Gradient-descent optimization of fermion nodes in the diffusion Monte Carlo technique

John McFarland[1,*] and Efstratios Manousakis[1,2,†]

[1]*Department of Physics and National High Magnetic Field Laboratory, Florida State University, Tallahassee, Florida 32306, USA*

[2]*Department of Physics, National and Kapodistrian University of Athens, Panepistimioupolis, Zografos, 157 84 Athens, Greece*

(Received 4 November 2021; accepted 2 March 2022; published 17 March 2022)

We present a method for optimizing the location of the fermion ground-state nodes using a combination of diffusion Monte Carlo (DMC) and projected gradient descent (PGD). A PGD iteration shifts the parameters of an arbitrary node-fixing trial function in the opposite direction of the DMC energy gradient, while maintaining the cusp condition for atomic electrons. The energy gradient is calculated from DMC walker distributions by one of three methods we derive from an exact analytical expression. We combine our energy gradient calculation methods with different gradient-descent algorithms and a projection operator that maintains the cusp condition. We apply this stochastic PGD method to trial functions of Be, Li_2, and Ne, all consisting of a single Slater determinant with randomized parameters, and find that the nodes dramatically improve to the same DMC energy as nodes optimized by variational Monte Carlo. Our method, therefore, departs from the standard procedure of optimizing the nodes with a non-DMC scheme such as variational Monte Carlo, density-functional theory, or configuration-interaction-based calculation, which do not directly minimize the DMC energy.

DOI: 10.1103/PhysRevA.105.032815

I. INTRODUCTION

Diffusion Monte Carlo (DMC) [1–3], also known as Green's-function Monte Carlo or projector Monte Carlo, is a technique for projecting the many-body wave function to the ground state. It has been used to accurately tackle bosonic or nonfrustrated quantum-spin systems (see Ref. [4] for a list of references). In the case of fermionic systems, in general one has to restrict oneself to the fixed-node approximation, where it has been accurately applied to nuclear physics [5–7] when the required initial trial function was accurate enough. Node relaxation has led to accurate results for the electron gas in the continuum [2] and for electrons on a lattice [8–10]. There are variants of the method, such as the constraint path [11,12], which have been successfully applied to condensed-matter physics [12,13] problems.

The DMC method samples the ground-state wave function Ψ of N particles with walkers. These are points in a dN-dimensional space (d being the dimensionality of space) with a probability density equal to $\Psi\Psi_g$, where Ψ_g is a guiding function used for importance sampling. DMC projects Ψ to the ground state by propagating it along the imaginary-time axis, which is done by changing the position and weight of each walker in a way that samples the Green's function (i.e., the matrix elements of the evolution operator). To prevent large variations in weights, walkers are regularly deleted, duplicated, or combined in a way that the initial and final $\Psi\Psi_g$ are proportional.

For a fermionic Ψ, it is generally necessary to impose zero boundary conditions at an *a priori* chosen nodal surface. That is, the nodal surface of Ψ is taken to be the nodal surface of a trial function Ψ_t, which is an approximate ground-state wave function produced by a non-DMC method. Typically Ψ_t is a product of a symmetric Jastrow function [14] factor, which describes correlations, and an antisymmetric factor, which is a single Slater determinant or a combination of Slater determinants [15] that describes the nodes. It is standard to set $\Psi_t = \Psi_g$, in which case nodal boundary conditions are naturally imposed by imaginary-time propagation alone. In the present paper we distinguish Ψ_t from Ψ_g because we often use a nodeless Ψ_g, in which case nodal boundary conditions are imposed by deleting those walkers that attempt to cross the nodes.

DMC error is improved with a Ψ_t that better approximates the exact ground-state wave function. The only source of error that cannot be eliminated with the DMC propagation is the fixed-node error, which is contained in just the antisymmetric part of Ψ_t. The other sources of error, namely, the finite time-step error, statistical error, and population control error, can be respectively controlled with a smaller time step, a greater sample number, and a larger walker population.

This makes the choice of the nodes of Ψ_t the fundamental approximation of DMC. The resulting fixed-node error typically ranges from about 82 to 435 meV per atom [16], and is generally controlled by increasing the complexity of Ψ_t and better optimizing its parameters. Because DMC energy is an upper bound to the true ground-state energy [17], a parameter optimization method that minimizes DMC energy is the most accurate. However, optimization methods in use will minimize some other quantity, such as the variational Monte Carlo (VMC) energy, the Kohn-Sham energy, or the local VMC energy variance [18,19].

The first attempt to optimize the parameters of Ψ_t using DMC walker distributions alone was by Reboredo *et al.* [20].

*swqecs@gmail.com
†manousakis@gmail.com

They iteratively generated a new Ψ_t by projecting the coefficients of its determinants (or pfaffians) from the walker distribution of the previous Ψ_t. Mindful that it becomes expensive to evaluate a Ψ_t with an increasing number of determinants, they also proposed a method to reduce the complexity of Ψ_t by minimizing a cost function between the initial and reduced Ψ_t that heavily penalized changes of the nodes.

We propose to directly use the DMC energy as a "cost function" when optimizing the parameters of Ψ_t. This differs from using the VMC energy as a cost function [21]. Our method iteratively performs projected gradient descent (PGD) on Ψ_t by shifting its parameters in a direction roughly opposite to the energy gradient, while maintaining any constraints (the electron-nuclear cusp condition in our examples) by projecting the parameters back to the surface satisfying the constraint with each iteration. Gradient descent has an advantage for large parameter number N_p in that it does not require an estimate of $N_p \times N_p$ matrices, which is required by many others, e.g., the linear method.

We calculate the gradient of each parameter iteration from walker samples using one of three methods that we label A, B, and C. The walker distributions are produced by DMC imaginary-time propagation with nodes fixed by the parameters. We experiment with several gradient-descent algorithms commonly used for machine learning. In order to keep our method self-reliant, during PGD we do not rely on a preoptimized Jastrow function, which does not affect the fixed-node error, although it does affect the other errors.

Our paper is organized as follows: In Sec. II we present the three methods for evaluating the derivative of the energy from the DMC walker distribution. In Sec. III we describe our implementation of gradient descent, including the DMC scheme used, practical issues, and the form of Ψ_t and Ψ_g. In Sec. IV we present tests on the accuracy and speed of the three methods used and results of PGD on trial functions of Be, Li_2, Ne, and F_2. In Sec. V, we discuss the advantages of our method and ways to improve it. We discuss parameter fluctuations in Appendix A, and we describe the gradient-descent algorithms used in Appendix B and we compare them in Appendix C.

II. METHOD

A PGD iteration shifts the nodes in a direction expected to lower the DMC energy E. The nodes are determined by the parameters θ_i of the antisymmetric part of Ψ_t, and the shift of the nodes is determined by the gradient $\frac{\partial E}{\partial \theta_i}$. The gradient of a nodal surface is calculated with walker samples produced by DMC imaginary-time propagation of $\Psi\Psi_g$, with Ψ the fixed-node ground state and Ψ_g a guiding function. Thus PGD iterations change the nodal surface, while DMC iterations generate data for the gradient of a given nodal surface.

Provided the θ_i are not already at a local minimum, E will be lowered by the following change of parameters:

$$\theta_i \to \theta_i - a_i \frac{\partial E}{\partial \theta_i}, \tag{1}$$

provided that a_i is positive and sufficiently small. However, the presence of stochastic error in our gradient means that only the expectation of E will be lowered with sufficiently small a_i. If the θ_i are constrained (e.g., from the cusp condition or symmetry) we modify Eq. (1) to

$$\theta_i \to P\left(\theta_i - a_i \frac{\partial E}{\partial \theta_i}\right), \tag{2}$$

where P is the projection operator that moves the parameters to the nearest point on the manifold that satisfies the constraint. Some gradient-descent algorithms, e.g., Adam, simulate momentum with friction by replacing $\frac{\partial E}{\partial \theta_i}$ with an exponential trailing average of current and past iterations.

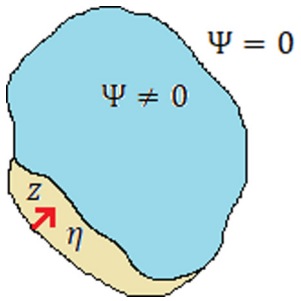

FIG. 1. Nodal pocket.

A. Derivative of the energy

Central to our method is calculating the DMC energy gradient in parameter space. Working in atomic units, we derive an expression for $\frac{\partial E}{\partial \theta_i}$, also derived by Berman [22]. To simplify the derivation, let us again set the ground state Ψ to zero except for one nodal pocket, as justified in Sec. II A. Now let us examine how the energy, given as

$$E = \frac{\langle \Psi | H | \Psi \rangle}{\langle \Psi | \Psi \rangle}, \tag{3}$$

changes due to an infinitesimal change $\delta\theta_i$ in one of the parameters, which shifts the node and, thus, changes Ψ to $\Psi + \delta\Psi$. By doing that the change in energy δE is given by

$$\delta E = \frac{\langle \delta\Psi | (H - E) | \Psi \rangle}{\langle \Psi | \Psi \rangle} + \frac{\langle \Psi | (H - E) | \delta\Psi \rangle}{\langle \Psi | \Psi \rangle} + \frac{\langle \delta\Psi | (H - E) | \delta\Psi \rangle}{\langle \Psi | \Psi \rangle}. \tag{4}$$

We convert the above expression to an integral over the $3N$ dimensional position vector $\mathbf{R}$ and integrate by parts to obtain

$$\delta E = \frac{\int d\mathbf{R} \;\; 2\delta\Psi(H - E)\Psi + \delta\Psi(H - E)\delta\Psi}{\langle \Psi | \Psi \rangle}. \tag{5}$$

The $\delta\Psi(H - E)\Psi$ term of Eq. (5) is zero except at the original nodes, where the action of the Laplacian on a discontinuity of $\nabla\Psi$ produces a delta function. The term $\delta\Psi(H - E)\delta\Psi$ of the above equation is second order in $\delta\theta_i$ except at the original and shifted nodes, where the action of the Laplacian on the discontinuities of $\nabla\delta\Psi$ produces zeroth-order delta functions. Thus, we only need to consider the operation of the Laplacian on these discontinuities at the original and shifted nodes to evaluate Eq. (5) up to first order in $\delta\theta_i$.

Let us now choose a coordinate system with the shape of the nodes. Let z be the coordinate perpendicular to the node, zero at the initial node, and pointing towards the direction where $\Psi \neq 0$ (see Fig. 1 for an illustration). Let $\mathbf{A}$ be the

remaining $3N-1$ dimensional coordinates that parametrize the nodal surface at $z=0$. Then Ψ to first order is

$$\Psi(\mathbf{A},z)=g(\mathbf{A})\mathsf{R}(z), \tag{6}$$

where $g(\mathbf{A})\equiv|\nabla\Psi(\mathbf{A},z=0^+)|$, and R is the ramp function:

$$\mathsf{R}(z)=\begin{cases} z & \text{if } 0\leqslant z \\ 0 & \text{if } z<0 \end{cases}.$$

To express $\delta\Psi$ near the node, let us define $\eta(\mathbf{A})$ as the displacement of the node in the z direction that results from $\delta\theta_i$. Then, to first order in Ψ, $\delta\Psi$ is the difference between the displaced and original wave function, that is,

$$\delta\Psi(\mathbf{A},z)=g(\mathbf{A})\mathsf{R}[z-\eta(\mathbf{A})]-g(\mathbf{A})\mathsf{R}(z). \tag{7}$$

The delta functions resulting from the Laplacian are then given by

$$\nabla^2\Psi(\mathbf{A},z)=g(\mathbf{A})\delta(z), \tag{8}$$

$$\nabla^2\delta\Psi(\mathbf{A},z)=g(\mathbf{A})\delta(z-\eta)-g(\mathbf{A})\delta(z). \tag{9}$$

Since only the delta functions contribute to first order, we leave only their contribution to Eq. (5), yielding

$$\delta E=-\frac{\int d\mathbf{R}\;\;\delta\Psi(\mathbf{A},z)g(\mathbf{A})[\delta(z)+\delta(z-\eta)]}{2\langle\Psi|\Psi\rangle}. \tag{10}$$

We reduce this to an integral over the nodal surface by integrating over z. Absorbing the Jacobian determinant into dA we obtain

$$\delta E=-\frac{\int_{\text{node}} d\mathbf{A}\;g(\mathbf{A})[\delta\Psi(\mathbf{A},0)+\delta\Psi(\mathbf{A},\eta)]}{2\langle\Psi|\Psi\rangle}. \tag{11}$$

From Eq. (7) we find that $\delta\Psi(\mathbf{A},0)+\delta\Psi(\mathbf{A},\eta)=g(\mathbf{A})\eta(\mathbf{A})$ both when $\eta(\mathbf{A})>0$ and when $\eta(\mathbf{A})<0$. This yields

$$\delta E=\frac{\int_{\text{node}} d\mathbf{A}\;\eta(\mathbf{A})g^2(\mathbf{A})}{2\langle\Psi|\Psi\rangle}. \tag{12}$$

By taking a derivative with respect to θ_i we derive our analytical expression:

$$\frac{\partial E}{\partial\theta_i}=\frac{\int_{\text{node}} d\mathbf{A}\;|\nabla\Psi(\mathbf{A})|^2\partial_{\theta_i}\eta(\mathbf{A})}{2\langle\Psi|\Psi\rangle}. \tag{13}$$

B. Practical methods to estimate the energy gradient

Starting from Eq. (13) we have derived three different equations for calculating the energy gradient from walker distributions, which we refer to as method A, B, and C. Since DMC usually samples terms linear in Ψ, but Eq. (13) contains two terms bilinear in Ψ, an approximation used by methods A and B replaces the bilinear terms with a mixed estimate of the true ground-state wave function and the trial function, i.e.,

$$\langle\Psi|\Psi\rangle\rightarrow\langle\Psi_t|\Psi\rangle, \qquad |\nabla\Psi|^2\rightarrow\nabla\Psi_t\cdot\nabla\Psi. \tag{14}$$

Then, using

$$\frac{\partial\eta}{\partial\theta_i}=-\frac{1}{|\nabla\Psi_t|}\frac{\partial\Psi_t}{\partial\theta_i}, \tag{15}$$

we approximate Eq. (13) with

$$\frac{\partial E}{\partial\theta_i}\approx-\frac{\int_{\text{node}} dA\;|\nabla\Psi(A)|\partial_{\theta_i}\Psi_t(A)}{2\langle\Psi_t|\Psi\rangle}. \tag{16}$$

Unfortunately, this relies on the quality of Ψ_t, although this quality is improved with PGD. Method C does not make this approximation, and thus in principle requires no Jastrow optimization.

C. Method A: The energy gradient from a standard walker distribution

Method A calculates Eq. (16) with a standard walker distribution equal to $\Psi_t\Psi$, which we think will make it easiest to implement into existing DMC code. It requires the evaluation of Ψ_t, its parameter derivative, and the parameter derivative of its $3N$ dimensional Laplacian. To get Eq. (16) in a form where this is possible, we change the nodal integral to a volume integral. Using the z coordinate as defined in Sec. II and Gauss's theorem, we transform Eq. (16) to

$$\frac{\partial E}{\partial\theta_i}\approx-\frac{1}{2\langle\Psi_t|\Psi\rangle}\int_{\text{node}} d\mathbf{A}\lim_{\epsilon\rightarrow0}\int_{-\epsilon}^{\epsilon}dz\,\frac{\partial\Psi_t}{\partial\theta_i}\nabla^2\Psi, \tag{17}$$

where we used the fact that $|\nabla\Psi|=\frac{\partial\Psi}{\partial z}$ for $z>0$ and $\Psi=0$ for $z<0$.

We replace ∇^2 with $2(E-H)$ in Eq. (17) in order to increase the bounds of the z integral to all space, which is possible since $(E-H)\Psi=0$ everywhere except the nodes. We then integrate by parts to obtain

$$\frac{\partial E}{\partial\theta_i}\approx\frac{\int_{\text{vol}} dR\;\;\Psi(H-E)\partial_{\theta_i}\Psi_t}{\int_{\text{vol}} dR\;\Psi_t\Psi}. \tag{18}$$

Interestingly, one can also arrive at Eq. (18) by taking the derivative of the mixed estimator $\frac{\langle\Psi|H|\Psi_t\rangle}{\langle\Psi|\Psi_t\rangle}$ with Ψ fixed.

Method A calculates the integrals of Eq. (18) from positions R_k and weights w_k of a walker distribution equal to $\Psi_t\Psi$ with the equation

$$\frac{\partial E}{\partial\theta_i}\approx\frac{\sum_k w_k\Psi_t^{-1}(R_k)(H-E)\partial_{\theta_i}\Psi_t(R_k)}{\sum_k w_k}. \tag{19}$$

D. Method B: The energy gradient from a nodal walker distribution

Method B requires the evaluation of Ψ_t and its parameter derivative. It samples $|\nabla\Psi|$ of Eq. (16) directly from a walker distribution on the nodal surface, which we generate by recording walkers that cross the nodal surface and are thus deleted. We note that shifting crossed walker positions closer to the nodes using Newton's method noticeably improves performance. Unlike those of method A, the precrossed walkers cannot be guided by Ψ_t. This is partly because no walkers will cross the nodes of Ψ_t in the zero time-step limit. Instead we guide them with a nodeless function Ψ_g.

We wish to relate $|\nabla\Psi|$ at the nodal surface to the number of walkers that cross the nodal surface per area dF/dA during time $\Delta\tau$. Let us assume a constant fixed-node walker distribution equal to $\Psi_g\Psi$. If we suddenly remove the nodal surface, there will be an initial increase of walker population resulting from walkers that would have crossed the nodal surface where it is present. Therefore the resulting population increase per nodal area to first order in time is dF/dA. Let us make $\Delta\tau$ small enough that Ψ does not change at a distance ϵ from the node; then, the population increase per nodal area from the

released nodes is

$$\frac{dF(A)}{dA} = \Psi_g(\mathbf{R}_A)\int_{-\epsilon}^{\epsilon}\int_0^{\Delta\tau} dz d\tau \frac{d\Psi(\mathbf{A},z)}{d\tau}, \tag{20}$$

where $\mathbf{R}_A = (\mathbf{A}, 0)$. We can rewrite the above as

$$\frac{dF(A)}{dA} = \Psi_g(\mathbf{R}_A)\int_{-\epsilon}^{\epsilon}\int_0^{\Delta\tau} dz d\tau (-H)\Psi(\mathbf{A},z). \tag{21}$$

We take the $\epsilon \to 0$ limit of Eq. (21), which leaves only a kinetic contribution from the discontinuity of $\nabla\Psi$:

$$\frac{dF(A)}{dA} = \Psi_g(\mathbf{R}_A)\int_{-\epsilon}^{\epsilon}\int_0^{\Delta\tau} dz d\tau \tfrac{1}{2}\nabla^2\Psi(\mathbf{A},z). \tag{22}$$

Then, using

$$\begin{aligned}\nabla^2\Psi(\mathbf{A},\epsilon) &= \partial_z^2\Psi(\mathbf{A},\epsilon),\\ |\nabla\Psi(\mathbf{A},\epsilon)| &= \partial_z\Psi(\mathbf{A},\epsilon),\\ |\nabla\Psi(\mathbf{A},-\epsilon)| &= 0,\end{aligned} \tag{23}$$

we write

$$\frac{dF(A)}{dA} = \tfrac{1}{2}\Delta\tau\,\Psi_g(\mathbf{R}_A)|\nabla\Psi(\mathbf{R}_A)|. \tag{24}$$

We can now use Eq. (24) to sample the $|\nabla\Psi(\mathbf{R}_A)|$ term of Eq. (16), which yields the equation

$$\frac{\partial E}{\partial\theta_i} \approx -\frac{\sum_j w_j \Psi_g^{-1}(R_j)\,\partial_{\theta_i}\Psi_t(R_j)}{\Delta\tau\,\langle\Psi_t|\Psi\rangle}, \tag{25}$$

where R_j and w_j are, respectively, the positions and weights of walkers that crossed the nodes during the period $\Delta\tau$. To sample the denominator, we simply use

$$\Delta\tau\,\langle\Psi_t|\Psi\rangle = \int_0^{\Delta\tau} d\tau \sum_k \frac{\Psi_t(R_k,\tau)}{\Psi_g(R_k,\tau)} w_k(\tau), \tag{26}$$

with R_k and w_k the positions and weights of all walkers as a function of time τ.

Method B therefore calculates the energy gradient with

$$\frac{\partial E}{\partial\theta_i} \approx -\frac{\sum_j w_j \Psi_g^{-1}(R_j)\partial_{\theta_i}\Psi_t(R_j)}{\int_0^{\Delta\tau} d\tau \sum_k w_k \Psi_g^{-1}(R_k)\Psi_t(R_k)}. \tag{27}$$

E. Method C: The exact energy gradient

Method C does not use the approximation of Eq. (14). It samples one factor of $|\nabla\Psi|$ in Eq. (13) with the same nodal walker distribution used by method B, and samples the other factor of $|\nabla\Psi|$ using forward walking. Just as with method B, method C requires the evaluation of Ψ_t and its parameter derivative.

We start by writing the Ψ projector as a time evolution operator

$$\frac{|\Psi\rangle\langle\Psi|}{\langle\Psi|\Psi\rangle} = \lim_{\tau\to\infty}\exp[\tau(E\text{-}H)] \tag{28}$$

and contract the left side of Eq. (28) with $\langle\Psi_t|$ and the right with $|J\rangle_{\mathbf{R}}$ where

$$|J\rangle_{\mathbf{R}} \equiv \frac{1}{\Psi_t(\mathbf{R})}|\mathbf{R}\rangle, \tag{29}$$

resulting in

$$\frac{\Psi(\mathbf{R})\,\langle\Psi_t|\Psi\rangle}{\Psi_t(\mathbf{R})\,\langle\Psi|\Psi\rangle} = \lim_{\tau\to\infty}\langle\Psi_t|\exp[\tau(E\text{-}H)]|J\rangle_{\mathbf{R}}. \tag{30}$$

We recognize that

$$\langle\Psi_t|\exp[\tau(E\text{-}H)]|J\rangle_{\mathbf{R}} = \int d\mathbf{R}'\widetilde{G}(\mathbf{R}',\mathbf{R},\tau), \tag{31}$$

where $\widetilde{G}$, a Green's function, is the evolution operator of $\Psi_t\Psi$. Because $\Psi(\mathbf{R})/\Psi_t(\mathbf{R})$ approaches $|\nabla\Psi(\mathbf{R}_A)|/|\nabla\Psi_t(\mathbf{R}_A)|$ as $\mathbf{R}$ approaches the nodal surface (since $\nabla = \partial_z$ at $z \to 0^+$), at the nodal surface we have

$$|\nabla\Psi(\mathbf{R}_A)| = |\nabla\Psi_t(\mathbf{R}_A)|\frac{\langle\Psi|\Psi\rangle}{\langle\Psi_t|\Psi\rangle}\lim_{\tau\to\infty}\int d\mathbf{R}'\widetilde{G}(\mathbf{R}',\mathbf{R}_A,\tau). \tag{32}$$

We now substitute one $|\nabla\Psi|$ term of Eq. (13) with Eq. (32) and again use $\frac{\partial\eta}{\partial\theta_i} = -\frac{1}{|\nabla\Psi_t|}\frac{\partial\Psi_t}{\partial\theta_i}$ to get

$$\begin{aligned}\frac{\partial E}{\partial\theta_i} = -\frac{1}{2\,\langle\Psi_t|\Psi\rangle}&\int_{\text{node}} d\mathbf{A}\,\frac{\partial\Psi_t}{\partial\theta_i}\,|\nabla\Psi(\mathbf{R}_A)|\\ &\times\lim_{\tau\to\infty}\int d\mathbf{R}'\,\widetilde{G}(\mathbf{R}',\mathbf{R}_A,\tau).\end{aligned} \tag{33}$$

Equation (33) is identical to Eq. (16) except the added factor $\lim_{\tau\to\infty}\int d\mathbf{R}'\widetilde{G}(\mathbf{R}',\mathbf{R}_A,\tau)$. We sample this factor with the value ξ we define as the final weight of a walker that had an initial weight of 1 and an initial position of $\mathbf{R}_A$, and was then propagated for time τ with Ψ_t as the guiding function, that is,

$$\big\langle\xi(\mathbf{R}_A,\tau)\big\rangle = \int d\mathbf{R}'\widetilde{G}(\mathbf{R}',\mathbf{R}_A,\tau). \tag{34}$$

Instead of taking the $\tau\to\infty$ limit, we cut the propagation time short at τ_c. The value we choose should be long enough for $\langle\xi(\mathbf{R}_A,\tau)\rangle$ to become roughly constant.

Placing a walker guided by Ψ_t at the nodes can be problematic since the local energy and raw drift velocity $\mathbf{V} = \nabla\ln\Psi_t$ are divergent at the nodes when Ψ_t is used as the guiding function. When calculating ξ, we suppress the effects of both divergences in the standard way, where we modify the drift velocity and weight increase by multiplying them with $\frac{-1+\sqrt{1+2V^2\delta\tau}}{V^2\delta\tau}$, where $\delta\tau$ is the time step. To ensure walkers move towards the positive side of the node, we also multiply the drift velocity by $\text{sign}(\Psi_t)$.

Method C calculates Eq. (33) with node crossing walkers following the same logic we used for method B and Eq. (16), but with the extra factor ξ. We multiply Eq. (27) with $\xi(R_j,\tau_c)$ to obtain our equation for method C:

$$\frac{\partial E}{\partial\theta_i} \approx -\frac{\sum_j w_j\xi(R_j,\tau_c)\Psi_g^{-1}(R_j)\partial_{\theta_i}\Psi_t(R_j)}{\int_0^{\Delta\tau} d\tau\sum_k w_k\Psi_g^{-1}(R_k)\Psi_t(R_k)}. \tag{35}$$

III. IMPLEMENTATION

To test our method, we developed a DMC code with walker propagation that for the most part follows the prescription of Umrigar *et al.* [23]. In addition, we introduced our PGD iteration method in the code. The main steps are outlined next.

We will apply our technique on three systems: atomic Be, Li_2, and atomic Ne.

A. Trial and guiding function

The Ψ_t used for all of our calculations were of the well-known Slater-Jastrow form

$$\Psi_t = J\Psi_S, \qquad \Psi_S = D^{\uparrow}D^{\downarrow}, \tag{36}$$

where $D^{\uparrow}$ and $D^{\downarrow}$ are spin-up and spin-down Slater determinants of single-particle orbitals $\phi_a(r)$ described in the next paragraph. J is a Jastrow function:

$$J = \prod_{i<j} f_{ij}, \qquad f_{ij} = e^{-\frac{a_{ij}r_{ij}}{1+br_{ij}}} \tag{37}$$

where r_{ij} is the electron-electron distance and a_{ij} is equal to $1/4$ (or $1/2$) when both i and j correspond to electrons of parallel (or antiparallel) spin projections [23].

The single-particle orbitals ϕ of the Slater determinants are made of a basis of Slater functions, taking the form

$$\phi_a(r) = \sum_{b=1}^{N_{\text{basis}}} C_{ab} r_b^{n_b-1} e^{-\xi_b r_b} Y_{l_b m_b}(\hat{r}_b), \tag{38}$$

where $r_b = |\mathbf{r} - \mathbf{R_b}|$, and n_b, l_b, and m_b are quantum numbers characterizing the b basis state which is centered at the atomic nuclear position $\mathbf{R}_b$. The parameters we optimize are C_{ab} and ξ_b, with ξ_b being shared by all orbitals.

Since J is nodeless, and, thus, does not affect the energy, our method cannot improve it. We therefore do not include it during PGD since we have no *a priori* knowledge of its parameters. Although there is a reason to expect that a quality J will improve the methods A and B, since they rely on a $\Psi \to \Psi_t$ approximation, we do not notice such an improvement. Method C on the other hand makes no such approximation, so in principle, it does not require an optimized J. We only include J when evaluating the energy after optimization.

Our walker distributions for methods B and C use an electron-nuclear Jastrow function as a guiding function Ψ_g. It has the form

$$\Psi_g = \prod_{i,k} \exp\left(-\frac{Z_k|\mathbf{R_k} - \mathbf{r}_i|}{1 + |\mathbf{R_k} - \mathbf{r}_i|}\right), \tag{39}$$

with $\mathbf{R_k}$ the nuclear coordinate, Z_k the nuclear charge, and $\mathbf{r}_i$ the electron coordinate. We choose this Ψ_g instead of $\Psi_g = 1$ to reduce the statistical, time-step, and population growth errors. This Ψ_g also results in many more walkers crossing the node per time, and the electrons are far less likely to ionize. Ionization can still be a problem with poor nodes; when this is the case, we surround the system with a potential barrier.

B. Projection on the cusp-condition satisfying parameter space

The electron-nuclear cusp condition [24,25] is given by

$$S_k \equiv -\frac{1}{2}\frac{d\ln\overline{\Psi_t^2}}{dr}\Big|_{r=0}, \qquad S_k = Z_k, \tag{40}$$

where r_k is the radial coordinate from nucleus k, Z_k is the nuclear charge, and $\overline{\Psi_t^2}$ is defined as the angular average of Ψ_t^2 about $r_k = 0$. It is a necessary requirement to avoid a large divergence of local energy near the nucleus, which gives rise to a large increase of all errors besides the fixed-node error. Since the cusps S_k of our Ψ_t depend on Ψ_S and its varied parameters θ_i, we project θ_i back to the manifold of the cusp condition after each iteration of gradient descent.

The cusp condition of Ψ_S is satisfied when the cusps S_{ka} of all single-particle orbitals ϕ_a satisfy $S_{ka} = Z_k$. We assume the shift in parameters is small enough (due to small a_i) that a linear approximation of the cusp $\widetilde{S}_{ka}$ can be made at the preprojected parameters θ_{i0}:

$$\widetilde{S}_{ka}(\theta_i) \equiv S_{ka}(\theta_{i0}) + \sum_i (\theta_i - \theta_{i0})\frac{\partial S_{ka}(\theta_{i0})}{\partial \theta_i}. \tag{41}$$

θ_i is then shifted to the closest point satisfying

$$\widetilde{S}_{ka}(\theta_i) = \widetilde{S}_{ka}(\theta_{i0}) + c[Z_k - \widetilde{S}_{ka}(\theta_{i0})], \tag{42}$$

where c is added for stability and is between zero and one (we used 0.5). We repeat this until $S_{ka} - Z_k$ is within a threshold.

C. Details of the PGD iterations

After each PGD iteration, we update the energy and effective time step [17] with data of that iteration. We use the mixed estimate

$$E = \frac{\langle\Psi_t|H|\Psi\rangle}{\langle\Psi_t|\Psi\rangle},$$

for method A, and for methods B and C we use the growth estimate

$$E = \Delta\tau^{-1}\ln\frac{\Sigma(0)}{\Sigma(\Delta\tau)},$$

where $\Sigma(\tau)$ is the sum of the weights of all the walkers at imaginary time τ, ignoring weight normalization.

To calculate the gradient, we propagate the walker distribution f until the number of samples (walker positions and weights) reaches a threshold. Samples for method A are taken every time step, and samples for methods B and C are taken from all node-crossing walkers. After an iteration of projected gradient descent, we propagate f for an extra time (around 0.1 Ha^{-1}) to adjust it to the new Ψ_t before taking samples for the next gradient. This is necessary to do for method A because after a shift of the node, f does not go to zero at the nodes fast enough for the expectation of the sum of Eq. (19) to be finite, due to the Ψ_t^{-1} factor. For method A, requiring f to go to zero sufficiently fast also requires us to use an accept-reject step.

The gradient has stochastic error that prevents PGD of the parameters θ_i from fully settling to an accurate minimum, where the signal to noise ratio of the gradient diverges. Instead, θ_i continues to fluctuate around the minimum after some PGD iterations. We argue in Appendix A that to first order in the a_i of Eq. (1), the variance of fluctuations is proportional to both the variance of the gradient error and a_i. Thus we can suppress the fluctuations through smaller a_i and/or by reducing the statistical error of the energy derivative with more samples.

An important choice to make is the value of a_i. The optimal value depends on many factors, such as the parameters θ_i, the form of Ψ_t, the number of PGD iterations, and the

stochastic gradient error. Generally, decreasing a_i requires more PGD iterations to reach a minimum, and may exacerbate the problem of getting stuck in local minima, while increasing a_i increases fluctuations from the gradient error and may cause overshooting of θ_i from its optimum value.

The most basic gradient-descent scheme uses a common value of a_i for all parameters and iterations. We experiment with this and with the two adaptive gradient-descent algorithms Adam [26] and RMSprop [27]. The adaptive algorithms make a_i inversely proportional to a trailing root mean square of past gradient components, and are described more in Appendix B. The a_i of all our algorithms have a common factor which we label the learning rate α, which we determine empirically. We sometimes decrease this value with PGD iterations to suppress fluctuations of θ_i. In Appendix C we examine the effective range of α for different gradient-descent algorithms and systems.

IV. RESULTS

We tested our methods for three systems—atomic Be, Li_2, and atomic Ne—and we compared our results to those obtained by the DMC calculation of Umrigar *et al.* [23] where the fermion nodes were optimized using VMC.

A. Derivative of the energy calculation

We present tests of the energy gradient calculation methods A, B, and C in this subsection. They examine one parameter of a basis we added to the simple Be trial function that is fully described in Ref. [23]. The basis has quantum numbers $n = 2$, $l = 1$, and $m = 0$, and the parameter is the non-normalized coefficient of the basis. We varied this parameter and obtained its partial first and second energy derivatives by means of a fixed-node DMC calculation, which is shown in the upper panel of Fig. 2. As can be noticed, the energy obtained from the fixed-node DMC is well approximated with a parabola. We wish to compare the derivative of the parabola with the energy derivatives obtained by our three methods which are shown in the center panel of Fig. 2 and are well approximated with lines intersecting the origin.

The energy derivative obtained by method C agrees with that from the parabola to within 1%, while the energy derivative obtained using methods A and B is proportional to that from the parabola but off by a factor of 2, which we attribute to the approximation of Eq. (14). We expect that when each of the components of the energy derivative has the correct sign it results in shifting the parameters towards the direction where the energy is lower. In addition, many gradient-descent algorithms divide each gradient component by a trailing root mean square of past gradient components; thus, the exact value of the slope is not needed by such algorithms. As a result methods A and B may still be useful.

The bottom panel of Fig. 2 shows how the energy derivative obtained with method C depends on the cutoff time τ_c. Our choice of τ_c when optimizing with method C is based on this dependence.

We compare the efficiency of the three methods in Fig. 3 by plotting the error of the energy derivative versus wall time, which is proportional to the sample size. In this figure we show the best-fit power laws which, as expected, are approximately proportional to the inverse square root of sample size. Methods B and C seem to require similar wall time to reach a given error level when using $\tau_c = 0.25$ Ha^{-1}. Method A on the other hand is considerably more expensive, though a fair comparison is difficult partly because method A may not require the same time step, and the error depends both on the associate parameter and on Ψ_t.

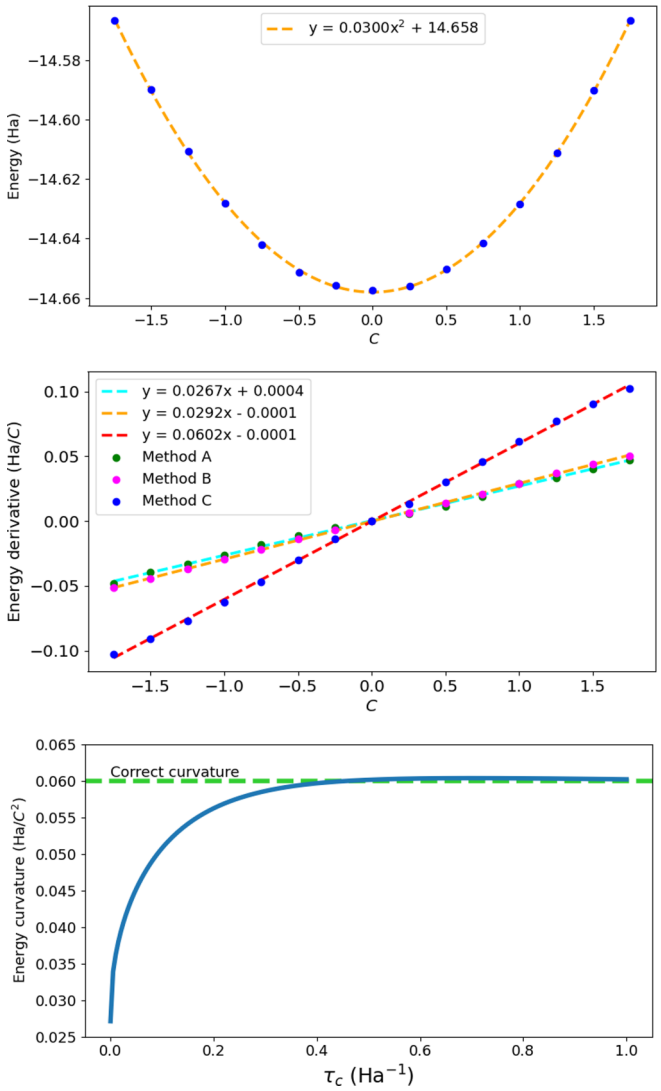

FIG. 2. We compare the DMC energy (top) of a Be trial function for a varied parameter with the energy derivative with respect to said parameter (center) that is calculated with methods A–C. Data points are shown as dots with an associated best fit line or parabola. The parabola curvature is within 1% of the slope of method C, which uses $\tau_c = 1(Ha^{-1})$. We examine how the slope of method C (bottom) depends on τ_c.

B. Projected gradient-descent evolution

In order to demonstrate the utility of the method, we start from random values for the parameters C and ξ of all-electron single-determinant wave functions of Be, Li_2, and Ne, and we perform PGD iterations with our cusp condition projection.

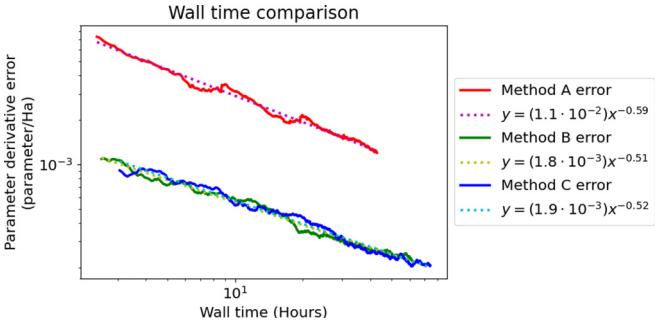

FIG. 3. We compare the efficiency of methods A, B, and C by plotting their energy derivative error for the parameter of Fig. 2 as wall time (sample size) is increased, using a common time step. Best-fit (dotted lines) power laws are roughly proportional to the inverse square root of sample size. Methods B and C require similar wall time to achieve the same error, while method A requires on the order of 50 times more.

We present a history of the energy of their nodes in Fig. 4. One can see the DMC energies rapidly decrease with iteration to the energy of the VMC optimized nodal surface obtained by Umrigar *et al.* [23] (red dotted lines). Ne continues to fluctuate after initial decrease, which we attribute to statistical error of the energy gradient and an excessively large decent rate [a_i term of Eq. (1)]. Adam was used for all examples, and we decreased the descent rate [α parameter of Eq. (B3)] with iteration number. Method C was used for all except Be, where method A was used, demonstrating the utility of the approximation of Eq. (16).

We chose these three systems because a full description of their VMC optimized wave functions was present in the literature [23], allowing us to directly compare our PGD optimization using wave functions of the same form. There are DMC calculations for larger atoms and many atom systems; however, many of the atomic electrons (the inner) are absent in most of these calculations because they are using pseudopotentials and, therefore, do have to implement the cusp condition. Nevertheless, we have applied PGD to a somewhat larger system of an all electron single determinant of F_2, shown in Fig. 5. Unfortunately we do not have a wave function of the same form optimized by other means for comparison. However, a DMC calculation of F_2 was done by Giner *et al.* [28], who used a configuration-interaction optimized wave function consisting of 10^5 determinants. They achieved an energy of $-199.2977(1)$ Ha, while our lowest PGD iteration was $-196.7(3)$ Ha. This discrepancy can partly be attributed to the fact that our wave function had just one determinant (a current limitation of our code), and a single-particle basis of just 20 Slater-type orbitals. The decrease in error of the F_2 energy with iteration number suggests that PGD also indirectly reduces local energy variance.

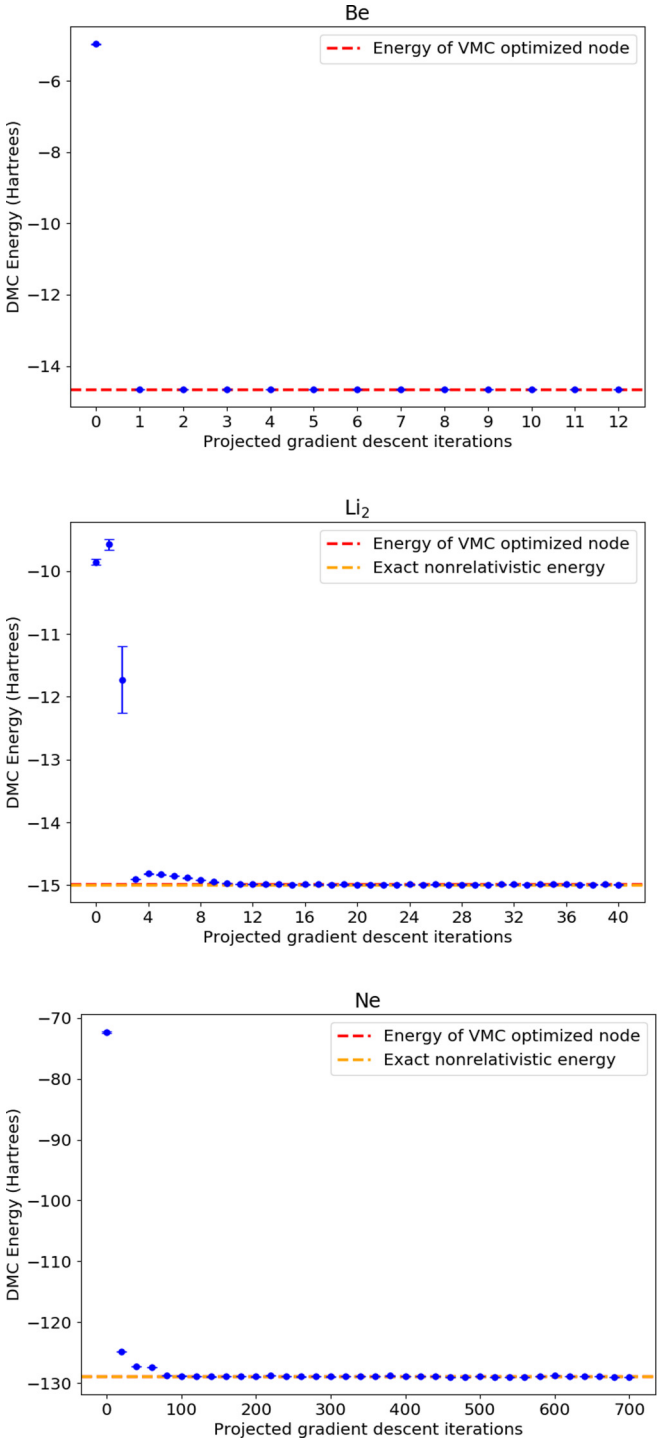

FIG. 4. Projected gradient descent of initially random parameters. See text for detailed explanation.

C. Energy upper-bound estimation

The set $\{E_1, E_2, \ldots, E_n, \ldots\}$ formed by the values of the energy E_i at the ith step of the PGD method shown in Fig. 4 is to be taken as a set of energy upper bounds. As already discussed the variations from step to step are due to the fact that the sign of the energy derivative with respect to the PGD parameter α is sampled and, thus, its choice has some random element. For every such choice, the DMC evolution can be carried out with a controlled level of error because the nodal surface is fixed. Therefore, the best energy upper bound $\mathcal{E}$ corresponds to the lowest value of the energy in the set, i.e., $\mathcal{E} = \min\{E_1, E_2, \ldots, E_n, \ldots\}$. The last iteration energy obtained for the three systems studied in this paper

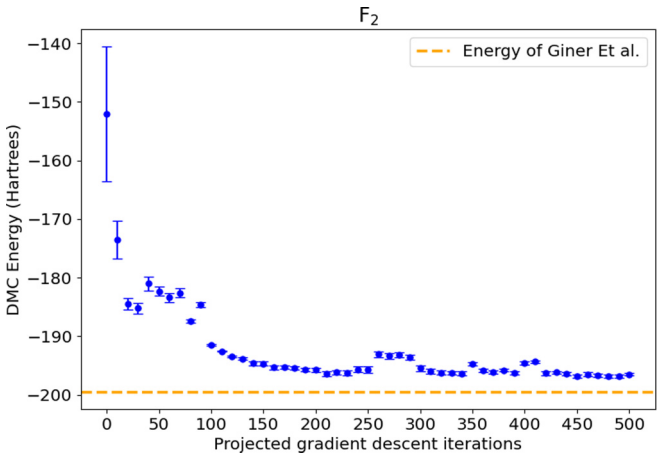

FIG. 5. See text for detailed explanation.

are compared with the best DMC energy obtained with nodes determined by a VMC optimization [23] in Table I.

The present calculation suggests that the nodes obtained with VMC optimization are not far from the optimum at least within the error of the present calculation. Therefore, the small discrepancy with the exact nonrelativistic values could be attributed to the fact that the form of the one-body factor is limited with the space of a single Slater determinant and the lack of spin-spin correlations because the one-body factor is taken to be a product of the form $D^{\uparrow}D^{\downarrow}$.

V. DISCUSSION

We are not yet comfortable claiming that one of the methods A, B, or C is ideal, and the best choice may depend on circumstance. For example, although method A was least efficient in our test, it may not require as small of a time step, and we think it would be the simplest to implement into existing code due to its use of a standard walker distribution. Although the efficiency of method C was comparable to that of method B in our test, it requires additional wall time to calculate ξ, which also introduces the additional parameter τ_c for which we do not yet have a good procedure for determining.

Nevertheless, method C performed the best in our tests and also has the advantage of not relying on the quality of Ψ_t. We recommend pairing method C with Adam. In our tests we were able to optimize the location of the nodes to roughly the same energy as with the VMC optimized nodes using any of the methods and without using Jastrow factors in our guiding function.

TABLE I. DMC energy comparison between the last PGD iteration nodes with VMC optimized nodes. Exact nonrelativistic values are presented.

System	Last PGD iteration	VMC optimized nodes	Exact energy
Be	−14.6567(5)	−14.6571(1)	−14.66736 [29]
Li_2	−14.989(1)	−14.9898(1)	−14.9954 [29]
Ne	−128.92(1)	−128.919(3)	−128.939 [30]

Without the requirement of an optimized Jastrow function, our method is independent of prerequisite optimization by VMC. Nevertheless, simultaneously optimizing the Jastrow and Slater parameters may be a fruitful avenue, as a quality Jastrow function reduces many sources of error (except for the fixed-node error) and should improve the approximation of methods A and B. This could be done with the standard VMC method, or possibly with DMC and gradient descent, which require a gradient different than one of the DMC energy, perhaps one of the local energy variance.

An important advantage of PGD optimization is that the use of DMC propagation produces correct correlations in the fixed-node wave function, whereas VMC optimizes the nodal surface within a limited Jastrow form, which captures only pairwise correlations. Thus, the exact path to optimization due to the interplay between the adjustment of these correlations as the position of the nodal surface changes and vice versa was not fully accounted for in previous DMC studies.

We would like to give an example of a system which demonstrates the significance of the present paper. Variational calculations of liquid ^{3}He, where just a Jastrow-Slater wave function is used, yield [31] an unphysical result that the fully polarized liquid is of lower energy at its equilibrium density than the unpolarized liquid at its equilibrium density. Adding three-body correlation factors in the wave function does not change the above conclusion. It is when one includes backflow correlations, which are state dependent and modify the nodal surface of the variational wave function, that one finds [31] that the unpolarized state of liquid ^{3}He is energetically favorable.

Therefore, let us pretend that we do not know the fact that the naturally occurring liquid ^{3}He is spin unpolarized; we might then be naive and use a Slater-Jastrow variational calculation to determine the optimum polarization. The nodes that would be determined by such a VMC optimization would be those that correspond to a polarized liquid. A subsequent fixed-node DMC calculation with these nodes is not going to change this result, and we would reach an unphysical conclusion. The PGD method described in the present paper, however, should find that the unpolarized state is the ground state.

Another advantage of PGD is that it requires only one gradient calculation to shift all the parameters in the correct direction, whereas if one were to vary the parameters and select values with lower calculated DMC energy, the number of required energy calculations would scale with the number of parameters used. We should point out that in our PGD evolution examples, far more CPU time was spent calculating the energy of a single nodal surface than the entire PGD process.

PGD optimization opens up a possible way to optimize atomic positions to minimize DMC energy. If one were to combine the nuclei and electrons into one walker type, then DMC imaginary-time propagation combined with the PGD method should naturally relax the atomic positions. Unfortunately we could not attempt this because the electron-nuclear cusp of the particular type of Ψ_t we used depends on both the atomic positions and the variational parameters, requiring the atoms be fixed. Using a Ψ_t that does not have this dependence would be an interesting avenue to explore.

VI. SUMMARY

We presented a general and self-reliant method using DMC and projected gradient descent that optimizes the nodal surface of a trial function to minimize the fixed-node ground state, i.e., the DMC energy. We derived three methods for calculating the DMC energy gradient from walker distribution (methods A, B, and C). Methods B and C required comparable CPU time in our test, while method A was more expensive.

We combined the three methods with several gradient-descent algorithms and a projection operation that maintains the cusp condition. We benchmarked this projected gradient-descent method to trial functions with randomized parameters of Be, Li_2, and Ne. Their energies were lowered to the same level as VMC optimized parameters, without the use of a Jastrow factor.

Our paper is a proof of concept using simple systems, but we see no reason it cannot be applied to larger and periodic systems, or be implemented into existing DMC code for various types of node determining wave functions. However, before attempting these goals, our paper may be further improved with better and more adaptive choices for various parameters, by simultaneously optimizing Jastrow parameters during PGD, by better suppressing PGD fluctuations, and by establishing criteria for convergence. In addition, we see no reason why the method could not be extended to DMC on a discrete lattice or to finite temperature path-integral Monte Carlo.

ACKNOWLEDGMENTS

We would like to thank the Florida State University high performance computing center for the allocation of CPU time to carry out the calculations presented in this paper.

APPENDIX A: PARAMETER FLUCTUATIONS

After a number of PGD iterations, the parameters will continue to fluctuate around a local minimum due to stochastic error of the energy gradient. Here, we investigate the variance of these fluctuations. First let us perform a coordinate shift so that the parameters are zero at the local minima, then let us rotate them so that the Hessian $\frac{\partial^2 E}{\partial\theta_i\partial\theta_j}$ is diagonal. Then, if we expand E to second order in θ_i, one iteration of gradient descent using Eq. (1) will result in new parameters θ_i' given by

$$\theta_i' = \theta_i - a_i(K_i\theta_i + \sigma_i\eta), \tag{A1}$$

where $K_i \equiv \frac{\partial^2 E}{\partial\theta_i^2}$, σ_i is the standard deviation of stochastic error, and η is a random number with $\langle\eta\rangle = 0$ and $\langle\eta^2\rangle = 1$.

Let us take the variance of both sides of Eq. (A1):

$$\langle\theta_i'^2\rangle = \langle\theta_i^2\rangle + a_i^2\left(K_i^2\langle\theta_i^2\rangle + \sigma_i^2\right) - 2a_iK_i\langle\theta_i^2\rangle. \tag{A2}$$

After enough PGD iterations, the variance of fluctuations will become constant, i.e.,

$$\langle\theta_i'^2\rangle = \langle\theta_i^2\rangle. \tag{A3}$$

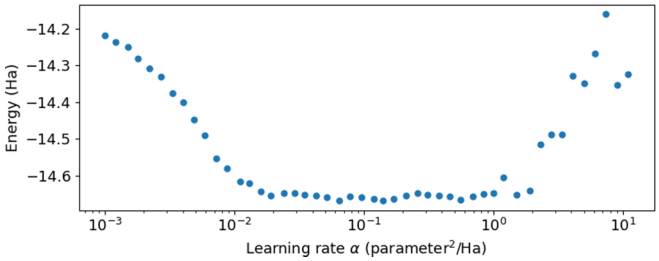

FIG. 6. An example of how the energy, calculated after 100 PGD iterations, depends on different learning rates α. These data were used for the sixth row of Table II.

If this is the case, we can use Eq. (A1) to evaluate the variance of latter PGD iterations, namely,

$$\langle\theta_i^2\rangle = \frac{a_i\sigma_i^2}{2K_i - a_iK_i^2}. \tag{A4}$$

We notice that the variance blows up if any a_i approach $2/K_i$, which is the result of overshooting the minimum by more than double the distance. For $a_i << 2/K_i$ we see that the variance has the property

$$\langle\theta_i^2\rangle \propto a_i\sigma_i^2. \tag{A5}$$

APPENDIX B: GRADIENT-DESCENT ALGORITHMS

We list the gradient-descent algorithms we tested. We use $g_i \equiv \frac{\partial E}{\partial\theta_i}$ as the gradient, and determine α empirically. We sometimes decrease α with iteration number to suppress parameter fluctuations around the minimum.

Basic gradient descent:

$$\theta_i \to \theta_i - \alpha g_i. \tag{B1}$$

RMSprop:

$$\begin{aligned} v_i &\to \beta v_i + (\beta - 1)g_i^2, \\ \theta_i &\to \theta_i - \frac{\alpha g_i}{\sqrt{v_i}}. \end{aligned} \tag{B2}$$

Adam:

$$\begin{aligned} t &\to t + 1, \\ m_i &\to \beta_1 m_i + (\beta_1 - 1)g_i, \\ v_i &\to \beta_2 v_i + (\beta_2 - 1)g_i^2, \\ \theta_i &\to \theta_i - \frac{\alpha m_i g_i}{(1-\beta_1^t)\sqrt{v_i/(1-\beta_2^t)}}. \end{aligned} \tag{B3}$$

APPENDIX C: EFFECTIVE α

In Table II we show the effective ranges of the learning rate α, which is a hyperparameter proportional to a shift in parameters, as shown in Appendix B. We define this effective range as the values that bring the energy of a randomized Ψ_S down to the optimum level, and show an example in Fig. 6. We empirically determine this range for 100 PGD iterations for different combinations of gradient-descent algorithms, atomic systems, and methods A, B, and C. We display the end points of the effective range along with the logarithmic distance between them.

TABLE II. The range of the effective learning rate α for 100 iterations for three gradient-descent algorithms: basic gradient descent (common a_i), RMSprop, and Adam (see Fig. 6 for an example of the tenth row). RMSprop used $\beta = 0.99$. Adam used $\beta_1 = 0.9$, $\beta_2 = 0.99$. Method C used $\tau_c = 0.3\ \mathrm{Ha}^{-1}$.

Algorithm	Gradient method	Ψ_S	Minimum effective α	Maximum effective α	$\log_{10}(\frac{\alpha_{\max}}{\alpha_{\min}})$
Basic gradient descent	B	Be	1.2×10^{-1}	1.3×10^{0}	1.0
	C	Be	2.4×10^{-2}	6.4×10^{-1}	1.4
	A	Li_2	4.8×10^{-3}	2.0×10^{-1}	1.6
	B	Ne	4.5×10^{-2}	1.8×10^{0}	1.6
RMSprop	B	Be	5.5×10^{-2}	1.0×10^{0}	1.3
	C	Be	2.0×10^{-2}	1.0×10^{-1}	1.7
	A	Li_2	1.8×10^{-2}	2.1×10^{3}	5.1
	B	Ne	5.6×10^{-2}	1.5×10^{0}	1.4
Adam	B	Be	3.8×10^{-2}	1.0×10^{0}	1.4
	C	Be	1.1×10^{-2}	1.0×10^{0}	2.0
	A	Li_2	1.6×10^{-1}	7.0×10^{2}	3.6
	B	Ne	2.4×10^{-1}	9.2×10^{0}	1.6

Compared to basic gradient descent, the adaptive algorithms RMSprop and Adam have more consistent end points for methods A, B, and C and for different atomic systems, making the choice of α easier. We attribute the more consistent end points to the division of each gradient component by a trailing root mean square of its prior values. This suppresses the large variations of gradient magnitudes for different atomic systems. It also suppresses large variations of $\xi(R, \tau_c)$ of Eq. (35) used for method C, which has a factor of $\exp(\Delta E\ \tau_c)$, with ΔE the difference between estimated and correct energy.

[1] J. B. Anderson, A random-walk simulation of the schrödinger equation: H+3, J. Chem. Phys. **63**, 1499 (1975).

[2] D. M. Ceperley and B. J. Alder, Ground State of the Electron Gas by a Stochastic Method, Phys. Rev. Lett. **45**, 566 (1980).

[3] D. M. Ceperley and M. H. Kalos, Quantum many-body problems, in *Monte Carlo Methods in Statistical Physics*, edited by K. Binder (Springer-Verlag, Berlin, 1986), pp. 145–194.

[4] E. Manousakis, The spin-1/2 Heisenberg antiferromagnet on a square lattice and its application to the cuprous oxides, Rev. Mod. Phys. **63**, 1 (1991).

[5] J. Carlson, Green's function Monte Carlo study of light nuclei, Phys. Rev. C **36**, 2026 (1987).

[6] S. Gandolfi, D. Lonardoni, A. Lovato, and M. Piarulli, Atomic nuclei from quantum Monte Carlo calculations with chiral eft interactions, Front. Phys. **8**, 117 (2020).

[7] S. C. Pieper, R. B. Wiringa, and J. Carlson, Quantum Monte Carlo calculations of excited states in $a = 6 - -8$ nuclei, Phys. Rev. C **70**, 054325 (2004).

[8] M. Boninsegni and E. Manousakis, Green's-function Monte Carlo study of the t-j model, Phys. Rev. B **46**, 560 (1992).

[9] M. Boninsegni and E. Manousakis, Two-hole d-wave binding in the physical region of the t-j model: A Green's-function Monte Carlo study, Phys. Rev. B **47**, 11897 (1993).

[10] C. S. Hellberg and E. Manousakis, Phase Separation at all Interaction Strengths in the t-j Model, Phys. Rev. Lett. **78**, 4609 (1997).

[11] S. Zhang, J. Carlson, and J. E. Gubernatis, Constrained Path Quantum Monte Carlo Method for Fermion Ground States, Phys. Rev. Lett. **74**, 3652 (1995).

[12] S. Zhang and H. Krakauer, Quantum Monte Carlo Method Using Phase-Free Random Walks with Slater Determinants, Phys. Rev. Lett. **90**, 136401 (2003).

[13] M. Qin, C.-M. Chung, H. Shi, E. Vitali, C. Hubig, U. Schollwöck, S. R. White, and S. Zhang (Simons Collaboration on the Many-Electron Problem), Absence of Superconductivity in the Pure Two-Dimensional Hubbard Model, Phys. Rev. X **10**, 031016 (2020).

[14] R. Jastrow, Many-body problem with strong forces, Phys. Rev. **98**, 1479 (1955).

[15] J. C. Slater, The theory of complex spectra, Phys. Rev. **34**, 1293 (1929).

[16] K. M. Rasch, S. Hu, and L. Mitas, Communication: Fixed-node errors in quantum Monte Carlo: Interplay of electron density and node nonlinearities, J. Chem. Phys. **140**, 041102 (2014).

[17] P. J. Reynolds, D. M. Ceperley, B. J. Alder, and W. A. Lester, Fixed-node quantum Monte Carlo for molecules, J. Chem. Phys. **77**, 5593 (1982).

[18] C. J. Umrigar, K. G. Wilson, and J. W. Wilkins, Optimized Trial Wave Functions for Quantum Monte Carlo Calculations, Phys. Rev. Lett. **60**, 1719 (1988).

[19] C. J. Umrigar and C. Filippi, Energy and Variance Optimization of Many-Body Wave Functions, Phys. Rev. Lett. **94**, 150201 (2005).

[20] F. A. Reboredo, R. Q. Hood, and P. R. C. Kent, Self-healing diffusion quantum Monte Carlo algorithms: Direct reduction of the fermion sign error in electronic structure calculations, Phys. Rev. B **79**, 195117 (2009).

[21] M. E. Foulaadvand and M. Zarenia, Optimization of quantum Monte Carlo wave function: Steepest decent method, Int. J. Mod. Phys. C **21**, 523 (2010).

[22] D. H. Berman, Boundary effects in quantum mechanics, Am. J. Phys. **59**, 937 (1991).

[23] C. J. Umrigar, M. P. Nightingale, and K. J. Runge, A diffusion Monte Carlo algorithm with very small time-step errors, J. Chem. Phys. **99**, 2865 (1993).

[24] T. Kato, On the eigenfunctions of many-particle systems in quantum mechanics, Commun. Pure Appl. Math. **10**, 151 (1957).

[25] E. Steiner, Charge densities in atoms, J. Chem. Phys. **39**, 2365 (1963).

[26] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, arXiv:1412.6980.

[27] G. Hinton, *Neural Networks for Machine Learning, Lecture* (Department of Computer Science, the University of Toronto, Toronto, Canada, 2012).

[28] E. Giner, A. Scemama, and M. Caffarel, Fixed-node diffusion Monte Carlo potential energy curve of the fluorine molecule f2 using selected configuration interaction trial wavefunctions, J. Chem. Phys. **142**, 044115 (2015).

[29] D. Bressanini, G. Morosi, and S. Tarasco, An investigation of nodal structures and the construction of trial wave functions, J. Chem. Phys. **123**, 204109 (2005).

[30] E. R. Davidson, S. A. Hagstrom, S. J. Chakravorty, V. M. Umar, and C. F. Fischer, Ground-state correlation energies for two- to ten-electron atomic ions, Phys. Rev. A **44**, 7071 (1991).

[31] E. Manousakis, S. Fantoni, V. R. Pandharipande, and Q. N. Usmani, Microscopic calculations for normal and polarized liquid ^{3}He, Phys. Rev. B **28**, 3770 (1983).