdingą kvantinę susietį ar ilgo nuotolio struktūrą [12–14]. Todėl prasminga ieškoti šias prieigas papildančių variacinių metodų, kurie išnaudotų fizikinę būsenų sandarą ir nebūtų glaudžiai susieti su vienmate geometrija.

NQS atveju variacinė banginė funkcija parametrizuojama neuroniniu tinklu. Bet kuri banginė funkcija pasirinktoje bazėje išskleidžiama taip:

$$|\psi_{\boldsymbol{\theta}}\rangle = \sum_{\boldsymbol{s}} \psi_{\boldsymbol{\theta}}(\boldsymbol{s})|\boldsymbol{s}\rangle. \tag{S.1}$$

NQS nereikia saugoti visos amplitudžių lentelės. Neuroniniam tinklui pateikiama bazinės būsenos $|\boldsymbol{s}\rangle$ koduotė, o tinklas apskaičiuoja atitinkamą amplitudę $\psi_{\boldsymbol{\theta}}(\boldsymbol{s})$. Reikalingas parametrų skaičius N_θ paprastai auga daug lėčiau nei Hilberto erdvės dimensija, todėl NQS gali būti kompaktiškas banginės funkcijos aprašas. Šis požiūris remiasi tuo, kad žemos energijos būsenos nėra atsitiktiniai Hilberto erdvės vektoriai: jų sandarą lemia lokali hamiltoniano struktūra, simetrijos, tvermės dėsniai, koreliacijų ilgis, kvantinė susietis ir kiti fizikiniai veiksniai [15]. NQS gali sėkmingai išnaudoti šią struktūrą, jei tinkamai parenkami tinklo architektūra, nuostolio funkcija, optimizavimo metodas ir ėminių generavimo būdas.

Todėl NQS nereikėtų laikyti universaliu „juodosios dėžės“ metodu. Praktinį tikslumą lemia ne tik tinklo galimybės aprašyti būseną, bet ir optimizavimo stabilumas, pagal Borno skirstinį generuojamų ėminių kokybė, konfigūracijų kodavimas, simetrijų įtraukimas ir banginės funkcijos ženklų ar fazių sandara. Šiame darbe svarbu ne vien pasiekti mažą energijos paklaidą konkrečiuose modeliuose, bet ir nustatyti, kuri variacinės schemos dalis riboja skaičiavimo tikslumą. Todėl disertacijoje siekiama ne tik įvertinti NQS veikimą sudėtinguose fizikiniuose modeliuose, bet ir atskirti paklaidų priežastis bei pasiūlyti tikslingų patobulinimų.

Darbo tikslai, uždaviniai ir naujumas

Bendrasis darbo tikslas – padidinti NQS metodų patikimumą tiriant gardelines kvantines daugiadaleles sistemas. Tam reikia ne tik mažinti variacinės energijos paklaidą, bet ir nustatyti, kuri skaičiavimo schemos dalis konkrečiame fizikiniame režime riboja tikslumą. Disertacijoje išskiriami trys tarpusavyje susiję praktiniai sunkumai.

Pirmasis sunkumas – optimizacijos stabilumas. Net jei pasirinktas neuroninis tinklas iš principo geba aprašyti ieškomą būseną, energijos gradientai įvertinami stochastiškai, o variacinės daugdaros metrika gali būti blogai sąlygota. Tokiu atveju tiesioginis energijos minimizavimas pirmosios eilės optimizatoriais gali konverguoti lėtai, būti jautrus pradinėms sąlygoms arba tapti nestabilus.

Antrasis sunkumas – architektūros tinkamumas būsenai aprašyti. Didelis parametrų skaičius savaime negarantuoja kompaktiško ir tikslaus banginės funkcijos aprašo, jei architektūra neatitinka tikslinės būsenos ženklų, fazių, lokalumo, simetrijų ar topologinės sandaros.

Trečiasis sunkumas – ėminių generavimas pagal Borno skirstinį. Variaciniame Monte Karlo (VMC) metode stebimieji dydžiai vertinami iš konfigūracijų, pasiskirsčiusių pagal $|\psi_{\boldsymbol{\theta}}(\boldsymbol{s})|^2$. Jei Markovo grandinės maišosi lėtai arba neaplanko svarbių konfigūracijų erdvės sričių, energijos ir gradientų įverčiai tampa nepatikimi.

Šiems klausimams spręsti darbe iškelti keturi konkretūs uždaviniai.

1. Sukurti NQS optimizavimo schemą, išlaikančią menamojo laiko evoliucijos kryptį, tačiau nereikalaujančią kiekviename žingsnyje spręsti stochastinės rekonfigūracijos metrinio tenzoriaus tiesinių lygčių sistemos, ir įvertinti jos stabilumą bei tikslumą J_1–J_2 Heisenbergo modelyje su konkuruojančiomis sąveikomis.

2. Kontroliuojamame modelyje, kurį dėl nedidelio dydžio galima tiksliai diagonalizuoti, nustatyti, kada ir kodėl nespecializuotos NQS architektūros tikslinę būseną aprašo neefektyviai, atskiriant šį ribojimą nuo optimizavimo ir ėminių generavimo sunkumų.

3. Pasiūlyti metodą, leidžiantį ėmiklį parinkti nepriklausomai nuo banginę funkciją aproksimuojančio neuroninio tinklo ir tinkantį tiek tada, kai lokalios Markovo grandinės maišosi prastai, tiek tada, kai neįmanoma autoregresiniu būdu tiksliai generuoti ėminių pagal visą tikslinį skirstinį.

4. Torinio kodo pavyzdžiu parodyti, kaip į banginės funkcijos parametrizavimą ir ėminių generavimą įtraukti topologinę stabilizatorių bei kilpų struktūrą, ir nustatyti, kokią iš anksto žinomą fizikinę struktūrą būtina panaudoti norint atkurti topologinės tvarkos žemos energijos poerdvį.

Darbo naujumą lemia ne tik sukurti metodai, bet ir sisteminga NQS diagnostika, kuriai pasitelkiami skirtingi modeliai, atskleidžiantys skirtingus variacinės schemos apribojimus. Prižiūrimoji menamojo laiko evoliucija (angl. *supervised imaginary time evolution*, SITE, [P1]) menamojo laiko žingsnį paverčia prižiūrimojo mokymo uždaviniu su fiksuota tiksline bangine funkcija, todėl didelius nespecializuotus tinklus galima mokyti be tiesioginės metrinio tenzoriaus inversijos. Hofstadterio–Bose'o–Hubbardo modelio analizė ([P2]) pateikia kontroliuojamą pavyzdį, kai NQS tikslinę būseną aprašo neefektyviai: stipraus magnetinio srauto režime tikslumas labai pablogėja net tada, kai tinklas tiesiogiai mokomas pagal tiksliojo diagonalizavimo sprendinį. Neuroninis svarbos perrinkimas (angl. *neural importance resampling*, NIR, [P3]) ėminių generavimą paverčia atskiru mokomu variacinės schemos komponentu ir leidžia praktiškai įgyvendinti kelių būsenų determinantinę NQS formuluotę. Torinio kodo tyrimas sujungia šias temas topologinės tvarkos modelyje, kuriame žemos energijos sektorius apibūdina nelokalūs stabilizatorių ir kilpų kintamieji.

Ginamieji teiginiai

1. SITE menamojo laiko evoliuciją performuluoja kaip neuroninio tinklo prižiūrimąjį mokymą pagal fiksuotą tikslinę būseną. Šis metodas leidžia stabiliai optimizuoti didelius NQS tinklus pirmosios eilės metodais be metrinio tenzoriaus inversijos, ir pasiekti konkurencingą tikslumą lyginant su stochastine rekonfigūracija (SR). Jei tikslinė būsena atnaujinama po kiekvieno optimizatoriaus žingsnio, gaunamas tiesioginis energijos minimizavimas; šis ryšys paaiškina mažesnį tokio metodo stabilumą (žr. 3 skyrių).

2. Stiprus magnetinis srautas atskleidžia, kad MLP ir CNN architektūros Hofstadterio–Bose'o–Hubbardo modelio pagrindines būsenas aprašo sistemingai neefektyviai. Šiame režime paklaidos išauga keliomis eilėmis. Tikslumo nukritimas išlieka ir atsisakius VMC ėminių generavimo bei bandant skirtingus optimizavimo metodus, todėl pagrindiniu ribojančiu veiksniu laikytinas neefektyvus būsenos parametrizavimas, o pats modelis tampa kontroliuojamu iššūkiu ateities tyrimams (žr. 4 skyrių).

3. NIR leidžia ėmiklį pasirinkti nepriklausomai nuo variacinės banginės funkcijos: naudojamas mokomas autoregresinis pasiūlymų tinklas, svarbos perrinkimas ir prisitaikymas pagal efektyvųjį imties dydį. Šis metodas stabilizuoja variacinį Monte Karlo skaičiavimą, kai lokalios Markovo grandinės tampa nepatikimos, ir leidžia taikyti kelių būsenų determinantines NQS, kurių visam tiksliniam skirstiniui nėra tiksliai ėminius generuojančio metodo (žr. 5 skyrių).

4. Torinio kodo pavyzdys rodo, kad tikslus išsprendžiamumas dar negarantuoja, jog modelį pavyks ištirti įprasta NQS skaičiavimo schema: vien sukinių reikšmėmis paremta įvestis ir lokalus ėminių generavimas gali nepasiteisinti net nesant išoriniam laukui. Stabilizatorių tikrinės reikšmės ir kilpų paritetai, derinami su nelokaliu ėminių generavimu, leidžia atkurti mažų gardelių žemos energijos poerdvį. Vis dėlto didesnėse gardelėse rezultatus ima riboti mokymo nestabilumas, todėl šis modelis išklieka sudėtingas NQS metodų tikrinimo uždavinys (žr. 6 skyrių).

5. Vien variacinės energijos reikšmės nepakanka nustatyti, ar NQS rezultato tikslumą riboja optimizacijos stabilumas, būsenos parametrizavimas ar ėminių generavimas. Keturiuose tyrimuose tikslingai keičiant atskiras skaičiavimo schemos dalis parodyta, kad skirtinguose fizikiniuose režimuose gali vyrauti kiekvienas iš šių veiksnių, todėl kuriant patikimą NQS metodą juos būtina diagnozuoti atskirai (žr. 3–6 skyrius).

Bendroji metodinė schema

Disertacijoje taikoma skaičiavimų metodika remiasi variaciniu Monte Karlo metodu [24, 25]. Neuroniniu tinklu parametrizuojama variacinė banginė funkcija, o energija ir kiti stebimieji dydžiai vertinami iš konfigūracijų, sugeneruotų pagal Borno skirstinį. Šiame skyrelyje glaustai pristatomi bendri metodikos elementai, kuriais remiamasi tolesniuose skyriuose: NQS parametrizavimas, lokalieji įverčiai, energijos gradientas, menamojo laiko evoliucija ir stochastinė rekonfigūracija, ėminių generavimo strategijos, simetrijų įtraukimas ir kelių būsenų determinantinė formuluotė. Atskiri šių dalių patobulinimai aptariami vėlesniuose rezultatų skyreliuose.

Banginė funkcija parametrizuojama neuroniniu tinklu: bazinė konfigūracija $\boldsymbol{s}$ pateikiama tinklui, o jo išvestimi laikoma atitinkama (paprastai nenormuota) kompleksinė amplitudė $\psi_{\boldsymbol{\theta}}(\boldsymbol{s})$ arba jos logaritmas. Disertacijoje naudojami MLP,

CNN bei transformerio tipo autoregresiniai modeliai. Pagal parametrizavimo būdą NQS galima skirti į dvi pagrindines klases. *Neišreikštinė NQS* (angl. *implicit NQS*) iš karto grąžina visos konfigūracijos amplitudę, todėl ėminius pagal Borno skirstinį tenka generuoti netiesiogiai, paprastai Markovo grandinių Monte Karlo metodais. *Autoregresinėje NQS* banginė funkcija išskaidoma į normuotus sąlyginius daugiklius pagal pasirinktą kintamųjų eilę,

$$\psi_{\boldsymbol{\theta}}(\boldsymbol{s}) = \prod_{k=1}^{N} \psi_k(s_k \mid \boldsymbol{s}_{<k}), \qquad \boldsymbol{s}_{<k} = (s_1, \ldots, s_{k-1}), \tag{S.2}$$

todėl iš jos galima generuoti tikslius nepriklausomus ėminius pagal Borno skirstinį ir tiesiogiai apskaičiuoti jų tikimybes. Tačiau toks suskaidymas sumažina architektūros lankstumą ir apsunkina simetrijų ar kitų apribojimų įtraukimą.

Bet kurio stebimojo dydžio $\hat{A}$ vidurkį galima išreikšti per *lokalųjį įvertį*

$$A_{\mathrm{loc}}(\boldsymbol{s}) = \sum_{\boldsymbol{s}'} \frac{\psi_{\boldsymbol{\theta}}(\boldsymbol{s}')}{\psi_{\boldsymbol{\theta}}(\boldsymbol{s})} A_{\boldsymbol{s}\boldsymbol{s}'}, \qquad A_{\boldsymbol{s}\boldsymbol{s}'} = \langle \boldsymbol{s} | \hat{A} | \boldsymbol{s}' \rangle, \tag{S.3}$$

o stebimojo dydžio vidurkis gaunamas vidurkinant A_{loc} pagal Borno skirstinį, kuriame konfigūracijos $\boldsymbol{s}$ tikimybė yra

$$p_{\psi}(\boldsymbol{s}) = \frac{|\psi_{\boldsymbol{\theta}}(\boldsymbol{s})|^2}{\sum_{\boldsymbol{s}'} |\psi_{\boldsymbol{\theta}}(\boldsymbol{s}')|^2}. \tag{S.4}$$

Taigi $\langle \hat{A} \rangle = \langle A_{\mathrm{loc}}(\boldsymbol{s}) \rangle$. Energijos atveju A_{loc} yra lokalioji energija $E_{\mathrm{loc}}(\boldsymbol{s})$, o variacinis principas užtikrina

$$E(\boldsymbol{\theta}) = \langle E_{\mathrm{loc}}(\boldsymbol{s}) \rangle = \frac{\langle \psi_{\boldsymbol{\theta}} | \hat{H} | \psi_{\boldsymbol{\theta}} \rangle}{\langle \psi_{\boldsymbol{\theta}} | \psi_{\boldsymbol{\theta}} \rangle} \geq E_0, \tag{S.5}$$

kur E_0 yra tiksli pagrindinės būsenos energija. Lokalaus hamiltoniano matrica $H_{\boldsymbol{s}\boldsymbol{s}'}$ yra reta (angl. *sparse*), todėl kiekvienos konfigūracijos $\boldsymbol{s}$ lokaliąją energiją galima efektyviai apskaičiuoti net tada, kai visa Hilberto erdvė yra labai didelė.

Energijos gradientą neuroninio tinklo parametrų atžvilgiu taip pat patogu užrašyti kaip Monte Karlo vidurkį. Įvedus logaritmines išvestines

$$O_k(\boldsymbol{s}) = \frac{\partial}{\partial \theta_k} \ln \psi_{\boldsymbol{\theta}}(\boldsymbol{s}), \tag{S.6}$$

gaunama

$$\frac{\partial E}{\partial \theta_k} = 2\,\mathrm{Re}\left[\langle O_k^* E_{\mathrm{loc}}\rangle - \langle O_k^*\rangle\langle E_{\mathrm{loc}}\rangle\right]. \tag{S.7}$$

Logaritminės išvestinės skaičiuojamos automatiniu diferencijavimu. Apskaičiavus šį gradientą, optimizavimui galima taikyti įprastus pirmosios eilės metodus, pavyzdžiui, Adam optimizatorių. Kartu ši formulė aiškiai parodo, kad ėminių kokybė tiesiogiai veikia gradiento įverčio triukšmą, taigi ir optimizavimo stabilumą.

Kita svarbi šios disertacijos idėja yra menamojo laiko evoliucija

$$|\psi(\tau)\rangle = \mathrm{e}^{-\tau\hat{H}}|\psi(0)\rangle. \tag{S.8}$$

Jei pradinis vektorius turi nenulinį persidengimą su pagrindine būsena, evoliucijos metu jis asimptotiškai artėja prie pagrindinių būsenų poerdvio. Projektuojant šią evoliuciją į variacinę daugdarą, gaunamas stochastinės rekonfigūracijos (angl. *stochastic reconfiguration*, SR) metodas [27–29]. Jame įprasta Euklido metrika parametrų erdvėje pakeičiama Hilberto erdvės indukuota metrika. Kiekviename SR žingsnyje sprendžiama tiesinių lygčių sistema

$$\sum_l S_{kl}\delta\theta_l = \eta f_k, \tag{S.9}$$

kur S_{kl} yra logaritminių išvestinių kovariacijų matrica (iki įprastinio daugiklio sutampanti su kvantine Fišerio informacijos matrica), o f_k atitinka neigiamą energijos gradientą. SR dažnai pagerina mokymo stabilumą, tačiau didelių tinklų metrinio tenzoriaus sudarymas ir tiesinių lygčių sistemos sprendimas pareikalauja daug skaičiavimo išteklių, todėl metodą sunku taikyti giliems tinklams. Naujesni SR variantai, pavyzdžiui, minSR, šią problemą sušvelnina perkeldami daugiausia sąnaudų reikalaujančius tiesinės algebros skaičiavimus į ėminių erdvę, kurios matmuo yra imties dydis M, paprastai daug mažesnis už neuroninio tinklo parametrų skaičių N_θ [34, 35]. Alternatyvų sprendimą siūlo šiame darbe sukurtas SITE metodas: jame menamojo laiko žingsnis suformuluojamas kaip prižiūrimojo mokymo uždavinys, todėl metrinio tenzoriaus konstruoti ir invertuoti nereikia.

Svarbi praktinė metodikos dalis yra ėminių generavimas. Iš Borno skirstinio, kurį apibrėžia neišreikštinė NQS, ėminių negalima generuoti tiesiogiai, todėl konfigūracijos paprastai gaunamos Metropolio–Hastingso algoritmu, atliekant lokalius pokyčius, pavyzdžiui, apverčiant vieną sukinį. Šis būdas yra asimptotiškai tikslus, bet jo praktinis veiksmingumas priklauso nuo pasiūlymų schemos:

vienas po kito generuojami ėminiai yra koreliuoti, o Markovo grandinės gali maišytis lėtai, kai $|\psi_{\boldsymbol{\theta}}|^2$ yra ryškiai daugiamodis arba turi silpnai sujungtus sektorius. Autoregresinė NQS, priešingai, leidžia generuoti tikslius nepriklausomus ėminius ir tiksliai apskaičiuoti jų tikimybes. Šiame darbe siūlomas tarpinis sprendimas yra NIR metodas. Jame ėminius generuojantis tinklas parenkamas nepriklausomai nuo variacinės banginės funkcijos; metodas išsamiau aptariamas tolesniame skyrelyje.

NQS galima naudoti ne tik pagrindinei būsenai rasti, bet ir K žemiausių būsenų poerdviui aprašyti. Tam taikoma natūraliųjų sužadintųjų būsenų VMC (NES-VMC) formuluotė. Iš K atskirų NQS amplitudžių $\psi_a(\boldsymbol{s})$ ir K konfigūracijų rinkinio $\boldsymbol{S} = (\boldsymbol{s}_1, \ldots, \boldsymbol{s}_K)$ sudaroma bendroji banginė funkcija

$$\Psi_{\text{tot}}(\boldsymbol{S}) = \det[\psi_a(\boldsymbol{s}_\ell)]_{\ell,a=1}^{K} \,. \tag{S.10}$$

Determinantinė sandara neleidžia kelioms NQS aprašyti tos pačios būsenos: jei dvi ar daugiau jų sutampa, atitinkami stulpeliai tampa tiesiškai priklausomi ir Ψ_{tot} virsta nuliu, todėl nereikia atskirai įvesti ortogonalumo apribojimų. Minimizuojant išplėstinės sistemos energijos matricos pėdsaką, optimizuojamas visas žemiausių K būsenų poerdvis, o atskiros energijos atkuriamos diagonalizuojant gautąją energijos matricą. Borno skirstinys tokiu atveju yra apibrėžtas K kopijų konfigūracijų erdvėje, kuri yra eksponentiškai didesnė už vienos kopijos erdvę. Todėl ėminių generavimas tampa dar sudėtingesnis; šią problemą padeda spręsti šiame darbe sukurtas NIR metodas.

Prižiūrimoji menamojo laiko evoliucija

Šioje disertacijoje sukurtas SITE metodas menamojo laiko žingsnį suformuluoja kaip prižiūrimojo mokymo uždavinį. Kiekvienos epochos pradžioje užfiksuojama tuo metu turimų parametrų kopija $\bar{\boldsymbol{\theta}}$, iš kurios sudaroma tikslinė būsena

$$|\psi_T\rangle = \left(1 - \Delta\tau \hat{H}\right) |\psi_{\bar{\boldsymbol{\theta}}}\rangle. \tag{S.11}$$

Tada tinklas mokomas maksimizuoti normuotą persidengimą su šia fiksuota tiksline būsena, t. y. minimizuoti

$$\mathcal{L}(\boldsymbol{\theta}) = -\ln \frac{|\langle\psi_{\boldsymbol{\theta}}|\psi_T\rangle|^2}{\langle\psi_{\boldsymbol{\theta}}|\psi_{\boldsymbol{\theta}}\rangle\langle\psi_T|\psi_T\rangle}. \tag{S.12}$$

Taip išlaikoma menamojo laiko evoliucijos kryptis, tačiau kiekviename žingsnyje nereikia spręsti SR lygčių sistemos. Kadangi epochos metu tikslinė būsena nekinta, optimizatorius nuosekliai artėja prie to paties tikslo, todėl mokymas tampa stabilesnis. Jei tikslinė būsena būtų perskaičiuojama po kiekvieno optimizatoriaus žingsnio, SITE gradientas taptų proporcingas įprastam energijos gradientui. Taip galima paaiškinti, kodėl tiesioginis energijos minimizavimas gali būti nestabilus: jis atitinka ribinį SITE atvejį, kai mokymo tikslas nuolat kinta.

Laiko žingsnis $\Delta\tau$ adaptyviai parenkamas pagal žemesniųjų eilių energijos momentus. Tikslinės būsenos energija išreiškiama per $\langle E\rangle$, $\langle E^2\rangle$ ir $\langle E^3\rangle$, o ją minimizuojant $\Delta\tau$ atžvilgiu gaunama kvadratinė lygtis. Praktiniuose skaičiavimuose trečiasis momentas apskaičiuojamas naudojant apytikslę lokaliosios energijos išraišką, kurios skaičiavimo sąnaudos su sistemos dydžiu auga taip pat kaip įprasto VMC žingsnio.

Metodas išbandytas J_1–J_2 Heisenbergo modelyje kvadratinėje gardelėje. Tai sudėtingas optimizavimo uždavinys, nes ties $J_2/J_1 \approx 0{,}5$ konkuruojančios J_1 ir J_2 sąveikos lemia sudėtingą ženklų sandarą ir apsunkina optimizavimą [33–35]. Šiame darbe sąmoningai pasirinkta nespecializuota MLP architektūra be įdiegtų gardelės simetrijų, kad būtų vertinamas pats optimizavimo metodas, o ne konkrečiai pritaikytos architektūros pranašumas.

Šiuose skaitiniuose bandymuose SITE optimizavimo eiga buvo gerokai sklandesnė ir mažiau priklausė nuo pradinių sąlygų negu taikant tiesioginį energijos minimizavimą. 6×6 gardelėje SITE tikslumu konkuravo su SR, o viena SITE iteracija mūsų įgyvendinime vidutiniškai truko maždaug penkis kartus trumpiau, nes nereikėjo spręsti SR metrinio tenzoriaus lygčių sistemos. Didinant gardelę, SITE optimizuotų NQS energijos iki 8×8 dydžio gardelėse buvo artimos didelio ryšio matmens DMRG rezultatams, o didesnėse sistemose DMRG pasiekė mažesnes energijas [12, 14]. Tai svarbus apribojimas: SITE pagerina mokymą, tačiau nepanaikina poreikio tinkamai parinkti banginės funkcijos parametrizavimo ir įtraukti simetrijoms bei geometrijai pritaikytas struktūras.

Tikslumo kritimas Hofstadterio–Bose'o–Hubbardo modelyje

Antrajame pagrindiniame tyrime nagrinėjama situacija, kai NQS tikslumą riboja ne optimizavimas, o pačios architektūros galimybės aprašyti būseną. Tam $L_x \times$

L_y gardelėse su atvirosiomis kraštinėmis sąlygomis tirtas Hofstadterio–Bose'o–Hubbardo modelis begalinės lokaliosios stūmos riboje. Landau kalibruotėje hamiltonianas užrašomas

$$\hat{H} = -J \sum_{j=1}^{L_x-1} \sum_{k=1}^{L_y} \left(\hat{b}^\dagger_{j+1,k} \hat{b}_{j,k} + \text{h.c.} \right) - J \sum_{j=1}^{L_x} \sum_{k=1}^{L_y-1} \left(\mathrm{e}^{\mathrm{i}2\pi\alpha j} \hat{b}^\dagger_{j,k+1} \hat{b}_{j,k} + \text{h.c.} \right), \tag{S.13}$$

kur (j, k) yra gardelės mazgų koordinatės, $\hat{b}^\dagger_{j,k}$ ($\hat{b}_{j,k}$) yra bozonų kūrimo (naikinimo) operatoriai, o α yra magnetinis srautas per elementariąją gardelės celę. Šioje riboje kiekvieno gardelės mazgo užimtumo skaičius gali būti tik $n_{j,k} \in \{0, 1\}$. Atvirosios kraštinės sąlygos leidžia tolydžiai keisti α, o nedideles sistemas galima tiksliai diagonalizuoti ir gautus sprendinius naudoti NQS rezultatams vertinti.

NQS buvo optimizuotos SR ir SITE metodais, naudojant MLP bei CNN architektūras. Didinant magnetinį srautą nuo $\alpha = 0$ iki stipraus srauto režimo, santykinė energijos paklaida išaugo keliomis eilėmis. Svarbu tai, kad priklausomybė nuo srauto beveik nekito nei pakeitus optimizatorių, nei vietoj MLP naudojant CNN. Vadinasi, nepakanka pakeisti vien optimizatorių ar pasirinkti kitą bendro pobūdžio architektūrą.

Siekiant architektūros galimybes aprašyti būseną atskirti nuo optimizavimo ir ėminių generavimo sunkumų, tinklas prižiūrimuoju būdu tiesiogiai mokytas pagal tiksliojo diagonalizavimo sprendinį. Minimizuota ta pati persidengimu pagrįsta nuostolio funkcija kaip SITE, tik tiksline būsena pasirinkta tiksli pagrindinė būsena $|\psi_{\text{ED}}\rangle$. Mažose sistemose galima tiesiogiai apskaičiuoti visą banginės funkcijos amplitudžių lentelę, todėl ėminiai generuoti tiksliai, o prireikus nuostolio funkcija ir jos išvestinės apskaičiuotos be stochastinio triukšmo. Nustatytas sistemingas skirtumas: esant tam pačiam suspaudimo santykiui $N_\theta/\dim\mathcal{H}$, stipraus srauto būsenos ($\alpha = 0{,}3$) persidengimo nuokrypis

$$d(\psi_{\boldsymbol{\theta}}, \psi_{\text{ED}}) = 1 - |\langle\psi_{\boldsymbol{\theta}}|\psi_{\text{ED}}\rangle| \tag{S.14}$$

(abi būsenos normuotos į vienetą) buvo maždaug trimis ar keturiomis eilėmis didesnis nei nulinio srauto būsenos. Šis skirtumas išliko keičiant gardelės dydį, dalelių skaičių ir MLP ar CNN architektūros parametrus. Be to, stipraus srauto režime parametrų skaičius, reikalingas fiksuotai energijos tikslumo ribai pasiekti, gali viršyti $\dim\mathcal{H}$. Tokiu atveju NQS praranda dimensijos sumažinimo pranašumą lyginant su tiesioginiais metodais.

Toliau buvo patikrintas rinkinys hipotezių, kurios galėtų paaiškinti šį siste-

mingą tikslumo kritimą. Kompleksinė fazė negali būti pagrindinė priežastis: paklaida didėjo augant α net tada, kai tinklas mokytas atkurti tik amplitudžių modulius. Kvantinė susietis stipraus srauto būsenoje buvo didesnė, tačiau dirbtinai ją sumažinus klaida nesumažėjo iki nulinio srauto lygio. Nagrinėtose sistemose nerasta aiškių požymių, siejančių tikslumo kritimą su trupmeninio Cherno izoliatoriaus režimu. Nuostolio funkcijos paviršiaus projekcijos aplink išmokytą sprendinį buvo glotnios, o Hesės matricos spektras rodė gana platų lokalų minimumą be sudėtingos aplinkos. Niutono metodu patikslintas sprendinys persidengimą pagerino tik nežymiai. Taip pat nepadėjo paprasti architektūriniai pakeitimai: kompleksiniai neuroninio tinklo parametrai, Furjė požymiai, lokalių blokų kodavimas, išmokstami įvesties požymiai, kitos nuostolio funkcijos, mokymas palaipsniui didinant srautą ar 64 bitų slankiojo kablelio tikslumas.

Todėl pagrindinė šio skyriaus išvada atsargi, bet aiški: stipraus srauto Hofstadterio–Bose'o–Hubbardo būsenos turi struktūrą, kurią nespecializuotos MLP ir CNN NQS architektūros sistemingai aprašo neefektyviai. Tai nėra vien prasto optimizavimo pasekmė. Dėl to šis modelis tampa iššūkį keliančia etalonine sistema naujoms NQS architektūroms, kurios galėtų tiesiogiai įtraukti magnetines transliacijas, kalibruotės kovariantiškumą, Landau lygmenimis pagrįstus požymius ar kitą magnetinio srauto fiziką.

Neuroninis svarbos perrinkimas

Trečiasis šios disertacijos indėlis skirtas ėminių generavimo problemai. Įprastame VMC konfigūracijos generuojamos Markovo grandinėmis, naudojant Metropolio–Hastingso algoritmą. Šis ėminių generavimo būdas yra asimptotiškai tikslus, tačiau gali tapti nepraktiškas, kai tikslinis skirstinys yra stipriai sukoncentruotas, daugiamodis arba turi silpnai sujungtus sektorius. Tokiais atvejais autokoreliacijos sumažina efektyvųjį imties dydį, o maišymosi laiką tampa sunku patikimai įvertinti.

Šioje disertacijoje sukurtame NIR metode Markovo grandinę pakeičia svarbos perrinkimas, o pasiūlymai generuojami atskiru autoregresiniu tinklu. Šis pasiūlymus generuojantis tinklas (angl. *sampling neural network*, SNN) aprašo skirstinį

$$q(\boldsymbol{s}) = \prod_{k=1}^{N} q(s_k|\boldsymbol{s}_{<k}), \tag{S.15}$$

todėl iš jo galima generuoti nepriklausomas konfigūracijas ir tiksliai apskaičiuoti

$q(\boldsymbol{s})$. Sugeneravus pasiūlymus $\boldsymbol{s}^{(j)} \sim q$, apskaičiuojami svarbos svoriai

$$w_j \propto \frac{p(\boldsymbol{s}^{(j)})}{q(\boldsymbol{s}^{(j)})}, \tag{S.16}$$

kur p yra NQS apibrėžtas tikslinis skirstinys, proporcingas $|\psi_{\boldsymbol{\theta}}|^2$ arba $|\Psi_{\text{tot}}|^2$ kelių būsenų variacinės formuluotės atveju. Pagal šiuos svorius iš pasiūlymų sudaromas naujas ėminių rinkinys, apytiksliai pasiskirstęs pagal p. Perrinkimas būtinas, nes NQS ir pasiūlymų tinklas yra atskiri, tad jų aprašomi skirstiniai nebūtinai sutampa.

Svarbiausia praktinė sąlyga yra pakankamas skirstinių q ir p persidengimas. Tam SNN mokomas mažinti Kullbacko–Leiblerio divergenciją. Darbe lyginamos tiesioginė ir atvirkštinė Kullbacko–Leiblerio divergencijos. Šiuose eksperimentuose stabilesnė buvo tiesioginė divergencija, nes ji griežčiau baudžia už tikslinio skirstinio sričių praleidimą. Kadangi atnaujinus NQS parametrus pakinta ir skirstinys p, SNN turi prie jo nuolat prisitaikyti. Tam naudojamas efektyvusis imties dydis

$$\text{ESS} = \frac{\left(\sum_j w_j\right)^2}{\sum_j w_j^2}. \tag{S.17}$$

Jei svoriai vienodi, ESS reikšmė artima pasiūlymų skaičiui; jei dominuoja vos keli pasiūlymai, ESS yra mažas. Pasiūlymai generuojami tol, kol ESS pasiekia iš anksto nustatytą ribą. Jei tam reikia per daug pasiūlymų, prieš atnaujinant NQS parametrus vykdomos papildomos SNN mokymo iteracijos.

NIR ypač naudingas kelių būsenų NQS formuluotėms. Vienu metu kelioms žemiausioms būsenoms aprašyti disertacijoje naudojama determinantinė formuluotė. Esant K atskiroms NQS amplitudėms $\psi_a(\boldsymbol{s})$ ir K konfigūracijų rinkiniui $\boldsymbol{S} = (\boldsymbol{s}_1, \ldots, \boldsymbol{s}_K)$, bendroji banginė funkcija užrašoma

$$\Psi_{\text{tot}}(\boldsymbol{S}) = \det[\psi_a(\boldsymbol{s}_\ell)]_{\ell,a=1}^{K}. \tag{S.18}$$

Determinantas neleidžia kelioms NQS aprašyti tos pačios būsenos, nes sutapus bent dviem NQS atitinkami stulpeliai tampa tiesiškai priklausomi, o determinantas virsta nuliu. Vis dėlto tikslinis skirstinys dabar apibrėžtas K kopijų konfigūracijų erdvėje. Net jei kiekviena atskira NQS būtų autoregresinė, nėra žinomo tiesioginio būdo tiksliai generuoti ėminius pagal visą determinantu apibrėžtą skirstinį. NIR leidžia išvengti šios kliūties, nes SNN mokomas nepriklausomai nuo variacinės banginės funkcijos parametrizavimo.

Skaitiniai bandymai atlikti dvimačiame skersinio lauko Izingo modelyje (angl. *transverse-field Ising model*, TFIM) ir J_1–J_2 Heisenbergo modelyje su konkuruojančiomis sąveikomis. Feromagnetiniame Izingo režime, kai $g \ll 1$, tikslinis skirstinys yra ryškiai dvimodis: jame vyrauja dvi dideliu Hemingo atstumu atskirtos konfigūracijos, kurių visi sukiniai nukreipti aukštyn arba žemyn. Lokalus MCMC čia dažnai užstringa, ypač kelių būsenų $K = 3$ formuluotėje. NIR leido stabiliai mokyti NQS 8×8 gardelėje, o gautos energijos gerai sutapo su DMRG rezultatais: santykiniai skirtumai buvo apie 10^{-7} toli nuo kritinės srities ir padidėjo iki maždaug 10^{-5} ten, kur energijos tarpas mažiausias.

J_1–J_2 modelyje ties $J_2/J_1 = 1/2$ NIR pritaikytas vienos būsenos skaičiavimui 10×10 gardelėje su simetrijas įtraukiančia konvoliucine NQS. Galutinės energijos nepasiekė geriausių literatūroje pateiktų reikšmių, tačiau adaptyvusis ESS kriterijus leido veiksmingai generuoti ėminius; mažesnes SNN galimybes pavyko kompensuoti didesniu pasiūlymų skaičiumi. Praktinė riba taip pat aiški: jei nepavyksta nedidelėmis sąnaudomis išmokyti SNN, kurio skirstinys pakankamai persidengtų su tiksliniu skirstiniu, reikalingų pasiūlymų skaičius išauga ir NIR praranda skaičiavimo pranašumą.

Torinis kodas topologinei struktūrai tirti

Paskutiniame tyrime nagrinėjamas torinis kodas korio gardelėje su išoriniu magnetiniu lauku. Sukiniai išdėstyti briaunose, o hamiltonianas užrašomas

$$\hat{H} = -\frac{1}{2}\sum_{v}\hat{A}_v - \frac{1}{2}\sum_{p}\hat{B}_p - \sum_{j}\boldsymbol{h}\cdot\hat{\boldsymbol{\sigma}}_j, \tag{S.19}$$

su stabilizatoriais

$$\hat{A}_v = \prod_{j\in v}\hat{\sigma}_j^z, \qquad \hat{B}_p = \prod_{j\in p}\hat{\sigma}_j^x. \tag{S.20}$$

Visi stabilizatoriai tarpusavyje komutuoja, o nuliniame išoriniame magnetiniame lauke pagrindinės būsenos tenkina $\hat{A}_v = +1$ ir $\hat{B}_p = +1$ [4, 6]. Periodinė toro geometrija lemia keturgubą pagrindinių būsenų poerdvio išsigimimą, o sektorius skiria du nesutraukiamų kilpų paritetai. Skaitiniuose bandymuose magnetinis laukas imamas z kryptimi, $\boldsymbol{h} = (0, 0, h_z)$, todėl $\hat{A}_v$ stabilizatoriai komutuoja su hamiltonianu ir mažose gardelėse galima lyginti su tiksliojo diagonalizavimo rezultatais atitinkamame stabilizatorių sektoriuje.

Fizikiniu požiūriu šis modelis įdomus dėl topologinės tvarkos atsparumo išoriniam laukui. Vis dėlto šiame darbe torinis kodas pirmiausia naudojamas kaip

NQS diagnostikos priemonė. Nuliniame lauke tikslinės banginės funkcijos stabilizatorinis aprašas yra paprastas, tačiau natūralūs jos kintamieji yra nelokalūs: uždaros kilpos, stabilizatorių apribojimai ir topologiniai sektoriai. Tinklas, kuriam pateikiamos vien briaunų sukinių reikšmės be papildomų fizikinių požymių, iš principo gali šiuos dydžius apskaičiuoti, tačiau praktiškai juos išmokti gali būti sunku. Dėl keturgubo išsigimimo taikoma $K = 5$ kelių būsenų determinantinė formuluotė: keturios būsenos aprašo topologinį pagrindinių būsenų poerdvį, o penktoji leidžia įvertinti pirmąjį sužadintąjį lygmenį.

Standartinė NQS skaičiavimo schema šiame modelyje veikia nepatikimai. Lokaliai atnaujinamoms Markovo grandinėms sunku pakeisti nesutraukiamų kilpų paritetus, nes tam reikia apversti sukinius išilgai nesutraukiamos kilpos. Vien sukinių reikšmėmis aprašytos konfigūracijos tiesiogiai neparodo stabilizatorių verčių ir topologinių sektorių. Todėl energijos paklaida dažnai nustoja mažėti pasiekusi stabilizatoriaus defekto energijai artimą dydį, o vėliau mokymas tampa nestabilus.

Darbe įvesti keturi fizikinėmis žiniomis pagrįsti pakeitimai.

1. Konfigūracijų, kuriose pažeista per daug $\hat{A}_v$ stabilizatorių, logaritminei amplitudei taikoma bauda.

2. Įvesties požymiai papildomi stabilizatorių tikrinėmis reikšmėmis ir dviem Wilsono kilpų paritetais. Taip nelokalūs, sunkiai išmokstami požymiai pateikiami tinklui tiesiogiai.

3. Naudojama dvišakė MLP architektūra, kurioje vietiniai stabilizatorių požymiai ir visos sistemos kilpų paritetai pirmiausia apdorojami atskirai, o vėliau sujungiami.

4. Vietoje lokalaus Markovo grandinių ėminių generavimo taikomas anksčiau aprašytas NIR.

Su šiais pakeitimais 4×4 korio gardelėje, turinčioje 48 sukinius, $K = 5$ NQS energijos nuo tiksliojo diagonalizavimo rezultatų skiriasi santykinėmis paklaidomis nuo maždaug 10^{-6} iki 2×10^{-5} kiekvienam lygmeniui laukų intervale $h_z \in [0, 0{,}3]$. Vienu metu atkuriamos keturios išsigimusios topologinio poerdvio būsenos ir pirmoji sužadintoji būsena. Tai rodo, kad tinkamai parinkti pakeitimai leidžia NQS atkurti topologinį žemos energijos poerdvį mažose gardelėse.

Abliacijos analizė rodo, kaip šios sudedamosios dalys veikia kartu. Mažoje 2×2 gardelėje jau po kai kurių pavienių pakeitimų dalis nepriklausomų

optimizavimo bandymų konverguoja, o visų keturių pakeitimų derinys duoda tiksliausius rezultatus. 4×4 gardelėje nė vienas pavienis pakeitimas nėra pakankamas, o visuose konverguojančiuose deriniuose naudojamas nelokalus ėminių generavimas. Dvišakė architektūra tuomet dar labiau padidina tikslumą. Vis dėlto didesnėse gardelėse net su visais pakeitimais mokymas tapo nestabilus. Taigi torinis kodas lieka naudingas NQS metodų tikrinimo uždavinys: mažose gardelėse būsenos aprašymo sunkumus iš esmės pavyksta suvaldyti, o didesnėse išryškėja mokymo stabilumo problema.

Bendrosios išvados ir perspektyvos

Šios disertacijos rezultatai atskleidžia ir NQS galimybes, ir jų ribas. NQS yra lanksti variacinių banginių funkcijų šeima, leidžianti derinti automatinį diferencijavimą, įvairias neuroninių tinklų architektūras, Monte Karlo įverčius ir skirtingus ėminių generavimo metodus [21, 30–32]. Tačiau sėkmės negarantuoja nei didelis parametrų skaičius, nei tinklo gylis. Patikimumą lemia tai, ar banginės funkcijos parametrizavimo būdas, optimizatorius, ėmiklis ir tinklui pateikiami požymiai atitinka tikslinės būsenos fizikinę struktūrą.

Visų keturių tyrimų rezultatai rodo, kad nėra vieno visur dominuojančio sunkumo. J_1–J_2 Heisenbergo modelyje su konkuruojančiomis sąveikomis naudojant bendro pobūdžio NQS svarbiausias ribojantis veiksnys buvo mokymo stabilumas: architektūra galėjo pasiekti konkurencingas energijas, bet reikėjo stabilizuoti stochastinį optimizavimą. Stipraus srauto Hofstadterio–Bose'o–Hubbardo modelyje tikslumą ribojo architektūros galimybės aprašyti būseną, nes atotrūkis išliko net tiesiogiai mokant pagal tikslias pagrindines būsenas. Kelių būsenų ir feromagnetinio TFIM skaičiavimuose svarbiausias tapo ėminių generavimas: nuo NQS nepriklausomai mokomas ėmiklis veikė ten, kur lokalios Markovo grandinės strigo. Toriniame kode pirmiausia reikėjo įtraukti nelokalius požymius ir nelokalų ėminių generavimą, o didesnėse gardelėse vėl išryškėjo mokymo stabilumo riba.

Pirmoji metodinė išvada: būtina atskirti optimizavimo ir būsenos aprašymo nesėkmes. SITE skyrius rodo, kad po menamojo laiko žingsnio gauta fiksuota tikslinė būsena gali stabilizuoti mokymą tada, kai nespecializuota NQS yra pakankamai lanksti, o tiesioginis energijos minimizavimas nestabilus. Hofstadterio–Bose'o–Hubbardo skyrius rodo priešingą situaciją: net tiesioginis mokymas pagal tikslią būseną nepanaikina tikslumo kritimo stipraus magnetinio srauto režime, todėl vien geresnis optimizatorius problemos neišsprendžia.

Antroji metodinė išvada: ėminių generavimas nėra antraeilis įgyvendinimo klausimas. VMC energijų, gradientų, persidengimų ir metrikų įverčiai tiesiogiai priklauso nuo ėminių. Jei ėmiklis lėtai maišosi arba praleidžia svarbius sektorius, optimizatorius gauna iškreiptą informaciją. NIR rodo, kad ėmiklį verta įgyvendinti kaip atskirą mokomą variacinės schemos komponentą, o jo skirstinio atitiktį NQS Borno skirstiniui galima stebėti pagal efektyvųjį imties dydį.

Trečioji metodinė išvada: į skaičiavimo schemą svarbu įtraukti iš anksto žinomą informaciją apie sistemą. Nereikėtų tikėtis, kad dabartiniai NQS metodai visada patikimai patys atras simetrijas, tvermės dėsnius, stabilizatorių apribojimus, kilpų paritetus, ženklų sandarą ar kitus žinomus fizikinius požymius. Torinio kodo atvejis tai parodo aiškiausiai, bet tas pats principas galioja ir magnetams su konkuruojančiomis sąveikomis bei magnetinio srauto veikiamiems bozonų modeliams. Tikėtina, kad patikimiausi NQS metodai bus hibridiniai: pakankamai lankstūs aprašyti analitiškai nežinomas koreliacijas, bet kartu apriboti ar papildyti taip, kad atitiktų žinomus fizikinius dėsningumus.

Iš šių išvadų natūraliai kyla tolesnių tyrimų kryptys. SITE verta jungti su simetrijas įtraukiančiomis ar ekvivariantinėmis NQS architektūromis ir sistemingai lyginti su naujausiomis dideliems tinklams pritaikytomis SR versijomis. Hofstadterio–Bose'o–Hubbardo modeliui spręsti reikėtų architektūrų, tiesiogiai įtraukiančių magnetinių transliacijų ir kalibruotės struktūrą. Tokios fizikinėmis žiniomis pagrįstos architektūros jau kuriamos; be to, nauji mažesnės dispersijos fazės gradiento įverčiai rodo, kad būsimuose tyrimuose reikės dar tiksliau atskirti būsenos parametrizavimo ir stochastinio įvertinimo ribojimus [142, 143]. NIR galima tobulinti pasitelkiant fiziškai labiau pritaikytus autoregresinius ar normuojamųjų srautų tipo pasiūlymų modelius ir nuostolio funkcijas, labiau skatinančias aprėpti visą tikslinį skirstinį. Topologiniams modeliams reikia architektūrų ir ėminių generavimo metodų, į kuriuos stabilizatorių sektoriai bei topologinės kilpos būtų įtraukti tiesiogiai, o ne palikti tinklui išmokti iš sukinių konfigūracijų.

Apibendrinant, šio darbo išvada yra dvejopa. Pirma, NQS yra lanksti ir perspektyvi kvantinių daugiadalelių sistemų modeliavimo priemonė, o šiame darbe pasiūlyti metodai patobulina jos optimizavimą ir ėminių generavimą. Antra, aptiktos ribos nėra atsitiktinės įgyvendinimo klaidos. Jos parodo, kad NQS metodus būtina grįsti žinoma fizikine struktūra: atsižvelgti į sistemos geometriją ir apribojimus, tinkamai parinkti ėminių generavimo būdą ir atskirai diagnozuoti kiekvieną galimą tikslumo ribojimą. Šiais principais grindžiamos

NQS gali tapti patikimesne kvantinių daugiadalelių sistemų tyrimo priemone, o ne vien lanksčia variacine forma, kurios galutinę energiją sunku interpretuoti.

This page is intentionally left blank.

Curriculum vitae

Eimantas Ledinauskas
eimantas.ledinauskas@ff.stud.vu.lt

Research interests: neural quantum states and variational methods for quantum many-body systems; machine learning for scientific computing; computer vision; numerical astrophysics.

Education

Doctoral studies in Theoretical Physics, Vilnius University 2022–2026
Thesis: *Neural Quantum States for Many-Body Physics: Limitations and Alternative Optimization Approaches*.

MSc in Theoretical Physics and Astrophysics, Vilnius University 2016–2018
Graduated *magna cum laude*; researched the formation and evolution of isolated dwarf galaxies.

BSc in Computational Physics, Vilnius University 2012–2016
Graduated *cum laude*; studied the hydrodynamic evolution of a star-forming molecular cloud.

Research Experience

R&D, Baltic Institute of Advanced Technology 2018–2025
Developed machine-learning and computer-vision methods for EU- and EDA-funded programmes and conducted research on deep spiking neural networks.

Research, Center for Physical Sciences and Technology 2014–2018
Developed analytical and numerical models of star formation and dwarf-galaxy evolution.

Research, Vilnius University, Department of Theoretical Physics 2013–2014
Performed and analyzed numerical simulations in quantum chemistry.

Teaching

Vilnius Vytautas Magnus Gymnasium 2013–2014
Prepared students in grades 9–12 for physics olympiads in an extracurricular class.

Acknowledgements

First and foremost, I am deeply grateful to my supervisor, Egidijus Anisimovas, for guiding me throughout my doctoral studies: for the many hours spent brainstorming whenever I found myself stuck, for co-authoring the publications that make up this thesis, and, not least, for remaining a generous and genuinely good-humoured mentor even while carrying the responsibilities of directing the Institute of Theoretical Physics and Astronomy.

I would also like to thank my colleagues Julius Ruseckas and Mantas Gabrielaitis for the many thought-provoking discussions that helped me untangle difficult problems and shape new research ideas. I am equally grateful to Artūras Acus, Vidas Regelskis, Jevgenij Chmeliov, and Andrius Gelžinis for their engaging and valuable courses and exercises.

I am grateful to the manuscript reviewers, Donatas Narbutis and Jevgenij Chmeliov, whose careful reading, useful comments, and thoughtful suggestions substantially improved this thesis.

Last but by no means least, I thank my wife, Gabrielė Ledinauskienė, for her steady support, her patience and understanding, and for keeping me going when my own resolve ran low.

List of publications

Peer-reviewed publications

[P1] Eimantas Ledinauskas and Egidijus Anisimovas. "Scalable imaginary time evolution with neural network quantum states". *SciPost Physics* 15.6 (2023), p. 229. DOI: 10.21468/SciPostPhys.15.6.229

[P2] Eimantas Ledinauskas and Egidijus Anisimovas. "Universal performance gap of neural quantum states applied to the Hofstadter-Bose-Hubbard model". *SciPost Physics* 18.1 (2025), p. 011. DOI: 10.21468/SciPostPhys.18.1.011

[P3] Eimantas Ledinauskas and Egidijus Anisimovas. "Neural importance resampling: a practical sampling strategy for neural quantum states". *Machine Learning: Science and Technology* 7.1 (2026), p. 015020. DOI: 10.1088/2632-2153/ae37c9

List of presentations

Conference poster presentations

1. *Challenges in Neural Quantum State Performance: Insights from the Bose–Hubbard Model with Magnetic Field*, Open Readings 2024, Vilnius, Lithuania, 2024.

2. *Exploring the Quantum Robustness of the Toric Code Using Neural Quantum States*, 29th Central European Workshop on Quantum Optics (CEWQO29), Vilnius, Lithuania, 2025.

3. *Quantum Robustness of the Toric Code via Multi-State Neural Quantum States*, 46th Lithuanian National Physics Conference, Kaunas, Lithuania, 2025.

Vilniaus universiteto leidykla
Saulėtekio al. 9, III rūmai, LT-10222 Vilnius
El. p. info@leidykla.vu.lt, www.leidykla.vu.lt
bookshop.vu.lt, journals.vu.lt
Tiražas 6 egz.