Rayleigh-Gauss-Newton optimization with enhanced sampling for variational Monte Carlo

Robert J. Webber and Michael Lindsey
Courant Institute of Mathematical Sciences, New York University, New York 10012, USA

(Received 20 June 2021; accepted 17 June 2022; published 3 August 2022)

Variational Monte Carlo (VMC) is an approach for computing ground-state wave functions that has recently become more powerful due to the introduction of neural network-based wave-function parametrizations. However, efficiently training neural wave functions to converge to an energy minimum remains a difficult problem. In this work, we analyze optimization and sampling methods used in VMC and introduce alterations to improve their performance. First, based on theoretical convergence analysis in a noiseless setting, we motivate a new optimizer that we call the Rayleigh-Gauss-Newton method, which can improve upon gradient descent and natural gradient descent to achieve superlinear convergence at no more than twice the computational cost. Second, to realize this favorable comparison in the presence of stochastic noise, we analyze the effect of sampling error on VMC parameter updates and experimentally demonstrate that it can be reduced by the parallel tempering method. In particular, we demonstrate that RGN can be made robust to energy spikes that occur when the sampler moves between metastable regions of configuration space. Finally, putting theory into practice, we apply our enhanced optimization and sampling methods to the transverse-field Ising and XXZ models on large lattices, yielding ground-state energy estimates with remarkably high accuracy after just 200 parameter updates.

DOI: 10.1103/PhysRevResearch.4.033099

I. INTRODUCTION

Computing the ground-state wave function of a many-body Hamiltonian operator is a demanding task, requiring the solution of an eigenvalue problem whose cost grows exponentially with system size in traditional numerical approaches. Variational Monte Carlo (VMC, [1,2]) is an alternative strategy that avoids this curse of dimensionality by using stochastic optimization to find the best wave function within a tractable function class.

VMC has recently seen rapid and encouraging development due to the incorporation of insights from the machine learning community. In 2017, Carleo and Troyer [3] applied VMC with a two-layer neural network ansatz to accurately represent the ground-state wave function of quantum spin systems with as many as 100 spins. Since then, there has been major progress in extending neural network-based VMC to the setting of electronic structure, including the development of the neural network backflow ansatz for second-quantized lattice problems [4], as well as of FermiNet [5] and PauliNet [6] for quantum chemistry problems in first quantization. These new approaches have been extended to systems as large as bicyclobutane (C_4H_6), which has 30 interacting electrons [5,7].

VMC is highly flexible, since it extends without significant modification to systems of arbitrary spatial dimension. However, the price paid for this flexibility is a difficult optimization problem that relies on Monte Carlo sampling. Efficiently solving this optimization problem has proven challenging. Recent works [4,6–10] raise concerns about the speed and stability of wave-function training and report that VMC can suffer from long training times [5,7], lose stability [10], or converge to unreasonable solutions [11]. Thus, there is motivation for the development of faster and more stable optimization and sampling solutions.

Our goal is to apply numerical and probabilistic analysis to evaluate and improve upon the optimization and sampling strategies in VMC. To this end, we first provide a unified perspective on several major VMC optimizers, namely, gradient descent, quantum natural gradient descent (also known as stochastic reconfiguration), and the linear method. Reviewing these methods in a unified way clarifies a path toward improvement. Specifically, we introduce a new *Rayleigh-Gauss-Newton* (RGN) method and prove RGN achieves superlinear convergence as the wave function approaches the ground state.

Next we analyze the Markov chain Monte Carlo (MCMC) sampling used in VMC. We establish a quantitative extension of the zero-variance principle [1,2] of VMC that we call the vanishing-variance principle. This principle guarantees that the energy estimates converge to the true energy as the wave function nears an eigenstate. However, away from an eigenstate, the accuracy of the energy estimates is *not* guaranteed. The energy estimates can have a high variance and can even exhibit energy spikes (see Fig. 4). To stabilize these energy estimates, variance reduction strategies are needed. Using a standard MCMC sampler as in [3], the wave function is slow to recover from the energy spikes ($\sim 10^3$ iterations); however, using the parallel tempering MCMC method [12], the

recovery period is much quicker ($\sim 10^2$ iterations). Variance reduction strategies such as parallel tempering can be essential for realizing the full potential of VMC in large-scale applications.

Lastly, by using the Rayleigh-Gauss-Newton method along with parallel tempering, we obtain highly accurate variational estimates for the ground-state energies of transverse-field Ising and XXZ models with as many as 400 spins. Compared to past benchmark results obtained using natural gradient descent [3], we obtain the same or higher accuracy in fewer iterations. Since RGN is only slightly more expensive than natural gradient descent, by less than a factor of two in our tests, we conclude that RGN can improve the overall efficiency of VMC.

The rest of the paper is organized as follows. Section II gives an overview of variational Monte Carlo, Sec. III analyzes optimization methods, Sec. IV analyzes sampling methods, Sec. V presents numerical experiments, and Sec. VI concludes.

Throughout the paper, $\Re z$ denotes the real part of a complex number z. $\boldsymbol{v}^T$, $\overline{\boldsymbol{v}}$, and $\boldsymbol{v}^*$ denote the transpose, complex conjugate, and conjugate transpose of a vector $\boldsymbol{v}$, and similar conventions are adopted for matrices. We use single bars $|\cdot|$ for the Euclidean norm of a scalar, vector, or matrix and $\|\cdot\|_2$ for the spectral norm of a matrix. Last, we consider a finite- or infinite-dimensional Hilbert space of unnormalized wave functions ψ and use $\langle\cdot,\cdot\rangle$ and $\|\cdot\|$ to denote the associated inner product and norm.

II. OVERVIEW OF VMC

The main goal of variational Monte Carlo (VMC) is the identification of the ground-state energy and wave function of the Hamiltonian operator $\mathcal{H}$ for a quantum many-body system. We denote the ground-state energy and wave function using λ_0 and ψ_0, respectively. In addition to solving the eigenvalue equation $\mathcal{H}\psi_0 = \lambda_0\psi_0$, these admit a variational characterization in terms of the energy functional

$$\mathcal{E}[\psi] = \frac{\langle\psi, \mathcal{H}\psi\rangle}{\langle\psi, \psi\rangle}. \tag{1}$$

The ground-state energy λ_0 is the minimum value of $\mathcal{E}$, and the ground-state wave function ψ_0 is the minimizer, which we assume to be unique up to an arbitrary multiplicative constant.

Identifying λ_0 and ψ_0 becomes difficult when the Hilbert space associated with $\mathcal{H}$ is high-dimensional or infinite-dimensional. For example, in the Heisenberg model for spin-1/2 particles on a graph [13], $\mathcal{H}$ is the operator

$$\mathcal{H} = \sum_{i\sim j}\left[J_x\sigma_i^x\sigma_j^x + J_y\sigma_i^y\sigma_j^y + J_z\sigma_i^z\sigma_j^z\right] + h\sum_i \sigma_i^x, \tag{2}$$

where σ_i^x, σ_i^y, and σ_i^z are Pauli operators for the ith spin, $i \sim j$ signifies that i and j are neighboring spins, and J_x, J_y, J_z, and h are real-valued parameters. In the case, e.g., of a 10×10 square lattice, the ground-state wave function can be viewed as a vector of length 2^{100}, which is far too large to store in memory, much less calculate with any conventional eigensolver, direct or iterative.

VMC must approximate this high-dimensional eigenvector using a tractable parametrization $\psi = \psi_{\boldsymbol{\theta}}$, where $\boldsymbol{\theta}$ is a vector of real- or complex-valued parameters. VMC uses an iterative approach for updating the $\boldsymbol{\theta}$ parameters, with the goal of minimizing the energy within the parametric class. VMC iterates over the following three steps.

(1) Draw random samples from the wave-function density $\rho_{\boldsymbol{\theta}} = |\psi_{\boldsymbol{\theta}}|^2/\langle\psi_{\boldsymbol{\theta}}, \psi_{\boldsymbol{\theta}}\rangle$.

(2) Use the random samples to estimate the energy $\mathcal{E}[\psi_{\boldsymbol{\theta}}]$, the energy gradient $\nabla_{\boldsymbol{\theta}}\mathcal{E}[\psi_{\boldsymbol{\theta}}]$, and possibly other quantities needed for the optimization.

(3) Update the $\boldsymbol{\theta}$ parameters to reduce the energy.

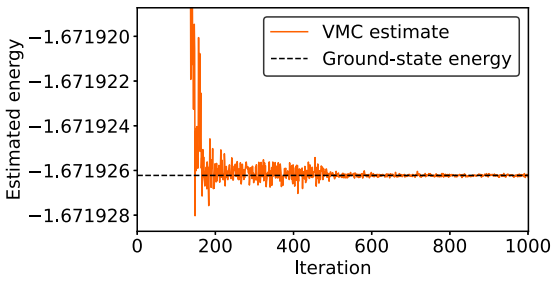

FIG. 1. VMC ground-state energy estimates for a 200×1 Ising model with a transverse magnetic field ($h = 1.5$). Computational details are provided in Sec. V.

In VMC, we ideally find that the estimated energies fall quickly in the first iterations and decrease more slowly at subsequent iterations, yielding increasingly accurate estimates of λ_0, as shown in Fig. 1.

Additionally, as seen in Fig. 1, there is a *vanishing-variance principle* by which the energy estimator's variance converges to zero as the wave function approaches the ground state of $\mathcal{H}$ (see Proposition 3). Because of this principle, reductions in the energy mean and reductions in the energy variance both indicate that the wave function is approaching the ground state. The vanishing-variance principle is essential in applications, since it enables VMC to provide accurate energy estimates even though the variance at the early stages of the optimization would appear to render such high accuracy impossible.

III. OPTIMIZATION APPROACHES

In this section, we obtain formulas for the energy gradient and Hessian, use these formulas to motivate optimization methods for VMC, and lastly derive theoretical convergence rates for VMC optimizers. Throughout the section, we assume that optimization methods are applied exactly without any Monte Carlo sampling.

A. The energy gradient and Hessian

To begin, we derive formulas for the energy gradient and Hessian with respect to the parameters. By adopting the convention of intermediate normalization [14], we obtain compact expressions for these quantities that differ from past presentations, e.g., Ref. [1, chap. 9].

We fix a vector of parameters $\boldsymbol{\theta}$ and consider a small parameter update $\boldsymbol{\theta} + \boldsymbol{\delta}$. The resulting wave function, after

intermediate normalization, is written

$$\widehat{\psi}_{\boldsymbol{\theta}+\boldsymbol{\delta}} = \frac{\langle \psi_{\boldsymbol{\theta}}, \psi_{\boldsymbol{\theta}} \rangle}{\langle \psi_{\boldsymbol{\theta}}, \psi_{\boldsymbol{\theta}+\boldsymbol{\delta}} \rangle} \psi_{\boldsymbol{\theta}+\boldsymbol{\delta}}. \tag{3}$$

This intermediate-normalized wave function is a scalar multiple of the unnormalized wave function $\psi_{\boldsymbol{\theta}+\boldsymbol{\delta}}$ and hence has the same energy. However, $\widehat{\psi}_{\boldsymbol{\theta}+\boldsymbol{\delta}}$ has been rescaled to fix the inner product with $\psi_{\boldsymbol{\theta}}$.

We assume that $\boldsymbol{\delta} \mapsto \widehat{\psi}_{\boldsymbol{\theta}+\boldsymbol{\delta}}$ is a locally analytic function of real or complex parameters and consider the second-order Taylor series expansion,

$$\widehat{\psi}_{\boldsymbol{\theta}+\boldsymbol{\delta}} = \widehat{\psi} + \sum_i \boldsymbol{\delta}_i \widehat{\psi}_i + \frac{1}{2} \sum_{ij} \boldsymbol{\delta}_i \boldsymbol{\delta}_j \widehat{\psi}_{ij} + \mathcal{O}(|\boldsymbol{\delta}|^3), \tag{4}$$

where $\widehat{\psi}$, $\widehat{\psi}_i$, and $\widehat{\psi}_{ij}$ denote the normalized wave function and its partial derivatives

$$\widehat{\psi} = \widehat{\psi}_{\boldsymbol{\theta}}, \quad \widehat{\psi}_i = \partial_{\boldsymbol{\theta}_i} \widehat{\psi}_{\boldsymbol{\theta}}, \quad \widehat{\psi}_{ij} = \partial^2_{\boldsymbol{\theta}_i \boldsymbol{\theta}_j} \widehat{\psi}_{\boldsymbol{\theta}}. \tag{5}$$

Manipulating Eq. (4), we then decompose the energy difference $\mathcal{E}[\widehat{\psi}_{\boldsymbol{\theta}+\boldsymbol{\delta}}] - \mathcal{E}[\widehat{\psi}_{\boldsymbol{\theta}}]$ into the sum of gradient and Hessian terms:

$$\underbrace{\mathcal{E}[\widehat{\psi}_{\boldsymbol{\theta}+\boldsymbol{\delta}}] - \mathcal{E}[\widehat{\psi}_{\boldsymbol{\theta}}]}_{\text{energy difference}} = \underbrace{\boldsymbol{\delta}^* \boldsymbol{g} + \boldsymbol{g}^* \boldsymbol{\delta}}_{\text{gradient terms}} + \underbrace{\boldsymbol{\delta}^* \boldsymbol{H} \boldsymbol{\delta} + \Re(\boldsymbol{\delta}^T \overline{\boldsymbol{J}}\, \boldsymbol{\delta})}_{\text{Hessian terms}} + \mathcal{O}(|\boldsymbol{\delta}|^3). \tag{6}$$

These gradient and Hessian terms are given explicitly by

$$\boldsymbol{g}_i = \frac{\langle \widehat{\psi}_i, \widehat{\mathcal{H}}\, \widehat{\psi} \rangle}{\langle \widehat{\psi}, \widehat{\psi} \rangle}, \quad \boldsymbol{H}_{ij} = \frac{\langle \widehat{\psi}_i, \widehat{\mathcal{H}}\, \widehat{\psi}_j \rangle}{\langle \widehat{\psi}, \widehat{\psi} \rangle}, \tag{7}$$

$$\boldsymbol{J}_{ij} = \frac{\langle \widehat{\psi}_{ij}, \widehat{\mathcal{H}}\, \widehat{\psi} \rangle}{\langle \widehat{\psi}, \widehat{\psi}\, r \rangle}, \tag{8}$$

where $\widehat{\mathcal{H}} = \mathcal{H} - \mathcal{E}[\widehat{\psi}]$ is an energy-shifted version of the operator $\mathcal{H}$.

Equations (7) and (8) offer transparent formulas for the energy gradient and Hessian. In the case of real-valued parameters, the energy gradient is $2\boldsymbol{g}$, and the energy Hessian is $2\boldsymbol{H} + 2\boldsymbol{J}$. In the case of complex-valued parameters (such as in the setting of Ref. [3]), the Wirtinger gradient [15,16] of the energy is $\begin{pmatrix} \boldsymbol{g} \\ \overline{\boldsymbol{g}} \end{pmatrix}$, and the Wirtinger Hessian is $\begin{pmatrix} \boldsymbol{H} & \boldsymbol{J} \\ \overline{\boldsymbol{J}} & \overline{\boldsymbol{H}} \end{pmatrix}$.

The structure of the Hessian simplifies near the ground state, since $\boldsymbol{J} \to \boldsymbol{0}$ as the wave function approaches any eigenstate of $\mathcal{H}$.

Proposition 1. The matrix $\boldsymbol{J}$ is bounded by

$$|\boldsymbol{J}_{ij}| \leqslant \frac{\|\widehat{\psi}_{ij}\|}{\|\widehat{\psi}\|} \min_{\lambda \in \mathbb{R}} \frac{\|(\mathcal{H} - \lambda)\widehat{\psi}\|}{\|\widehat{\psi}\|}. \tag{9}$$

Therefore, $\boldsymbol{J} \to \boldsymbol{0}$ as $\min_{\lambda \in \mathbb{R}} \|(\mathcal{H} - \lambda)\widehat{\psi}\| / \|\widehat{\psi}\| \to 0$, assuming uniformly bounded $\|\widehat{\psi}_{ij}\| / \|\widehat{\psi}\|$ terms.

Proof. Apply the Cauchy-Schwartz inequality to Eq. (8), and use the fact that $\|\widehat{\mathcal{H}}\, \widehat{\psi}\| = \min_{\lambda \in \mathbb{R}} \|(\mathcal{H} - \lambda)\widehat{\psi}\|$ ■.

As the wave function approaches an eigenstate, Proposition 1 reveals that the Hessian or Wirtinger Hessian takes a simple structure, depending only on first derivatives of the wave function. To our knowledge this fact has not been previously identified. An important implication, to be spelled out below in Sec. III D, is that first derivatives suffice to achieve superlinear convergence in VMC optimization, under the assumption that the true ground state lies within our parametric class.

B. Gradient descent methods

The main idea in gradient descent methods is to first approximate the energy using

$$\mathcal{E}_{\text{linear}}[\hat{\psi}_{\boldsymbol{\theta}+\boldsymbol{\delta}}] - \mathcal{E}[\hat{\psi}_{\boldsymbol{\theta}}] = \boldsymbol{\delta}^* \boldsymbol{g} + \boldsymbol{g}^* \boldsymbol{\delta} \tag{10}$$

and then choose $\boldsymbol{\delta}$ to minimize Eq. (10), plus a penalization term that keeps the update small. The penalization term may take the form

$$\frac{|\boldsymbol{\delta}|^2}{\epsilon} \quad \text{or} \quad \frac{\angle(\hat{\psi}_{\boldsymbol{\theta}}, \hat{\psi}_{\boldsymbol{\theta}+\boldsymbol{\delta}})^2}{\epsilon}, \tag{11}$$

where $\epsilon > 0$ is a tunable parameter. In the first case, we are restricting the Euclidean norm $|\boldsymbol{\delta}|$ and the resulting method is standard gradient descent. In the second case, we are restricting the angle between wave functions:

$$\angle(\hat{\psi}_{\boldsymbol{\theta}}, \hat{\psi}_{\boldsymbol{\theta}+\boldsymbol{\delta}}) = \arccos \frac{|\langle \hat{\psi}_{\boldsymbol{\theta}}, \hat{\psi}_{\boldsymbol{\theta}+\boldsymbol{\delta}} \rangle|}{\|\hat{\psi}_{\boldsymbol{\theta}}\| \|\hat{\psi}_{\boldsymbol{\theta}+\boldsymbol{\delta}}\|}. \tag{12}$$

This leads to a method called "stochastic reconfiguration" or "(quantum) natural gradient descent" [2], which has been used extensively to optimize traditional [17,18] and more recent [3,5] VMC wave-function ansatzes.

In a high-dimensional or infinite-dimensional Hilbert space, the angle $\angle(\hat{\psi}_{\boldsymbol{\theta}}, \hat{\psi}_{\boldsymbol{\theta}+\boldsymbol{\delta}})$ cannot be computed exactly, so natural gradient descent takes advantage of the Taylor series expansion

$$\angle(\hat{\psi}_{\boldsymbol{\theta}}, \hat{\psi}_{\boldsymbol{\theta}+\boldsymbol{\delta}})^2 = \boldsymbol{\delta}^* \boldsymbol{S} \boldsymbol{\delta} + \mathcal{O}(|\boldsymbol{\delta}|^3), \tag{13}$$

where

$$\boldsymbol{S}_{ij} = \frac{\langle \widehat{\psi}_i, \widehat{\psi}_j \rangle}{\langle \widehat{\psi}, \widehat{\psi} \rangle} \tag{14}$$

is a positive semidefinite matrix known as the Fubini-Study metric or quantum information metric [19]. However, instead of directly using a penalization term

$$\frac{\boldsymbol{\delta}^* \boldsymbol{S} \boldsymbol{\delta}}{\epsilon}, \tag{15}$$

natural gradient descent uses a slightly modified penalization term

$$\frac{\boldsymbol{\delta}^* (\boldsymbol{S} + \eta \boldsymbol{I}) \boldsymbol{\delta}}{\epsilon}. \tag{16}$$

Here, $\eta > 0$ is a parameter that makes the matrix $\boldsymbol{S} + \eta \boldsymbol{I}$ positive definite and prevents large updates when the Taylor series expansion (13) is not very accurate.

To make the preceding discussion precise, we formalize gradient descent (GD) methods as follows.

Algorithm 1 (GD methods). Choose $\boldsymbol{\delta}$ to solve

$$\min_{\boldsymbol{\delta}} \left[\boldsymbol{\delta}^* \boldsymbol{g} + \boldsymbol{g}^* \boldsymbol{\delta} + \frac{\boldsymbol{\delta}^* \boldsymbol{R} \boldsymbol{\delta}}{\epsilon} \right], \tag{17}$$

where $\boldsymbol{R} = \boldsymbol{I}$ in GD and $\boldsymbol{R} = \boldsymbol{S} + \eta \boldsymbol{I}$ in natural GD. Equivalently, set

$$\boldsymbol{\delta} = -\epsilon \boldsymbol{R}^{-1} \boldsymbol{g}. \tag{18}$$

In addition to GD and natural GD, alternative gradient descent methods such as Adam [20] and AMSGrad [21] have recently gained traction in the VMC community [5,22–24]. These "momentum-based" methods form updates by combining the energy gradient at the current iteration and past iterations. While such strategies are potentially promising, recent tests [5,25] suggest that natural GD still outperforms momentum-based methods on several challenging VMC test problems. Therefore, we focus on GD and natural GD, leaving analysis of other gradient descent methods for future work.

C. Rayleigh-Gauss-Newton method

Whereas gradient descent methods are based on a linear approximation of the energy, we now introduce a method—which we call the Rayleigh-Gauss-Newton (RGN) method—based on the following quadratic energy approximation:

$$\mathcal{E}_{\text{quad}}[\widehat{\psi}_{\boldsymbol{\theta}+\boldsymbol{\delta}}] - \mathcal{E}[\widehat{\psi}_{\boldsymbol{\theta}}] = \boldsymbol{\delta}^* \boldsymbol{g} + \boldsymbol{g}^* \boldsymbol{\delta} + \boldsymbol{\delta}^* \boldsymbol{H} \boldsymbol{\delta}. \tag{19}$$

Here, $\boldsymbol{\delta}^* \boldsymbol{g}$ and $\boldsymbol{g}^* \boldsymbol{\delta}$ are the exact gradient terms, while $\boldsymbol{\delta}^* \boldsymbol{H} \boldsymbol{\delta}$ is just one of the Hessian terms. There is a strong practical motivation for ignoring the other Hessian term $\Re(\boldsymbol{\delta}^T \boldsymbol{J} \boldsymbol{\delta})$, since evaluating this term would requiring taking second derivatives of the wave function with respect to all pairs of parameters, which becomes burdensome as the number of parameters grows large.

In the RGN method, we minimize the quadratic objective function (19) plus a "natural" penalization term, as described below.

Algorithm 2 (RGN method). Choose $\boldsymbol{\delta}$ to solve

$$\min_{\boldsymbol{\delta}} \left[\boldsymbol{\delta}^* \boldsymbol{g} + \boldsymbol{g}^* \boldsymbol{\delta} + \boldsymbol{\delta}^* \boldsymbol{H} \boldsymbol{\delta} + \frac{\boldsymbol{\delta}^* \boldsymbol{R} \boldsymbol{\delta}}{\epsilon} \right], \tag{20}$$

where $\boldsymbol{R} = \boldsymbol{S} + \eta \boldsymbol{I}$. Equivalently, set

$$\boldsymbol{\delta} = -(\boldsymbol{H} + \epsilon^{-1} \boldsymbol{R})^{-1} \boldsymbol{g}. \tag{21}$$

The parameter $\eta > 0$ is again chosen to make $\boldsymbol{H} + \epsilon^{-1}(\boldsymbol{S} + \eta \boldsymbol{I})$ positive definite and help prevent large parameter updates.

To our knowledge, the RGN method has not appeared before in the literature. However, it is closely connected to the classical Gauss-Newton method for nonlinear least squares problems [26], which can be viewed as deriving from a similar Hessian splitting. Also, RGN is related to previous VMC optimization methods, including the linear method for energy minimization [2] and a Gauss-Newton-like method for variance minimization [27]. All these approaches can be can be described by first linearizing a class of functions

$$\hat{\psi}_{\boldsymbol{\theta}+\boldsymbol{\delta}} \approx \hat{\psi} + \sum_i \boldsymbol{\delta}_i \hat{\psi}_i, \tag{22}$$

and then minimizing a nonlinear loss function applied to the linearized function class

$$\min_{\boldsymbol{\delta}} \mathcal{L}\left[\hat{\psi} + \sum_i \boldsymbol{\delta}_i \hat{\psi}_i \right]. \tag{23}$$

For example, in the linear method for VMC, one first linearizes the intermediate-normalized wave function $\widehat{\psi}_{\boldsymbol{\theta}+\boldsymbol{\delta}}$ and then minimizes

$$\mathcal{E}\left[\widehat{\psi} + \sum_i \boldsymbol{\delta}_i \widehat{\psi}_i \right] - \mathcal{E}[\widehat{\psi}] = \frac{\boldsymbol{\delta}^* \boldsymbol{g} + \boldsymbol{g}^* \boldsymbol{\delta} + \boldsymbol{\delta}^* \boldsymbol{H} \boldsymbol{\delta}}{1 + \boldsymbol{\delta}^* \boldsymbol{S} \boldsymbol{\delta}}, \tag{24}$$

plus an additional penalization term. Minimization of Eq. (24) is equivalent to solving the generalized eigenvalue problem

$$\begin{pmatrix} 0 & \boldsymbol{g}^* \\ \boldsymbol{g} & \boldsymbol{H} \end{pmatrix} \begin{pmatrix} 1 \\ \boldsymbol{\delta} \end{pmatrix} = \lambda \begin{pmatrix} 1 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{S} \end{pmatrix} \begin{pmatrix} 1 \\ \boldsymbol{\delta} \end{pmatrix} \tag{25}$$

for the smallest eigenvalue-eigenvector pair [1,2]. As a penalization term, the matrix $\boldsymbol{H}$ is padded with a diagonal matrix $\epsilon^{-1} \boldsymbol{I}$, which is similar to the penalization term used in GD.

The linear method has been observed to yield fast asymptotic convergence in VMC applications with small parameter sets. However, extending the linear method to larger parameter sets is an ongoing challenge [22,28]. Motivated by the linear method's potential for fast asymptotic convergence, we introduce RGN as an updated strategy with improved convergence. RGN differs from the linear method in two ways, as detailed below.

First, instead of approximating the energy using the formula (24), RGN uses the quadratic approximation

$$\mathcal{E}_{\text{quad}}[\widehat{\psi}_{\boldsymbol{\theta}+\boldsymbol{\delta}}] - \mathcal{E}[\widehat{\psi}_{\boldsymbol{\theta}}] = \boldsymbol{\delta}^* \boldsymbol{g} + \boldsymbol{g}^* \boldsymbol{\delta} + \boldsymbol{\delta}^* \boldsymbol{H} \boldsymbol{\delta}. \tag{26}$$

This quadratic approximation agrees with Eq. (24) up to $\mathcal{O}(|\boldsymbol{\delta}|^3)$ terms, but it only requires the solution of a linear system instead of a generalized eigenvalue problem. Although the parametrizations considered in our numerical experiments below are small enough so that neither of these linear algebra routines imposes a computational bottleneck, the distinction may become important for large parameter sets. For example, Ref. [28] introduced a matrix-free approach for solving the linear system in stochastic reconfiguration, but a similar scheme for the generalized eigenvalue problem has not achieved the same success [29].

Second, RGN uses a "natural" penalization term $\epsilon^{-1} \boldsymbol{\delta}^* (\boldsymbol{S} + \eta \boldsymbol{I}) \boldsymbol{\delta}$, which differs from the penalization term used in the linear method. Because of the penalization, the linear method converges as $\epsilon \to 0$ to give standard GD updates. In contrast, RGN converges as $\epsilon \to 0$ to give natural GD updates, which are more efficient than standard GD updates when optimizing many VMC wave-function ansatzes [5,25].

D. Convergence rate analysis

GD, natural GD, and RGN can all be presented in the standardized form

$$\boldsymbol{P}^i (\boldsymbol{\theta}^{i+1} - \boldsymbol{\theta}^i) = -\boldsymbol{g}(\boldsymbol{\theta}^i), \quad i = 1, 2, \ldots \tag{27}$$

Here, the parameter update $\boldsymbol{\theta}^{i+1} - \boldsymbol{\theta}^i$ is written as the solution to a linear system involving a positive definite preconditioning matrix $\boldsymbol{P}^i$ and a negative energy gradient $-\boldsymbol{g}(\boldsymbol{\theta}^i)$. Table I shows the different preconditioners corresponding to the different optimization approaches.

To help quantify the efficiency of the various optimization methods, Proposition 2 considers a general sequence of positive definite preconditioners $\boldsymbol{P}^1, \boldsymbol{P}^2, \ldots$ and derives sharp asymptotic bounds on the resulting energies

TABLE I. Different preconditioners for energy minimization.

Method	Preconditioner $\boldsymbol{P}$
Gradient descent	$\epsilon^{-1}\boldsymbol{I}$
Natural gradient descent	$\epsilon^{-1}(\boldsymbol{S}+\eta\boldsymbol{I})$
Rayleigh-Gauss-Newton	$\boldsymbol{H}+\epsilon^{-1}(\boldsymbol{S}+\eta\boldsymbol{I})$

$\mathcal{E}[\psi_{\boldsymbol{\theta}^1}], \mathcal{E}[\psi_{\boldsymbol{\theta}^2}], \ldots$. Proposition 2 is based on standard optimization theory (e.g., Ref. [26]), but here we extend this theory to the complex-valued wave functions often used in VMC.

Proposition 2. Consider the parameter updates $\boldsymbol{P}^i(\boldsymbol{\theta}^{i+1} - \boldsymbol{\theta}^i) = -\boldsymbol{g}(\boldsymbol{\theta}^i)$. Assume $\boldsymbol{\theta}^1, \boldsymbol{\theta}^2, \ldots$ converges to a local energy minimizer $\boldsymbol{\theta}^*$, and the Hessian or Wirtinger Hessian is positive definite at $\boldsymbol{\theta}^*$. Then,

$$\limsup_{i\to\infty} \frac{\mathcal{E}[\psi_{\boldsymbol{\theta}^{i+1}}] - \mathcal{E}[\psi_{\boldsymbol{\theta}^*}]}{\mathcal{E}[\psi_{\boldsymbol{\theta}^i}] - \mathcal{E}[\psi_{\boldsymbol{\theta}^*}]} \leqslant \limsup_{i\to\infty} \left\| \boldsymbol{I} - (\boldsymbol{H}+\boldsymbol{J})^{\frac{1}{2}} \boldsymbol{P}_i^{-1} (\boldsymbol{H}+\boldsymbol{J})^{\frac{1}{2}} \right\|_2^2 \tag{28}$$

or

$$\limsup_{i\to\infty} \frac{\mathcal{E}[\psi_{\boldsymbol{\theta}^{i+1}}] - \mathcal{E}[\psi_{\boldsymbol{\theta}^*}]}{\mathcal{E}[\psi_{\boldsymbol{\theta}^i}] - \mathcal{E}[\psi_{\boldsymbol{\theta}^*}]} \leqslant \limsup_{i\to\infty} \left\| \boldsymbol{I} - \begin{pmatrix} \boldsymbol{H} & \boldsymbol{J} \\ \overline{\boldsymbol{J}} & \overline{\boldsymbol{H}} \end{pmatrix}^{\frac{1}{2}} \begin{pmatrix} \boldsymbol{P}_i & \boldsymbol{0} \\ \boldsymbol{0} & \overline{\boldsymbol{P}_i} \end{pmatrix}^{-1} \begin{pmatrix} \boldsymbol{H} & \boldsymbol{J} \\ \overline{\boldsymbol{J}} & \overline{\boldsymbol{H}} \end{pmatrix}^{\frac{1}{2}} \right\|_2^2 \tag{29}$$

in the real and complex cases, respectively, where $\boldsymbol{H} = \boldsymbol{H}(\boldsymbol{\theta}^*)$ and $\boldsymbol{J} = \boldsymbol{J}(\boldsymbol{\theta}^*)$.

Proof. See Appendix A. ∎

The convergence rate in Proposition 2 depends on a matrix $\boldsymbol{J}$ which vanishes at the ground state. Therefore, if the RGN method is applied with penalization parameters $\epsilon = \epsilon^i$ tending to infinity and wave functions $\psi_{\boldsymbol{\theta}^i}$ approaching the ground state, then the rate of convergence is *superlinear*, i.e.,

$$\limsup_{i\to\infty} \frac{\mathcal{E}[\psi_{\boldsymbol{\theta}^{i+1}}] - \mathcal{E}[\psi_{\boldsymbol{\theta}^*}]}{\mathcal{E}[\psi_{\boldsymbol{\theta}^i}] - \mathcal{E}[\psi_{\boldsymbol{\theta}^*}]} = 0. \tag{30}$$

In practice, our parametric class does not usually contain the *exact* ground state for $\mathcal{H}$, but if ϵ is large and the energy minimizer is close to the ground state, then Proposition 2 still quantifies a fast linear convergence rate for RGN. In the numerical experiments presented in Sec. V, we achieve such a fast convergence rate by gradually moving the parameter ϵ closer to zero as we make progress in optimizing the wave function.

IV. VMC SAMPLING ANALYSIS

In this section, we review VMC sampling and prove a vanishing-variance principle that quantifies the sampling error in the estimated energies and gradients. Then, we discuss challenges in VMC sampling and motivate strategies to improve the sampling.

A. VMC sampling

VMC requires quantities such as $\mathcal{E}$, $\boldsymbol{g}$, $\boldsymbol{S}$, and $\boldsymbol{H}$ that are constructed as sums or integrals over a high-dimensional or infinite-dimensional state space. To compute such quantities, VMC relies on the power of Monte Carlo sampling. In VMC, we first generate a large number of samples $\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2, \ldots, \boldsymbol{\sigma}_T$ from the normalized wave-function density

$$\rho(\boldsymbol{\sigma}) = \frac{|\psi(\boldsymbol{\sigma})|^2}{\langle \psi, \psi \rangle}, \tag{31}$$

using an appropriate Markov chain Monte Carlo (MCMC [30]) sampler. Then we approximate $\mathcal{E}$, $\boldsymbol{g}$, $\boldsymbol{S}$, and $\boldsymbol{H}$ using the following estimators:

$$\hat{\mathcal{E}} = \mathbb{E}_{\hat{\rho}}[E_L(\boldsymbol{\sigma})], \tag{32}$$

$$\hat{\boldsymbol{g}}_i = \mathrm{cov}_{\hat{\rho}}[\boldsymbol{v}_i(\boldsymbol{\sigma}), E_L(\boldsymbol{\sigma})], \tag{33}$$

$$\hat{\boldsymbol{S}}_{ij} = \mathrm{cov}_{\hat{\rho}}[\boldsymbol{v}_i(\boldsymbol{\sigma}), \boldsymbol{v}_j(\boldsymbol{\sigma})], \tag{34}$$

$$\hat{\boldsymbol{H}}_{ij} = \mathrm{cov}_{\hat{\rho}}[\boldsymbol{v}_i(\boldsymbol{\sigma}), E_{L,j}(\boldsymbol{\sigma})] - \hat{\boldsymbol{g}}_i \mathbb{E}_{\hat{\rho}}[\boldsymbol{v}_j(\boldsymbol{\sigma})] - \hat{\mathcal{E}}\hat{\boldsymbol{S}}_{ij}. \tag{35}$$

Here, $\mathbb{E}_{\hat{\rho}}$ and $\mathrm{cov}_{\hat{\rho}}$ denote expectations and covariances with respect to the empirical measure

$$\hat{\rho} = \frac{1}{T}\sum_{t=1}^{T} \delta_{\boldsymbol{\sigma}_t}, \tag{36}$$

and we have introduced the functions

$$E_L(\boldsymbol{\sigma}) = \frac{\mathcal{H}\psi(\boldsymbol{\sigma})}{\psi(\boldsymbol{\sigma})}, \quad E_{L,i}(\boldsymbol{\sigma}) = \frac{\mathcal{H}\partial_{\boldsymbol{\theta}_i}\psi(\boldsymbol{\sigma})}{\psi(\boldsymbol{\sigma})}, \tag{37}$$

$$\boldsymbol{v}_i(\boldsymbol{\sigma}) = \frac{\partial_{\boldsymbol{\theta}_i}\psi(\boldsymbol{\sigma})}{\psi(\boldsymbol{\sigma})}. \tag{38}$$

The functions E_L and $\boldsymbol{v}_i$ are known as the local energy and logarithmic derivative, respectively.

Next we state the vanishing-variance principle, which quantifies the asymptotic variance of several VMC estimators of interest.

Proposition 3. Assume the MCMC sampler is geometrically ergodic with respect to ρ, and for some $\epsilon > 0$, $\mathbb{E}_\rho|E_L(\boldsymbol{\sigma})|^{4+\epsilon} < \infty$ and $\sup_i \mathbb{E}_\rho|v_i(\boldsymbol{\sigma})|^{4+\epsilon} < \infty$. Then, as $T \to \infty$,

$$\sqrt{T}(\hat{\mathcal{E}}_T - \mathcal{E}) \xrightarrow{\mathcal{D}} \mathcal{N}(0, v^2), \tag{39}$$

$$\sqrt{T}(\hat{\boldsymbol{g}}_T - \boldsymbol{g}) \xrightarrow{\mathcal{D}} \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}), \tag{40}$$

where the asymptotic variances v^2 and $\boldsymbol{\Sigma}$ are given by

$$v^2 = \sum_{t=0}^{\infty} \mathrm{cov}_{\boldsymbol{\sigma}_0\sim\rho}[E_L(\boldsymbol{\sigma}_0), E_L(\boldsymbol{\sigma}_t)] + \sum_{t=1}^{\infty} \mathrm{cov}_{\boldsymbol{\sigma}_0\sim\rho}[E_L(\boldsymbol{\sigma}_t), E_L(\boldsymbol{\sigma}_0)], \tag{41}$$

$$\boldsymbol{\Sigma}_{ij} = \sum_{t=0}^{\infty} \mathrm{cov}_{\boldsymbol{\sigma}_0\sim\rho}[\boldsymbol{g}_i'(\boldsymbol{\sigma}_0), \boldsymbol{g}_j'(\boldsymbol{\sigma}_t)] + \sum_{t=1}^{\infty} \mathrm{cov}_{\boldsymbol{\sigma}_0\sim\rho}[\boldsymbol{g}_i'(\boldsymbol{\sigma}_t), \boldsymbol{g}_j'(\boldsymbol{\sigma}_0)], \tag{42}$$

and $\boldsymbol{g}'$ is defined as

$$\boldsymbol{g}'(\boldsymbol{\sigma}) = \overline{\{\boldsymbol{v}(\boldsymbol{\sigma}) - \mathbb{E}_{\boldsymbol{\sigma}'\sim\rho}[\boldsymbol{v}(\boldsymbol{\sigma}')]\}}[E_L(\boldsymbol{\sigma}) - \mathcal{E}]. \tag{43}$$

Proof. See Appendix A. ■

As a major takeaway from Proposition 3, the energy and energy gradient both have zero variance, i.e., $v^2 = 0$ and $\boldsymbol{\Sigma} = \mathbf{0}$, when the local energy E_L is constant, as occurs at any eigenstate of $\mathcal{H}$. Proposition 3 can thus be viewed as a robust and quantitative extension of the classic zero-variance principle [1,2] of VMC. The vanishing-variance principle is robust, since it holds when the wave function is not an eigenstate, and quantitative, since it gives a precise formula for the asymptotic variance of the energy and energy gradient estimators.

B. Improving estimation quality

Near an eigenstate, the vanishing-variance principle ensures the relative accuracy of VMC estimated energies. However, away from an eigenstate, VMC estimated energies and energy gradients can have a high variance and change erratically over the course of VMC estimation [11,24]. Therefore, variance reduction strategies are needed to ensure VMC's success.

Proposition 3 suggests three strategies for reducing the variance. The first strategy is to increase the number of Monte Carlo samples. We can do this either by running one MCMC sampler for a long time or by running many MCMC samplers in parallel and combining samples. The parallel sampling approach often leads to computational advantages, since vectorized code runs quickly on modern computers and MCMC samplers can be run on multiple nodes/cores to further cut down on the runtime. In the numerical experiments in Sec. V, we run 50 MCMC samplers per CPU core and use 48 CPU cores, thus generating 2400 parallel MCMC samplers.

The second variance reduction strategy is to reduce correlations among the samples $\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2, \ldots, \boldsymbol{\sigma}_T$ by applying a fast-mixing MCMC method such as parallel tempering [12]. Parallel tempering introduces interacting MCMC samplers that target different densities

$$\rho_i(\boldsymbol{\sigma}) \propto \rho(\boldsymbol{\sigma})^{i/m}, \quad i = 0, 1, \ldots, m. \tag{44}$$

Periodically, the samplers targeting adjacent densities ρ_i and ρ_{i+1} swap positions according to a Metropolis acceptance probability [31], which improves the mixing time for each of the samplers. Last, the samplers targeting ρ_m are used for estimating $\mathcal{E}$, $\boldsymbol{g}$, $\boldsymbol{S}$, and $\boldsymbol{H}$. Parallel tempering has reduced correlations in challenging VMC test problems in the past [11,32], and in Sec. V B we apply parallel tempering to improve the sampling for XXZ models on large lattices.

The third variance reduction strategy is to directly alter the VMC update formula to improve its stability. For example, Refs. [5,6] use an alternative gradient estimator in which the most extreme local energy values are adjusted to be closer to the median. Similarly, [4] rounds all positive gradient entries to $+1$, round all negative gradient entries to -1, and then assign a random independent magnitude to each entry. The approaches [4–6] all improve the stability of VMC updates, but they discard gradient information that could potentially be helpful. Therefore, we adopt a slightly different stabilization approach in Sec. V. At each iteration, we check that the parameter update is less than twice as large as the previous parameter update (in Euclidean norm). If not, then we shrink ϵ in half repeatedly until the parameter update is sufficiently small. This stabilization code eliminates the most erratic parameter updates in our experiments. The code is rarely triggered for TFI models (just 0–2 times per 1000 updates), but it is more commonly triggered for XXZ models (9–34 times per 1000 updates).

V. NUMERICAL EXPERIMENTS

To test the performance of VMC optimization and sampling methods, we estimate the ground-state energies for the transverse-field Ising (TFI) and XXZ models on 1D and 2D lattices with periodic boundary conditions. These models are specified by the Hamiltonians

$$\mathcal{H}_{\mathrm{TFI}} = -\sum_{i\sim j} \sigma_i^z \sigma_j^z - h \sum_i \sigma_i^x, \tag{45}$$

$$\mathcal{H}_{XXZ} = -\Delta \sum_{i\sim j} \sigma_i^z \sigma_j^z + \sum_{i\sim j} \left[\sigma_i^y \sigma_j^y - \sigma_i^x \sigma_j^x\right], \tag{46}$$

where $h > 0$ and $\Delta > 0$ are positive-valued parameters. The XXZ model is sometimes alternatively defined as

$$\mathcal{H}_{XXZ} = \Delta \sum_{i\sim j} \sigma_i^z \sigma_j^z + \sum_{i\sim j} \left[\sigma_i^x \sigma_j^x + \sigma_i^y \sigma_j^y\right], \tag{47}$$

which is a unitary transformation of Eq. (46), assuming a bipartite lattice. As a consequence of the Perron-Frobenius theorem and translational symmetry, the models (45) and (46) both admit unique, nonnegative, translationally invariant ground-state wave functions. For 1D lattices but not 2D lattices, the ground-state wave functions are known exactly [33,34].

As a wave-function ansatz, we use a restricted Boltzmann machine (RBM), which can be written as

$$\psi_{\boldsymbol{w},\boldsymbol{b}}(\boldsymbol{\sigma}) = \prod_{i=1}^{\alpha} \prod_{\mathcal{T}} \cosh\left[\sum_j \boldsymbol{w}_{ij} (\mathcal{T}\boldsymbol{\sigma})_j + \boldsymbol{b}_i\right]. \tag{48}$$

Here, α is the hidden-variable density that controls the number of parameters, $\mathcal{T}$ ranges over the translation operators on the periodic lattice, and $\boldsymbol{w}$ and $\boldsymbol{b}$ are vectors of complex-valued parameters, called *weights* and *biases*. This ansatz is an example of a two-layer neural network and is a simplification of the RBM ansatz used for VMC optimization in Ref. [3]. The ansatz involves $\alpha(n+1)$ parameters, where n is the number of spins and we set $\alpha = 5$ for all of our numerical experiments. We report additional implementation details in Appendix B.

A. Comparing optimization methods

To compare different VMC optimizers in the noiseless setting, we apply VMC to TFI model on a 10×1 lattice, which is small enough so that $\mathcal{E}$, $\boldsymbol{g}$, $\boldsymbol{S}$, and $\boldsymbol{H}$ can be computed by exact summation without the need for Monte Carlo sampling. Figure 2 evaluates the performance of different optimizers in this setting, i.e., with deterministic parameter updates. The figure shows that RGN leads to faster convergence and lower errors than GD, natural GD, and the linear method.

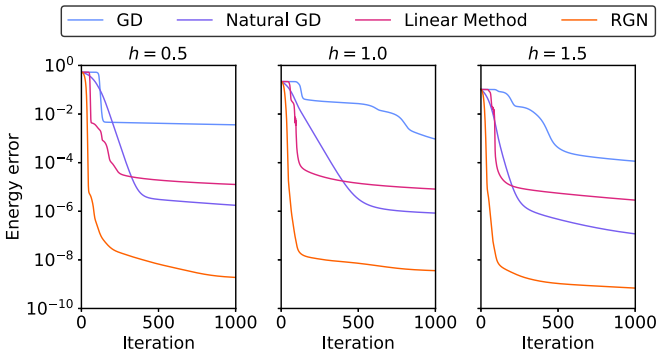

FIG. 2. RGN achieves low energy errors in ferromagnetic ($h = 0.5$, left), transitional ($h = 1.0$, center), and paramagnetic ($h = 1.5$, right) regimes. Plot shows relative error in ground-state energy estimates.

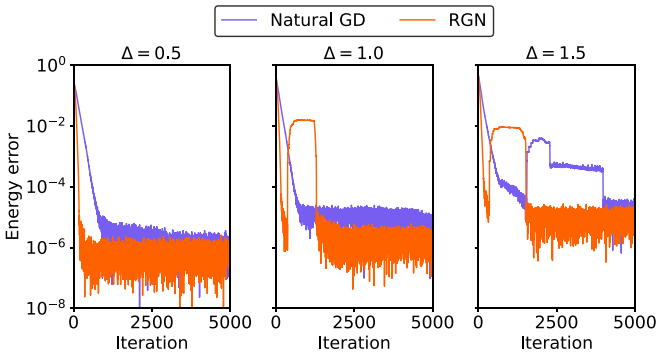

FIG. 4. VMC can lead to energy spikes if direct MCMC sampling is used. Plot shows relative error in ground-state energy estimates for an XXZ model on a 100×1 lattice.

In light of Proposition 2, we expect the most rapid energy convergence when the preconditioner is close to the true Hessian $\begin{pmatrix} \boldsymbol{H} & \boldsymbol{J} \\ \boldsymbol{J} & \boldsymbol{H} \end{pmatrix}$. Indeed, Fig. 3 confirms that the Hessian approximation used in RGN closely approximates the true Hessian, in concordance with the fast observed convergence rate.

B. Challenges in VMC sampling

We next apply VMC to larger lattices by incorporating MCMC sampling. For the TFI model, we initialize the MCMC samplers from a configuration chosen uniformly at random and propose random updates based on flipping a single spin. For the XXZ model, we confine the MCMC samplers to "balanced" configurations for which the magnetization is the same on both components of the bipartite lattice, since the ground-state wave function is only supported on these configurations. We initialize from a random balanced configuration and propose random balanced updates based on flipping two spins.

At every new optimization step, the MCMC samplers are continued from the final configurations at the previous step. The MCMC samplers are then run for $20 \times n$ time steps, and the local energies and logarithmic derivatives are evaluated at intervals of n time steps, where n denotes the number of spins.

The MCMC samplers are guaranteed to mix quickly when sampling the ground-state wave functions for the TFI model at $h = \infty$ or the XXZ model at $\Delta = -1$. For these extreme parameter settings, every Metropolis proposal is accepted, the relaxation time for the TFI sampler is $n/2$ [35], and the relaxation time for the XXZ sampler is $n/4$ [36]. Yet, there is no guarantee that the MCMC samplers remain efficient for the h and Δ values more reasonably encountered.

Indeed, the RGN and natural GD optimizers encounter difficulties when calculating ground-state energies for the XXZ model, as shown in Fig. 4. Initially during the optimization, the RGN energies decrease quickly, but at iteration 360 ($\Delta = 1.5$) or 370 ($\Delta = 1.0$), the energies exhibit a large spike, which persists over roughly 1000 optimization steps. The natural GD energies exhibit a spike later, during iterations 1500–4000 ($\Delta = 1.5$), which makes sense because the natural GD optimizer converges more slowly than the RGN optimizer overall.

The energy spikes are a major difference between exact VMC energies and energy estimates from MCMC sampling. The exact energies change slowly and continuously, as seen in Fig. 2. However, the MCMC energy estimates can spike if a slowly mixing MCMC sampler moves between metastable regions of configuration space. Indeed, Fig. 5 establishes that all 2400 MCMC samples typically lie in the ferromagnetic region of configuration space. At the onset of the energy spikes, a few MCMC samplers (5–30) enter the antiferromagnetic

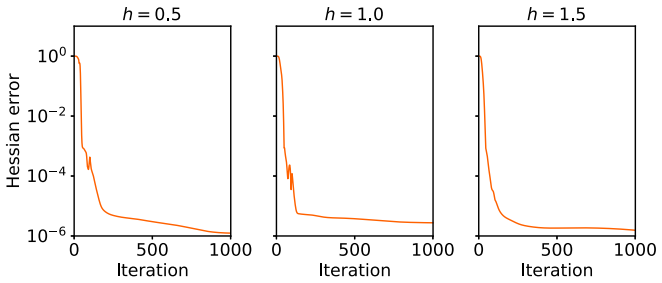

FIG. 3. RGN achieves accurate Hessian approximations with relative errors $<10^{-5}$ for most iterations. Plot shows relative error $|\begin{pmatrix} \boldsymbol{0} & \boldsymbol{J} \\ \boldsymbol{J} & \boldsymbol{0} \end{pmatrix}|/|\begin{pmatrix} \boldsymbol{H} & \boldsymbol{J} \\ \boldsymbol{J} & \boldsymbol{H} \end{pmatrix}|$ computed at each iteration.

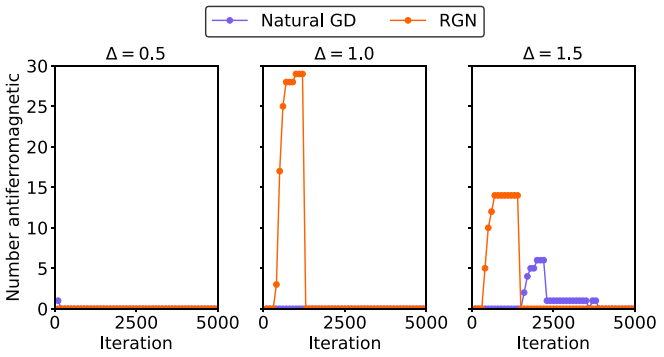

FIG. 5. Energy spikes occur when a few MCMC samplers enter the antiferromagnetic region of configuration space, defined by $\sum_{i\sim j} \sigma_i \sigma_j < 0$. Plot shows number of samplers in the antiferromagnetic region, evaluated at every 100 iterations.

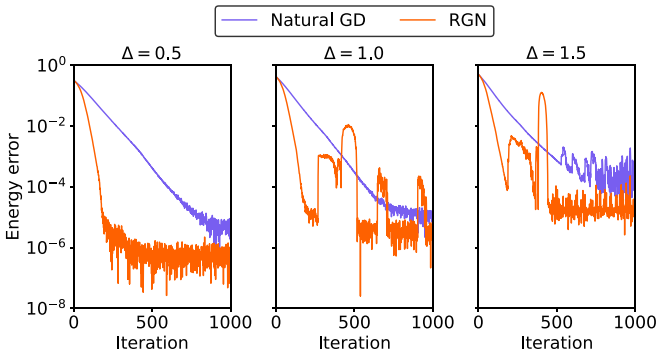

FIG. 6. VMC recovers more quickly from energy spikes if parallel tempering is used.

region of configuration space, encountering high densities and high local energies that have not been experienced before. The densities and local energies are extremely large due to generalization error, and the optimizers require 1000+ iterations to respond to the new MCMC data and eliminate the spikes.

To improve the sampling for XXZ models, we apply the parallel tempering method described in Sec. IV B using fifty target densities. Parallel tempering speeds up the mixing of the MCMC samplers and reduces the magnitude and longevity of the energy spikes, as shown in Fig. 6. With parallel tempering, the same number of MCMC samples (2400×20) are generated per iteration as in direct MCMC sampling, but the quality of the samples is much higher. The high-quality sampling reduces the generalization error and improves the overall stability of VMC.

C. Results for larger systems

Lastly, we apply VMC to estimate ground-state energies for TFI and XXZ models on lattices with up to 400 spins. We train highly accurate VMC wave functions for these large lattices by using RGN and (for XXZ models) parallel tempering. For 1D systems, we compare the estimated ground-state energies against the exact energies in Table II. For 2D systems, we report the estimated energies themselves in Table III.

Summarizing Tables II and III, we find that RGN energies converge more quickly and achieve greater accuracy than natural GD energies. In 1D lattices, RGN is more accurate than natural GD by up to four orders of magnitude, reaching error levels as low as 1.0×10^{-9} and 1.6×10^{-9}. In 2D lattices for which exact reference energies are not available, the energy estimates obtained by RGN are typically lower than those obtained using natural GD, and the convergence is very fast. After 200 iterations, RGN is converged to 4–6 significant digits, whereas natural GD is only converged to 1–4 significant digits.

We further illustrate the comparison between natural GD and RGN for TFI models in Figs. 7 and 8. These figures, showing the complete time series of energy estimates over 1000 optimization steps, demonstrate that RGN results after 200 iterations are typically more accurate than natural GD results after 1000 iterations. Because RGN is only slightly more expensive than natural GD (less than a factor of two in our experiments), we conclude that RGN makes it possible to obtain accurate ground-state estimates with reduced training time and computational cost.

TABLE II. Relative errors in ground-state energy estimates after 1000 iterations. Lower errors are marked in bold.

	200×1 TFI model		
	$h = 0.5$	$h = 1.0$	$h = 1.5$
Natural GD	3.9×10^{-5}	1.4×10^{-4}	8.5×10^{-8}
RGN	$\mathbf{1.0 \times 10^{-9}}$	$\mathbf{2.9 \times 10^{-6}}$	$\mathbf{1.6 \times 10^{-9}}$
	100×1 XXZ model		
	$\Delta = 0.5$	$\Delta = 1.0$	$\Delta = 1.5$
Natural GD	3.9×10^{-6}	1.2×10^{-5}	5.4×10^{-5}
RGN	$\mathbf{2.5 \times 10^{-7}}$	$\mathbf{3.3 \times 10^{-6}}$	$\mathbf{2.0 \times 10^{-5}}$

TABLE III. Ground-state energy estimates, normalized by the number of sites. Changes between iteration 200 and iteration 1000 are marked in bold.

	20×20 TFI model, Natural GD		
	$h = 2.0$	$h = 3.0$	$h = 4.0$
Iteration 200	$-2.\mathbf{3375353}$	$-3.1\mathbf{899006}$	$-4.133\mathbf{7097}$
Iteration 1000	$-2.\mathbf{5113061}$	$-3.1\mathbf{950035}$	$-4.133\mathbf{8352}$
	20×20 TFI model, RGN		
Iteration 200	-2.5113056	$-3.1949\mathbf{262}$	$-4.133\mathbf{5964}$
Iteration 1000	-2.5113069	$-3.1949\mathbf{974}$	$-4.133\mathbf{8354}$

VI. CONCLUSION

This work has analyzed VMC optimization and sampling methods, leading to both theoretical and computational advancements. First, we showed that the energy Hessian simplifies dramatically near an eigenstate, depending only on first derivatives of the wave function with respect to the parameters. Taking advantage of this simplification, we introduced a new Rayleigh-Gauss-Newton (RGN) optimizer that can achieve superlinear convergence. Second, we proved a vanishing-variance property that guarantees VMC energy estimates exhibit reduced variance near an eigenstate. This principle ensures accuracy in the energies near the ground state but not away from the ground state, so we suggested a parallel tempering approach to improve energy and gradient estimation for challenging test problems.

We highlight two opportunities for improving our optimization and sampling methods even further. First, for very large parametrizations, the linear system solve in the RGN method becomes numerically challenging. To address this difficulty, the Kronecker-factored approximate curvature method for efficient matrix inversion within natural gradient descent [5,37], in addition to the aforementioned matrix-free approach [28], could potentially be adapted to RGN. Second, while parallel tempering is a simple, broadly applicable enhanced sampling method, there exist a variety of alternative methods [38]. We anticipate that further analysis of enhanced MCMC

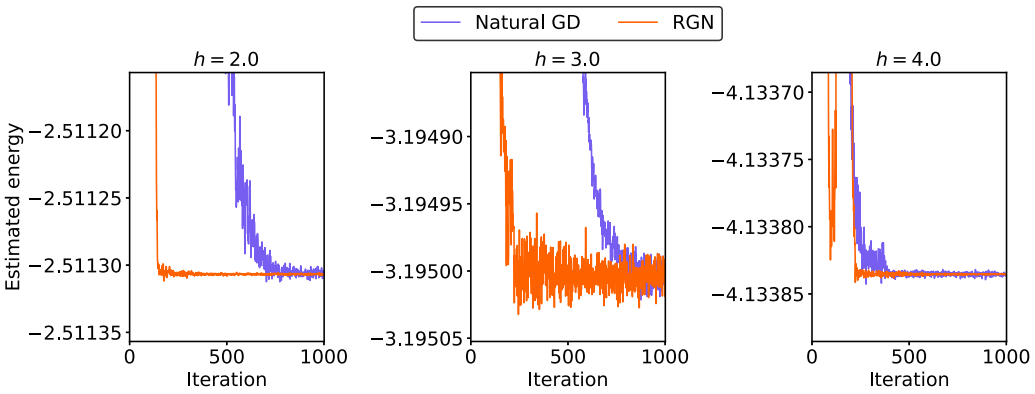

FIG. 7. RGN outperforms natural GD in ferromagnetic ($h = 0.5$, left), transitional ($h = 1.0$, center), and paramagnetic ($h = 1.5$, right) Ising models on a 20×20 lattice. Plot shows ground-state energy estimates normalized by the number of sites.

sampling will play an important role in realizing the full potential of VMC in future applications.

ACKNOWLEDGMENTS

R.J.W. and M.L. would like to acknowledge helpful conversations with Timothy Berkelbach, Aaron Dinner, Sam Greene, Lin Lin, Verena Neufeld, James Smith, Jonathan Siegel, Erik Thiede, Jonathan Weare, and Huan Zhang. R.J.W. is supported by New York University's Dean's Dissertation Fellowship and by the National Science Foundation through Award No. DMS-1646339. M.L. is supported by the National Science Foundation under Award No. DMS-1903031. The authors acknowledge support from the Advanced Scientific Computing Research Program within the DOE Office of Science through Award No. DE-SC0020427. Computing resources were provided by New York University's High Performance Computing.

APPENDIX A: PROOFS

Proof of Proposition 2. We prove only the second result. The first result is well-known, and the proof is similar. Because the wave function is analytic at $\boldsymbol{\theta}^*$ and the Wirtinger Hessian $\left(\begin{smallmatrix}\boldsymbol{H} & \boldsymbol{J} \\ \overline{\boldsymbol{J}} & \overline{\boldsymbol{H}}\end{smallmatrix}\right)$ is positive definite, as $i \to \infty$ we find

$$\begin{bmatrix} \boldsymbol{g}(\boldsymbol{\theta}^i) \\ \overline{\boldsymbol{g}(\boldsymbol{\theta}^i)} \end{bmatrix} \sim \begin{pmatrix} \boldsymbol{H} & \boldsymbol{J} \\ \overline{\boldsymbol{J}} & \overline{\boldsymbol{H}} \end{pmatrix} \begin{pmatrix} \boldsymbol{\theta}^i - \boldsymbol{\theta}^* \\ \overline{\boldsymbol{\theta}^i - \boldsymbol{\theta}^*} \end{pmatrix} \tag{A1}$$

and

$$\mathcal{E}[\psi_{\boldsymbol{\theta}^i}] - \mathcal{E}[\psi_{\boldsymbol{\theta}^*}] \sim \frac{1}{2} \left| \begin{pmatrix} \boldsymbol{H} & \boldsymbol{J} \\ \overline{\boldsymbol{J}} & \overline{\boldsymbol{H}} \end{pmatrix}^{\frac{1}{2}} \begin{pmatrix} \boldsymbol{\theta}^i - \boldsymbol{\theta}^* \\ \overline{\boldsymbol{\theta}^i - \boldsymbol{\theta}^*} \end{pmatrix} \right|^2. \tag{A2}$$

The ith parameter update satisfies

$$\begin{aligned} \begin{pmatrix} \boldsymbol{\theta}^{i+1} - \boldsymbol{\theta}^* \\ \overline{\boldsymbol{\theta}^{i+1} - \boldsymbol{\theta}^*} \end{pmatrix} \sim \left[\boldsymbol{I} - \begin{pmatrix} \boldsymbol{P}_i & \boldsymbol{0} \\ \boldsymbol{0} & \overline{\boldsymbol{P}_i} \end{pmatrix}^{-1} \begin{pmatrix} \boldsymbol{H} & \boldsymbol{J} \\ \overline{\boldsymbol{J}} & \overline{\boldsymbol{H}} \end{pmatrix} \right] \begin{pmatrix} \boldsymbol{\theta}^i - \boldsymbol{\theta}^* \\ \overline{\boldsymbol{\theta}^i - \boldsymbol{\theta}^*} \end{pmatrix} \\ + \mathcal{O}(|\boldsymbol{\theta} - \boldsymbol{\theta}^*|^2), \end{aligned} \tag{A3}$$

and the energy ratio satisfies

$$\frac{\mathcal{E}[\psi_{\boldsymbol{\theta}^{i+1}}] - \mathcal{E}[\psi_{\boldsymbol{\theta}^*}]}{\mathcal{E}[\psi_{\boldsymbol{\theta}^i}] - \mathcal{E}[\psi_{\boldsymbol{\theta}^*}]} \sim \frac{\left| \left[\boldsymbol{I} - \begin{pmatrix} \boldsymbol{H} & \boldsymbol{J} \\ \overline{\boldsymbol{J}} & \overline{\boldsymbol{H}} \end{pmatrix}^{\frac{1}{2}} \begin{pmatrix} \boldsymbol{P}^i & \boldsymbol{0} \\ \boldsymbol{0} & \overline{\boldsymbol{P}^i} \end{pmatrix}^{-1} \begin{pmatrix} \boldsymbol{H} & \boldsymbol{J} \\ \overline{\boldsymbol{J}} & \overline{\boldsymbol{H}} \end{pmatrix}^{\frac{1}{2}} \right] \begin{pmatrix} \boldsymbol{H} & \boldsymbol{J} \\ \overline{\boldsymbol{J}} & \overline{\boldsymbol{H}} \end{pmatrix}^{\frac{1}{2}} \begin{pmatrix} \boldsymbol{\theta}_i - \boldsymbol{\theta}^* \\ \overline{\boldsymbol{\theta}^i - \boldsymbol{\theta}^*} \end{pmatrix} \right|^2}{\left| \begin{pmatrix} \boldsymbol{H} & \boldsymbol{J} \\ \overline{\boldsymbol{J}} & \overline{\boldsymbol{H}} \end{pmatrix}^{\frac{1}{2}} \begin{pmatrix} \boldsymbol{\theta}_i - \boldsymbol{\theta}^* \\ \overline{\boldsymbol{\theta}^i - \boldsymbol{\theta}^*} \end{pmatrix} \right|^2}, \tag{A4}$$

whence

$$\begin{aligned} &\limsup_{i \to \infty} \frac{\mathcal{E}[\psi_{\boldsymbol{\theta}^{i+1}}] - \mathcal{E}[\psi_{\boldsymbol{\theta}^*}]}{\mathcal{E}[\psi_{\boldsymbol{\theta}^i}] - \mathcal{E}[\psi_{\boldsymbol{\theta}^*}]} \\ &\leqslant \limsup_{i \to \infty} \left\| \boldsymbol{I} - \begin{pmatrix} \boldsymbol{H} & \boldsymbol{J} \\ \overline{\boldsymbol{J}} & \overline{\boldsymbol{H}} \end{pmatrix}^{\frac{1}{2}} \begin{pmatrix} \boldsymbol{P}_i & \boldsymbol{0} \\ \boldsymbol{0} & \overline{\boldsymbol{P}_i} \end{pmatrix}^{-1} \begin{pmatrix} \boldsymbol{H} & \boldsymbol{J} \\ \overline{\boldsymbol{J}} & \overline{\boldsymbol{H}} \end{pmatrix}^{\frac{1}{2}} \right\|_2^2. \end{aligned} \tag{A5}$$

■

Proof of Proposition 3. The Markov chain central limit theorem [39] guarantees that

$$\frac{1}{T} \sum_{t=1}^{T} \begin{pmatrix} \overline{\boldsymbol{v}(\boldsymbol{\sigma}_t)} \\ E_L(\boldsymbol{\sigma}_t) \end{pmatrix} = \begin{pmatrix} \mathbb{E}_\rho[\,\overline{\boldsymbol{v}(\boldsymbol{\sigma})}\,] \\ \mathcal{E} \end{pmatrix} + \mathcal{O}_p\left(\frac{1}{\sqrt{T}} \right) \tag{A6}$$

as $T \to \infty$. Next, using the identity

$$\begin{aligned} &\{\boldsymbol{x} - \mathbb{E}_\rho[\,\overline{\boldsymbol{v}(\boldsymbol{\sigma})}\,]\}(y - \mathcal{E}) \\ &\quad = \mathcal{O}\{|\boldsymbol{x} - \mathbb{E}_\rho[\,\overline{\boldsymbol{v}(\boldsymbol{\sigma})}\,]|^2 + |y - \mathcal{E}|^2\}, \end{aligned} \tag{A7}$$

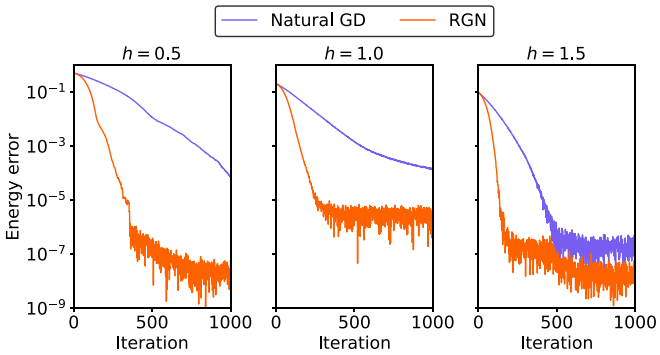

FIG. 8. Relative error in ground-state energy estimates for TFI models on a 200 × 1 lattice.

and substituting the empirical averages $\boldsymbol{x} = \frac{1}{T}\sum_{t=1}^{T} \overline{\boldsymbol{v}(\boldsymbol{\sigma}_t)}$ and $y = \frac{1}{T}\sum_{t=1}^{T} E_L(\boldsymbol{\sigma}_t)$, we obtain

$$\hat{\boldsymbol{g}}_T \tag{A8}$$

$$= \frac{1}{T}\sum_{t=1}^{T} \overline{\boldsymbol{v}(\boldsymbol{\sigma}_t)} E_L(\boldsymbol{\sigma}_t) - \frac{1}{T^2}\sum_{s,t=1}^{T} \overline{\boldsymbol{v}(\boldsymbol{\sigma}_s)} E_L(\boldsymbol{\sigma}_t) \tag{A9}$$

$$= \frac{1}{T}\sum_{t=1}^{T} \boldsymbol{g}'(\boldsymbol{\sigma}_t) + \mathcal{O}_p\left(\frac{1}{T}\right). \tag{A10}$$

Slutsky's lemma shows that the $\mathcal{O}_p(\frac{1}{T})$ term is asymptotically negligible, and another application of the Markov chain central limit theorem guarantees

$$\sqrt{T}(\hat{\mathcal{E}}_T - \mathcal{E}) \xrightarrow{\mathcal{D}} \mathcal{N}(0, v^2), \tag{A11}$$

$$\sqrt{T}(\hat{\boldsymbol{g}}_T - \boldsymbol{g}) \xrightarrow{\mathcal{D}} \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}), \tag{A12}$$

where the asymptotic variances v^2 and $\boldsymbol{\Sigma}$ are as given in Eqs. (41) and (42). ■

APPENDIX B: COMPUTATIONS

Here, we discuss the computational details for our experiments. These computations are implemented using the JAX library for Python [40], and complete scripts and output are available on github [41]. Using these scripts, estimating the ground-state wave function for a large lattice is relatively fast, requiring less than four days on a single 48-core CPU node (see Table IV). The resulting energies are presented in Table II for TFI models and Table III for XXZ models.

We initialize our neural network wave-function parameters as independent complex-valued $\mathcal{N}(0, 0.001)$ random variables, using a random seed of 123. We then update our parameters using GD, natural GD, LM, or RGN over 1000 iterations, as described in Secs. III B and III C. During the optimizations, we increase the penalization parameter ϵ from $\epsilon = \epsilon_{\min}$ to $\epsilon = \epsilon_{\max}$ and increase η from $\eta = \eta_{\min}$ to $\eta = \eta_{\max}$ at a geometric rate over τ iterations. Our specific choices of parameters $\epsilon_{\min}$, $\epsilon_{\max}$, $\eta_{\min}$, $\eta_{\max}$, and τ are detailed below in Table V.

TABLE IV. Runtimes per 1000 optimization steps on a single 48-core CPU node, with 2400 × 20 MCMC samples per optimization step.

	RGN	Natural GD
TFI 200 × 1	18–21 h	12–14 h
TFI 20 × 20	58–63 h	30–32 h
XXZ 100 × 1	97–100 h	85–90 h

TABLE V. Penalization parameters.

$\epsilon_{\min}$	0.001
$\epsilon_{\max}$	0.01 for GD and natural GD, 1 for LM, 1000 for RGN
$\eta_{\min}$	0.001
$\eta_{\max}$	0.001 for natural GD, 0.1 for RGN
τ	100 for deterministic updates, 500 for stochastic updates

Before evaluating the wave function $\psi(\boldsymbol{\sigma})$ or wave function derivative $\psi_i(\boldsymbol{\sigma})$, we check whether $\boldsymbol{\sigma}$ has "mostly negative" magnetization, as defined by

$$2\sum_i \boldsymbol{\sigma}_i + \boldsymbol{\sigma}_1 < 0. \tag{B1}$$

If we encounter a configuration $\boldsymbol{\sigma}$ for which Eq. (B1) is not satisfied, we transform it to $-\boldsymbol{\sigma}$. Indeed, the mostly negative configurations suffice to determine the complete wave function given the symmetry condition $\psi(\boldsymbol{\sigma}) = \psi(-\boldsymbol{\sigma})$, and VMC can lead to low-quality ground-state wave-function estimates when this symmetry condition is not enforced.

For 1D and 2D lattices, we can evaluate the log wave function and its derivatives in $\mathcal{O}(\alpha n \log n)$ operations using the discrete Fourier transform $\mathcal{F}$ and its inverse $\mathcal{F}^{-1}$. To show this, we write

$$\log \psi_{\boldsymbol{w},\boldsymbol{b}}(\boldsymbol{\sigma}) = \sum_{i=1}^{\alpha}\sum_j \log\cosh \boldsymbol{\theta}_{ij}, \tag{B2}$$

where we have introduced angles

$$\boldsymbol{\theta}_{ij} = [\mathcal{F}^{-1}(\mathcal{F}\boldsymbol{w}_{i\cdot} \odot \overline{\mathcal{F}\boldsymbol{\sigma}})]_j + \boldsymbol{b}_i \tag{B3}$$

and we have used $\odot$ to represent element-wise multiplication. Similarly, we write

$$\frac{\partial \log \psi_{\boldsymbol{w},\boldsymbol{b}}}{\partial \boldsymbol{b}_i}(\boldsymbol{\sigma}) = \sum_j \tanh \boldsymbol{\theta}_{ij}, \tag{B4}$$

$$\frac{\partial \log \psi_{\boldsymbol{w},\boldsymbol{b}}}{\partial \boldsymbol{w}_{ij}}(\boldsymbol{\sigma}) = \{\mathcal{F}^{-1}[\mathcal{F}(\tanh \boldsymbol{\theta}_{i\cdot}) \odot \mathcal{F}\boldsymbol{\sigma}]\}_j. \tag{B5}$$

When optimizing VMC wave functions, we occasionally encounter a sudden increase in the norm of the parameter updates, here defined as a factor of two or greater. When such a large update occurs, in addition to immediately restricting the size of the parameter update (by decreasing ϵ), we restore $\epsilon = \epsilon_{\min}$ and $\eta = \eta_{\min}$ and restart the geometric progression.

Last, to obtain the energy estimates reported in Tables II and III, we run the MCMC chains for an additional 2000 × n time steps and evaluate the local energies at intervals of n time steps.

[1] J. Gubernatis, N. Kawashima, and P. Werner, *Quantum Monte Carlo Methods* (Cambridge University Press, Cambridge, UK, 2016).

[2] F. Becca and S. Sorella, *Quantum Monte Carlo Approaches for Correlated Systems* (Cambridge University Press, Cambridge, UK, 2017).

[3] G. Carleo and M. Troyer, Solving the quantum many-body problem with artificial neural networks, Science **355**, 602 (2017).

[4] D. Luo and B. K. Clark, Backflow Transformations Via Neural Networks for Quantum Many-Body Wave Functions, Phys. Rev. Lett. **122**, 226401 (2019).

[5] D. Pfau, J. S. Spencer, A. G. D. G. Matthews, and W. M. C. Foulkes, *Ab initio* solution of the many-electron Schrödinger equation with deep neural networks, Phys. Rev. Research **2**, 033429 (2020).

[6] J. Hermann, Z. Schätzle, and F. Noé, Deep-neural-network solution of the electronic Schrödinger equation, Nat. Chem. **12**, 891 (2020).

[7] J. S. Spencer, D. Pfau, A. Botev, and W. M. C. Foulkes, Better, faster fermionic neural networks, arXiv:2011.07125.

[8] O. Sharir, Y. Levine, N. Wies, G. Carleo, and A. Shashua, Deep Autoregressive Models for the Efficient Variational Simulation of Many-Body Quantum Systems, Phys. Rev. Lett. **124**, 020503 (2020).

[9] L. Yang, Z. Leng, G. Yu, A. Patel, W.-J. Hu, and H. Pu, Deep learning-enhanced variational Monte Carlo method for quantum many-body physics, Phys. Rev. Research **2**, 012039(R) (2020).

[10] L. Yang, W. Hu, and L. Li, Scalable variational Monte Carlo with graph neural ansatz, arXiv:2011.12453.

[11] C.-Y. Park and M. J. Kastoryano, Geometry of learning neural quantum states, Phys. Rev. Research **2**, 023232 (2020).

[12] R. H. Swendsen and J.-S. Wang, Replica Monte Carlo Simulation of Spin-Glasses, Phys. Rev. Lett. **57**, 2607 (1986).

[13] S. Sachdev, *Quantum Phase Transitions* (Cambridge University Press, Cambridge, UK, 2009).

[14] Á. Szabados, Perturbation theory—Time-independent aspects of the theory applied in molecular electronic structure description, in *Elsevier Reference Module in Chemistry, Molecular Sciences, and Chemical Engineering* (Elsevier, Amsterdam, 2016).

[15] W. Wirtinger, Zur formalen theorie der funktionen von mehr komplexen veränderlichen, Math. Ann. **97**, 357 (1927).

[16] P. J. Schreier and L. L. Scharf, Complex differential calculus (Wirtinger calculus), in *Statistical Signal Processing of Complex-Valued Data: The Theory of Improper and Noncircular Signals* (Cambridge University Press, Cambridge, UK, 2010), pp. 277–286.

[17] S. Sorella, Generalized Lanczos algorithm for variational quantum Monte Carlo, Phys. Rev. B **64**, 024512 (2001).

[18] S. Sorella, M. Casula, and D. Rocca, Weak binding between two aromatic rings: Feeling the van der Waals attraction by quantum Monte Carlo methods, J. Chem. Phys. **127**, 014105 (2007).

[19] J. Stokes, J. Izaac, N. Killoran, and G. Carleo, Quantum natural gradient, Quantum **4**, 269 (2020).

[20] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, in *Proceedings of the 3rd International Conference on Learning Representations (ICLR'15), San Diego, CA, May 7–9, 2015, Conference Track Proceedings*, edited by Y. Bengio and Y. LeCun (IEEE, Piscataway, NJ, 2015).

[21] S. J. Reddi, S. Kale, and S. Kumar, On the convergence of Adam and beyond, in *Proceedings of the International Conference on Learning Representations* (IEEE, Piscataway, NJ, 2018).

[22] I. Sabzevari, A. Mahajan, and S. Sharma, An accelerated linear method for optimizing non-linear wave functions in variational Monte Carlo, J. Chem. Phys. **152**, 024111 (2020).

[23] M. Hibat-Allah, M. Ganahl, L. E. Hayward, R. G. Melko, and J. Carrasquilla, Recurrent neural network wave functions, Phys. Rev. Research **2**, 023358 (2020).

[24] T. Westerhout, N. Astrakhantsev, K. S. Tikhonov, M. I. Katsnelson, and A. A. Bagrov, Generalization properties of neural network approximations to frustrated magnet ground states, Nat. Commun. **11**, 1593 (2020).

[25] D. Wierichs, C. Gogolin, and M. Kastoryano, Avoiding local minima in variational quantum eigensolvers with the natural gradient optimizer, Phys. Rev. Research **2**, 043246 (2020).

[26] J. Nocedal and S. Wright, *Numerical Optimization* (Springer Science & Business Media, Berlin, 2006).

[27] A. Cuzzocrea, A. Scemama, W. J. Briels, S. Moroni, and C. Filippi, Variational principles in quantum Monte Carlo: The troubled story of variance minimization, J. Chem. Theory Comput. **16**, 4203 (2020).

[28] E. Neuscamman, C. J. Umrigar, and Garnet Kin-Lic Chan, Optimizing large parameter sets in variational quantum Monte Carlo, Phys. Rev. B **85**, 045103 (2012).

[29] L. Zhao and E. Neuscamman, A blocked linear method for optimizing large parameter sets in variational Monte Carlo, J. Chem. Theory Comput. **13**, 2604 (2017).

[30] J. S. Liu, *Monte Carlo Strategies in Scientific Computing* (Springer Science & Business Media, Berlin, 2008).

[31] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, Equation of state calculations by fast computing machines, J. Chem. Phys. **21**, 1087 (1953).

[32] K. Choo, G. Carleo, N. Regnault, and T. Neupert, Symmetries and Many-Body Excitations with Neural-Network Quantum States, Phys. Rev. Lett. **121**, 167204 (2018).

[33] H. Bethe, Zur theorie der metalle, Z. Phys. **71**, 205 (1931).

[34] P. Pfeuty, The one-dimensional Ising model with a transverse field, Ann. Phys. **57**, 79 (1970).

[35] D. A. Levin and Y. Peres, *Markov Chains and Mixing Times* (American Mathematical Society, Providence, RI, 2017), Vol. 107.

[36] P. Diaconis, *Group Representations in Probability and Statistics*, Institute of Mathematical Statistics Lecture Notes—Monograph Series, 11 (Institute of Mathematical Statistics, Hayward, CA, 1988).

[37] J. Martens and R. Grosse, Optimizing neural networks with Kronecker-factored approximate curvature, in *Proceedings of the 32nd International Conference on Machine Learning* (ACM, New York, NY, 2015).

[38] P. Tiwary and A. van de Walle, A review of enhanced sampling approaches for accelerated molecular dynamics, Multiscale Mater. Model. Nanomech.195 (2016).

[39] G. L. Jones *et al.*, On the Markov chain central limit theorem, Probabil. Surveys **1**, 299 (2004).

[40] J. Bradbury, R. Frostig, P. Hawkins, M. J. Johnson, C. Leary, D. Maclaurin, G. Necula, A. Paszke, J. VanderPlas, S. Wanderman-Milne, and Q. Zhang, JAX: Composable transformations of Python+NumPy programs (2018).

[41] R. J. Webber, RGN optimization, https://github.com/rjwebber/rgn_optimization (2021).