Fermionic neural network with effective core potential

Xiang Li,[1,*] Cunwei Fan,[2,†] Weiluo Ren,[1] and Ji Chen[3,‡]

[1]*ByteDance Inc., Zhonghang Plaza, No. 43, North 3rd Ring West Road, Haidian District, Beijing, People's Republic of China*

[2]*Department of Physics and Institute for Condensed Matter Theory, University of Illinois 1110 West Green Street, Urbana, Illinois 61801, USA*

[3]*School of Physics, Peking University, Beijing 100871, People's Republic of China*

(Received 6 September 2021; accepted 17 November 2021; published 10 January 2022)

Deep learning techniques have opened a new venue for electronic structure theory in recent years. In contrast to traditional methods, deep neural networks provide much more expressive and flexible wave function Ansätze, resulting in better accuracy and timescale behavior. In order to study larger systems while retaining sufficient accuracy, we integrate a powerful neural-network-based model (FermiNet) with the effective core potential method, which helps to reduce the complexity of the problem by replacing inner core electrons with additional semilocal potential terms in the Hamiltonian. In this work, we calculate the ground-state energy of 3*d* transition metal atoms and their monoxides, which is quite challenging for the original FermiNet work, and the results are consistent with both experimental data and other state-of-the-art computational methods. Our work is an important step for a broader application of deep learning in the electronic structure calculation of molecules and materials.

DOI: 10.1103/PhysRevResearch.4.013021

I. INTRODUCTION

The last decade has witnessed incredibly fast development of artificial intelligence. Deep learning [1] has become widely used and has received great success in computer vision [2,3], natural language processing [4,5], and recommendation systems [6,7], just to name a few areas. In the past few years, deep learning technology has been broadly applied in computational physics and chemistry to tackle key challenges in *ab initio* molecule modeling [8–20], which are crucial for materials design, drug discovery, and other applications.

Deep learning techniques can be roughly divided into two categories. The first category, machining learning force field, aims at improving the efficiency of simulation while retaining the accuracy at a higher level of theory [8–10]. These methods usually require labeled data (such as energy and force of a given structure) and then train the neural network by minimizing the deviation between its prediction and the labeled data. The second category, neural-network-based quantum Monte Carlo (QMC) [11–18], targets more accurate electronic structure and only needs unlabeled sample data in training. A deep neural network can provide a much more expressive and flexible wave function ansatz than traditional forms used in QMC, which leads to better accuracy. The recently developed PauliNet and FermiNet are two promising examples [13–15] that have shown the ability to outperform traditional methods such as coupled cluster with single, double and perturbative triple excitations [CCSD(T)] in certain systems. In spite of their advantages, these methods also have their own drawbacks: the enormous amount of parameters in deep neural networks strictly restrict the simulation speed and system size that can be studied. For example, it would take the FermiNet a month to simulate a system of about 30 electrons using 16 V100 GPUs in its TensorFlow version [13].

In order to extend neural network electronic structure calculations to larger systems, the computation complexity has to be reduced and one helpful approach is the so-called effective core potential (ECP) method (also known as pseudopotential). See Ref. [21] for a review. Electrons in each system can be divided into core electrons and valence electrons. Core electrons, filling inner shells of the system, are tightly bound around the atom cores, and it is mainly the valence electrons in the outer shell that determine the property of the system. The ECP method simply removes the core electrons from computation and introduces semilocal potential terms to effectively simulate their influence on valence electrons, and in this way the number of electrons in the calculation is reduced and the whole computation process is accelerated. The ECP method is widely employed in traditional electronic structure calculations, such as density functional theory (DFT), post Hartree-Fock, and QMC. In particular, the development of ECPs for QMC is still a hot subject of research. Nevertheless, there are already a number of ECPs designed or used in QMC calculations, such as Burkatzki-Filippi-Dolg (BFD) ECP [22], Stuttgart (STU) ECP [23], Trail-Needs (TN) ECP [24], and correlation consistent (cc) ECP [25,26]. These ECPs are examined carefully in traditional QMC simulations, but their numerical stability and performances have not yet been

*lixiang.62770689@bytedance.com
†cfan11@illinois.edu
‡ji.chen@pku.edu.cn

examined in deep neural networks, which has a much more complex structure to express wave functions.

In this work, we investigate the implementation of the ECP method into a deep neural network, namely, FermiNet. The main aim is to improve the efficiency of deep neural network modeling, so as to increase the size of the system that we can handle with FermiNet. Based on our method, we calculate the ground-state energy of $3d$ transition metal atoms and their monoxides. The results show satisfactory consistency with both experimental data and other accurate *ab initio* methods. We also discuss some details of how to improve the training efficiency of the ECP-based FermiNet.

The remainder of this paper is organized as follows. In Sec. II, we introduce briefly the theoretical framework of the neural network, the ECP method, the workflow, and the calculation details. In Sec. III, we present our calculations and results. In Sec. IV, we discuss our calculation and training in more detail. In Sec. V, we give a summary and outlook.

II. METHOD

A. Theoretical framework

Solving the Schrödinger equation is always the main task of electronic structure calculation. Under the Born-Oppenheimer approximation [27], atomic motion is frozen and the equation for the electron wave function ψ can be formulated as

$$\hat{H}\psi(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N) = E\psi(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N),$$
$$\hat{H} = \sum_i -\frac{1}{2}\Delta_i + \frac{1}{2}\sum_{i\neq j}\frac{1}{|\mathbf{r}_i - \mathbf{r}_j|} - \sum_{i,I}\frac{Z_I}{|\mathbf{r}_i - \mathbf{R}_I|} + \frac{1}{2}\sum_{I\neq J}\frac{Z_I Z_J}{|\mathbf{R}_I - \mathbf{R}_J|}, \tag{1}$$

where $\mathbf{x}_i = (\mathbf{r}_i, \sigma_i)$ denotes the spatial position and spin of the ith electron. $\mathbf{R}_I$ and Z_I are the spatial position and the charge of the Ith nucleus, which are treated as external parameters for a given molecule structure. Moreover, since electrons obey Fermi-Dirac statistics, the wave function ψ needs to be antisymmetric with respect to the permutation of $(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)$. The antisymmetric requirement together with its high-dimensional nature make the Schrödinger equation notoriously hard to solve. In order to obtain highly accurate results, the time complexity of the state-of-the-art methods, such as CCSD, scales as large as N^6 or more, where N is the total number of electrons in the system.

Recently, deep neural network models, such as FermiNet and PauliNet [13–15], are proposed and shed new light on the electronic structure problem. Based on the variational principle, these deep learning methods approach the ground-state wave function via minimizing the energy expectation value E_θ, which reads

$$E_\theta = \frac{\int d^3\mathbf{r}\ \psi_\theta^*(\mathbf{r}) H \psi_\theta(\mathbf{r})}{\int d^3\mathbf{r}\ \psi_\theta^*(\mathbf{r})\psi_\theta(\mathbf{r})}. \tag{2}$$

Here ψ_θ is simply the wave function output by the neural network and θ denotes all the parameters within.

Traditional Variational Monte Carlo (VMC) approaches usually start from the Hartree-Fock wave function, which reads

$$\psi_{\mathrm{HF}}(\mathbf{x}) = \begin{vmatrix} \phi_1(\mathbf{x}_1) & \cdots & \phi_1(\mathbf{x}_N) \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \phi_N(\mathbf{x}_1) & \cdots & \phi_N(\mathbf{x}_N) \end{vmatrix}, \tag{3}$$

where $\{\phi_i\}$ denotes the molecular orbitals for the electrons. In deep neural networks [12–15], the one-electron orbital $\phi_i(\mathbf{x}_j)$ is extended to $\phi_i(\mathbf{x}_j; \mathbf{x}_{\neq j})$, where $\mathbf{x}_{\neq j}$ denotes all the electron coordinates except $\mathbf{x}_j$, and it can be seen as a generalization of widely used backflow transformation [28]. In order to ensure the antisymmetry of ψ, $\phi_i(\mathbf{x}_j; \mathbf{x}_{\neq j})$ is required to be permutation invariant with respect to permutation of $\mathbf{x}_{\neq j}$. With this generalization employed, the simulation accuracy is highly improved and the time complexity is retained as the traditional VMC approach, which scales as N^4.

However, existing deep neural networks, namely, FermiNet and PauliNet, suffer from a large prefactor in its asymptotic N^4 time scaling. The enormous amount of parameters and linear operations within neural networks reduces their computation efficiency, and the largest system size they can study is limited up to dozens of electrons. In order to enlarge the system size we can study, it is natural to employ the ECP method, which has already been widely used in the quantum chemistry community. Within the ECP framework, core electrons decouple with the valence electron, and additional semilocal potential terms are added to the Hamiltonian to mimic core electron effects, which read as

$$\hat{V}_{\mathrm{ECP}} = \sum_{v=1}^{n_v} V_{\mathrm{loc}}(r_v) + \sum_{v=1}^{n_v}\sum_{l=0}^{l_{\max}} V_l(r_v)\sum_{m=-l}^{m=l} |lm\rangle\langle lm|, \tag{4}$$

where n_v denotes the number of valence electrons and $|lm\rangle$ represents the spherical harmonics. Moreover, the potential terms V_{loc} and V_l are simply functions of r_v, which represents the radial distance between the valence electron and the nucleus. These potential terms are usually expanded in Gaussian basis sets, which read as

$$V_l(r) = r^{-2}\sum_k A_{lk} r^{n_{lk}} e^{-B_{lk} r^2}, \tag{5}$$

where A_{lk}, B_{lk}, and n_{lk} are the expansion parameters, and l and k denote the angular quantum number and the expansion index, respectively.

Note that effective core potentials usually diverge near the nucleus due to the r^{-2} term in Eq. (5), making the QMC simulation unstable. There are several kinds of ECPs such as ccECP, STU ECP, BFD ECP, and TN ECP overcoming this problem via exact cancellation of diverging terms in Eq. (4), which are more suitable for quantum Monte Carlo simulations [22–26]. In this work, we adopt the recently proposed ccECP [25,26]. Other ECPs can be implemented in a similar way with FermiNet.

B. Workflow

The workflow of our work is an extension of FermiNet, with modifications related to the integrated ECP method.

The main workflow can be divided into three phases: pretrain, train, and inference. In the pretrain phase, the neural network is trained to match the Hartree-Fock wave function, obtained via the PYSCF package [29]. Then the neural network is trained to minimize the expected energy value. Note that with the ECP employed, the gradient formula for energy optimization is slightly modified as follows,

$$\begin{aligned} \text{Grad} &= \mathbb{E}_{\psi^2(\mathbf{r})}[(E(\mathbf{r}) - \mathbb{E}_{\psi^2(\mathbf{r})}[E(\mathbf{r})])\nabla_\theta \log|\psi|], \\ E(\mathbf{r}) &= \psi^{-1}(\mathbf{r})\hat{H}\psi(\mathbf{r}) + E_{\text{nl}}(\mathbf{r}), \end{aligned} \tag{6}$$

where $\mathbb{E}_{\psi^2(\mathbf{r})}[\cdot]$ represents the expectation value according to the distribution $\psi^2(\mathbf{x})$ and it is evaluated via the traditional Markov chain Monte Carlo (MCMC) approach. Moreover, E_{nl} denotes the semilocal energy contribution from the effective core potential and its specific form reads as follows [30]:

$$\sum_{vlm} V_l(r_v) Y_{lm}(\mathbf{\Omega}_v) \int d\mathbf{\Omega}'_v Y^*_{lm}(\mathbf{\Omega}'_v) \frac{\psi(\mathbf{r}_1, \ldots, \mathbf{r}'_v, \ldots, \mathbf{r}_N)}{\psi(\mathbf{r}_1, \ldots, \mathbf{r}_v, \ldots, \mathbf{r}_N)}, \tag{7}$$

where Y_{lm} denotes the spherical harmonics. This integral is over the solid angle $\mathbf{\Omega}'_v$ of the valence electron vector $\mathbf{r}'_v$ with respect to the nucleus. Although, in the mean-field approximation, the integration has a closed-form result, an analytic result does not exist for the wave function ψ produced by FermiNet and thus numerical integration methods are used. After a long enough training process until the energy converges, the final energy result can be obtained from a separate inference phase, in which energy is estimated via the pure MCMC approach without training.

C. ECP implementation details

Before carrying out the numerical integration concretely, we note that the integrand in Eq. (7) can be simplified further. The polar axis of the integrand can be set parallel to $\mathbf{r}_v$ and the outer $Y_{lm}(\mathbf{\Omega}_v)$ terms are reduced to constants [31], then terms in Eq. (7) read as follows:

$$\begin{aligned} &\sum_{m} V_l(r_v) Y_{lm}(\mathbf{\Omega}_v) \int d\mathbf{\Omega}'_v Y^*_{lm}(\mathbf{\Omega}'_v) \frac{\psi(\mathbf{r}_1, \cdots, \mathbf{r}'_v, \cdots, \mathbf{r}_N)}{\psi(\mathbf{r}_1, \cdots, \mathbf{r}_v, \cdots, \mathbf{r}_N)} \\ &= V_l(r_v) \frac{2l+1}{4\pi} \int d\mathbf{\Omega}'_v P_l(\cos\theta'_v) \frac{\psi(\mathbf{r}_1, \cdots, \mathbf{r}'_v, \cdots, \mathbf{r}_N)}{\psi(\mathbf{r}_1, \cdots, \mathbf{r}_v, \cdots, \mathbf{r}_N)}, \end{aligned} \tag{8}$$

where P_l is the Legendre polynomial and θ'_v denotes the polar angle with the polar axis set to $\mathbf{r}_v$ now. The integral in Eq. (8) is over the unit sphere of the vth valence electron with respect to the nucleus and there are quite a few choices of integration quadrature suitable for this situation [32–35]. We decided to use the 12-point icosahedron quadrature in the calculation because it has acceptable accuracy as well as a reasonably small number of points. In the 12-point icosahedron quadrature, 12 points and their weights are properly chosen so that the integration on a unit sphere is exact for integrands with only $l \leqslant 5$ components when decomposed in the Y_{lm} basis, which suffices for our calculations. Specifically, for the 12-point icosahedron integral quadrature, we approximate the integration

$$J = \frac{1}{4\pi} \int d\mathbf{\Omega}\, f(\mathbf{\Omega}) \tag{9}$$

on a unit sphere by the sum

$$J = A \sum_{i=1}^{2} f(a_i) + B \sum_{i=1}^{10} f(b_i), \tag{10}$$

with the points a_i and b_i written in spherical coordinates (θ, ϕ) as

$$\begin{aligned} a_1 &= [0, 0], \quad a_2 = [\pi, 0] \\ b_i &= \begin{cases} \left[\arctan 2, \frac{2\pi}{5}(i-1)\right] & \text{for } 1 \leqslant i \leqslant 5, \\ \left[\pi - \arctan 2, \frac{\pi}{5}(2i-11)\right] & \text{for } 6 \leqslant i \leqslant 10, \end{cases} \end{aligned} \tag{11}$$

and the coefficients A and B are

$$A = B = \tfrac{1}{12}. \tag{12}$$

The coordinates of the 12 points are written in a certain coordinate system, whose orientation remains arbitrary and each choice can be related by a global rotation. Generally speaking, for an arbitrary integrand, its integral result varies in a different coordinate system, since the employed 12-point icosahedron quadrature is only exact for $l \leqslant 5$ components. In order to cancel out the error introduced by the orientation choices, we could average the results from randomly chosen orientations, and then the result would actually be an unbiased Monte Carlo estimation on a unit sphere.

In our calculation, however, we only chose one random orientation for each sample at one Monte Carlo step to have satisfactory training speed, with some sacrifices in the variance of the energy estimation. Although more orientation configurations in each estimation step would help decrease the variance, in our calculation, a one-orientation estimation has already given a satisfactory variance compared to that introduced from the MCMC steps.

TABLE I. Numerical setup.

Name	Value
Framework	JAX [36]
Main computing resource	8 Nvidia V100 GPU cards
Optimizer	KFAC [37]
Optimization hyperparameters	Same as in Refs. [13,14]
Batch size	4096
Number of HF pretrain iteration	3000
Number of training iteration	500 000
Number of inference iteration	100 000
Pretrain basis	ccECP-cc-pVDZ [25]

D. Calculation specification

Based on the open-sourced FermiNet package [13,14], the numerical setup for our calculations is listed in Table I. Note that in most cases we used the default values for hyper-parameters provided in the open-sourced repository for FermiNet. The neural networks are trained on the internal machine learning platform in ByteDance Inc., which supports elastic resources and large-scale training tasks.

TABLE II. Calculated ground-state energy in Hartree for Ga, Kr, and 3*d* transition metals, using various methods with ccECP: FermiNet + ECP, CCSDT(Q), FCI with both 5Z basis (aug-cc-pV5Z for Ga and Kr, and cc-pCV5Z for 3*d* transition metals) and CBS limit, and DMC using millions of determinants in its trial wave function. The results of the latter three methods are from Ref. [26]. CCSDT(Q) at CBS limit achieves the lowest energy for 3*d* transition metals, and our method performs the second best. For heavier transition metals such as Cu and Zn, our calculated energies are around 10 millihartrees higher than the CCSDT(Q)/CBS ones, while the other methods' results are more than 40 millihartrees higher. FCI result for Ga is listed instead of CCSDT(Q) since Ga only has three valence electrons with ECP employed.

Element	FermiNet+ECP	CCSDT(Q)/5Z	CCSDT(Q)/CBS	DMC(MD)
Sc	−46.5536(1)	−46.5494	−46.557 04(81)	−46.550(1)
Ti	−58.0895(1)	−58.0826	−58.092 63(76)	−58.085(1)
V	−71.4346(1)	−71.4274	−71.441 78(59)	−71.421(2)
Cr	−86.6349(1)	−86.6236	−86.641 09(33)	−86.625
Mn	−103.8828(10)	−103.8718	−103.8919(10)	−103.859(2)
Fe	−123.3754(1)	−123.3626	−123.388 04(93)	−123.358(2)
Co	−145.1409(1)	−145.1232	−145.1541(10)	−145.115(4)
Ni	−169.3760(1)	−169.3546	−169.3912(12)	−169.345(2)
Cu	−196.3873(1)	−196.3584	−196.4038(10)	−196.353(3)
Zn	−226.3516(2)	−226.3210	−226.3699(18)	−226.320(3)
Kr	−18.472 355(8)	−18.4660	−18.472 59(27)	−18.4680(1)
	FermiNet+ECP	FCI/5Z	FCI/CBS	DMC(MD)
Ga	−2.039 853(2)	−2.0396	−2.0399 15(13)	−2.0392(2)

III. RESULTS

In this work, we mainly study 3*d* transition metals, namely, from Sc to Zn, and their monoxides. Their ground-state energy and dissociation energy (DE) are relatively difficult to compute using traditional methods.[1] Moreover, ccECP can remove a significant amount of core electrons from calculations for those systems. For instance, a Sc atom has 21 electrons in total, while we only need to consider 11 electrons with ccECP. We also performed calculations on the elements Ga and Kr, where most electrons are treated as core electrons by ccECP, so as to show the benefits to computation efficiency. From now on, we refer to our ECP-based FermiNet method as FermiNet+ECP.

For the ground-state energy of 3*d* transition metals, our results are close to the state-of-the-art CCSDT(Q) at the complete basis set (CBS) limit and outperform projection methods such as fixed-node diffusion Monte Carlo (DMC) [26]. For the dissociation energy of transition metal monoxides, our results are consistent with highly accurate CCSD(T), semistochastic heat bath configuration interaction (SHCI), auxiliary field quantum Monte Carlo (AFQMC), and density matrix renormalization group (DMRG) results [25,39].

A. Atoms

In this section, we study the ground-state energy of atoms using FermiNet with ccECP. We carry out calculations on the elements Ga and Kr and 3*d* transition metals, with the results listed in Table II. For comparison, we list CCSDT(Q) and DMC results with ccECP for 3*d* transition metals; moreover, full configuration interaction (FCI) results are listed for Ga instead of CCSDT(Q) since Ga has only three valence electrons. The listed data are all from Ref. [26]. Note that all the compared methods are dealing with the same effective core Hamiltonian since they use the same ECP; hence, it is fair to make comparisons on the ground-state energy.

Among all the compared methods employed for 3*d* transition metals, CCSDT(Q) at CBS limit, despite its nonvariational nature and basis-set extrapolation error, has the lowest energy for all considered systems, and thus we use its results as the reference, against which the differences from other results are plotted in Fig. 1. Our method achieves the second-best result. In particular, our result is better than that of the CCSDT(Q) method with basis cc-pCV5Z, the largest finite basis set used in CCSDT(Q) calculation for those atoms to our knowledge, which is as expected since a neural-network-

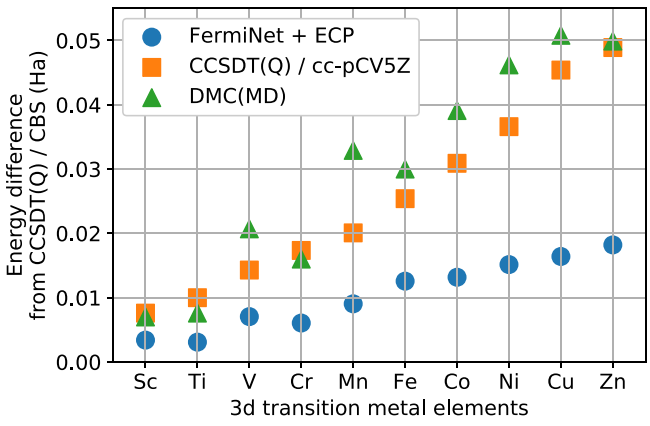

FIG. 1. Ground-state energy of 3*d* transition metals calculated using FermiNet+ECP (blue dots), compared with CCSDT(Q)/cc-pCV5Z (orange squares), and DMC (green triangles) results provided in Ref. [26], all with ccECP. Here we show the relative difference against CCSDT(Q)/CBS. It is clear that our method achieves better results than the compared ones, even though they are still around 10 millihartrees higher than the referential result for heavier atoms.

[1]Results calculated with various methods can be found in Refs. [25,26,38,39].

TABLE III. Comparison of calculated DE of 3*d* transition metal monoxides against the experimental data [41], the CCSD(T)/CBS result [25], and the calculated results from SHCI, AFQMC, and DMRG [39]. CCSD(T) results use ccECP and the associated basis while SHCI, AFQMC, and DMRG use TN ECP. Specifically, SHCI and AFQMC use the aug-ccpV5Z basis and DMRG uses the aug-ccpVDZ basis. Our DE result is calculated with the ground-state energy of both monoxides (listed here) and atoms (listed in Table II), and we use −75.066 55 hartree as the ground-state energy for atom O [13]. The bond length of monoxides in our calculation is the same as the one in the CCSD(T) result [25].

		Dissociation energy (hartree)					
System	Ground Sstate energy (hartree)	FermiNet+ECP	Experimental Data	CCSD(T)	SHCI	AFQMC	DMRG
ScO	−121.8765(1)	0.2564(2)	0.2566(3)	0.2550(11)	0.25684(1)	0.2585(16)	0.2422
TiO	−133.4073(1)	0.2513(2)	0.2548(25)	0.2517(11)	0.25444(1)	0.2568(16)	0.2397
VO	−146.7417(2)	0.2406(3)	0.2405(31)	0.2464(7)	0.24439(1)	0.2462(15)	0.2268
CrO	−161.8682(13)	0.1668(14)	0.1670(22)	0.1684(3)	0.17424(1)	0.1717(22)	0.1649
MnO	−179.0892(2)	0.1399(12)	0.1423(29)	0.1390(7)	0.14377(1)	0.1436(18)	0.1268
FeO	−198.5927(1)	0.1508(2)	0.1555(3)	0.1525(11)	0.15671(1)	0.1549(18)	0.1359
CoO	−220.3459(2)	0.1385(3)	0.1519(33)	0.1396(18)	NA	NA	NA
NiO	−244.5855(4)	0.1430(5)	0.1439(11)	0.1561(11)	NA	NA	NA
CuO	−271.5504(16)	0.0965(17)	0.1131(11)	0.1010(6)	0.10987(1)	0.1105(25)	0.1046
ZnO	−301.4705(3)	0.0523(5)	0.0608(14)	0.0536(18)	NA	NA	NA

related ansatz has the ability to approach the CBS limit [40]. Fixed-node DMC results with a multireference trial wave function are also plot in Fig. 1. DMC typically requires millions of determinants in its trial wave function in order to decrease the fixed-node error. In comparison, neural-network-based methods can achieve better accuracy than the DMC method with only dozens of determinants. Moreover, the gap between results from CCSDT(Q)with cc-pCV5Z and the CBS limit increases quite significantly as the number of electrons increases, and similarly for DMC. Comparatively, the discrepancy from our results to the referential ones is much smaller for heavier transition metals such as Cu and Zn, suggesting that the ECP method works well with FermiNet in those larger systems.

We also note that there is an insignificant difference between FermiNet+ECP and FCI+ECP/CBS results for Ga. There are two reasons that might account for the difference: FCI+ECP/CBS has a basis-set extrapolation error and the expressive ability of the FermiNet ansatz is restricted due to the finite number of neural network parameters.

B. Transition metal monoxides

In order to show our methods can be used to obtain high-quality *ab initio* data comparable to experiments, we compute the dissociation energy DE of 3*d* transition metal monoxides, defined as

$$\mathrm{DE}(X) = E(X) + E(\mathrm{O}) - E(X\mathrm{O}), \tag{13}$$

where XO denotes the monoxide of element X and $E(\cdot)$ denotes the ground-state energy. For monoxides, we only apply ECP to transition metal atoms while all electrons for the O atom are included, and we use −75.066 55 Ha for $E(\mathrm{O})$ (same as the result given in the original FermiNet paper [13]).

We compare our calculated DE against high-accuracy electronic structure methods such as SHCI, CCSD(T), DMC, DMRG, and AFQMC in Table III. Moreover, since DFT is broadly employed in computational studies of transition metal systems, we also list DFT results using the B3LYP exchange correlation functional for comparison. All these compared results are from Refs. [25,39], and we only take results calculated with the largest basis set provided. We use the same equilibrium bond length for monoxides as CCSD(T) results in the supplementary material of Ref. [25]. For completeness, we also list our calculated ground-state energy of monoxides in Table III.

To visualize the overall performance of the FermiNet + ECP calculation, in Fig. 2 we plot the fitted density of the difference between the calculated result and the experimental data [41] of the 3*d* transition metal monoxide DE for each method. Here we compare with the experimental data because they can serve as reasonable references when comparing

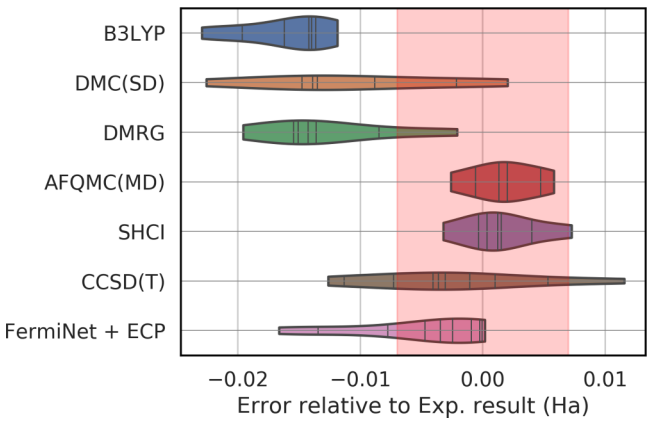

FIG. 2. Fitted density of the difference from calculated transition metal monoxide DE to the experimental data [41] in hartree for each method, where the violin shape corresponds to the distribution of calculated data. The CCSD(T) data (with ccECP) are from Ref. [25] while others (with TN ECP [24]) are from Ref. [39]. Individual data points are indicated by small vertical lines. The red shaded region, centered at 0 and with width 0.007 hartree, indicates the acceptable discrepancy between calculated results and experimental data. All the data from SHCI and AFQMC, as well as most data from our method and CCSD(T), fall into that region.

methods using different ECPs and basis sets. However, it is worth noting that experimental results are not completely self-consistent among different studies, which may differ from one to another by as much as 0.5 eV [39,41]. Therefore, we place a shaded region (with a width of 0.007 hartree) on Fig. 2 to indicate acceptable discrepancy from the experimental data. All data from very accurate methods such as AFQMC and SHCI fall into this interval. Our results mostly fall into this interval except for a long tail on the left, led by the CoO, CuO, and ZnO results. We comment on the issue associated with those monoxides later, but overall our calculations provide results almost as accurate as the current state-of-the-art electronic structure approaches. DMRG, DMC, and B3LYP data have less overlap with this interval. We want to point out that those methods perform less ideally for different reasons: DMRG is known to be an accurate method, but the heavy cost limits its calculation of these molecules to the aug-ccpVDZ basis set. So the basis set incompleteness error has dominated its DE underestimation. DMC is largely limited by the fixed-node approximation and the single determinant trial wave function employed. For transition metal oxides, a multiconfigurational wave function is needed and the fixed-node DMC based on the single determinant trial wave function does not perform satisfactorily. For DFT, it is known that the results strongly depends on the chosen exchange correlation functional. The comparison of different functionals is beyond the scope of this work, but we present the results of one functional, namely, the B3LYP functional, which is a hybrid functional known as one of the well-behaving functionals for transition metal oxides. Yet the results shows that B3LYP underestimates the DE by an average of 15 millihartrees. Therefore, we conclude that FermiNet + ECP is more accurate and reliable than DFT and should be promoted in future studies to tackle larger transition-metal-containing molecules and materials.

The fitted density shape of the CCSD(T) result is similar to ours, but it has long tails on both sides. The good performance of CCSD(T) on transition metal monoxides was also mentioned in a recent study using the full configuration interaction quantum Monte Carlo approach [38]. Here we show that FermiNet + ECP outperforms CCSD(T) slightly for the majority of 3*d* transition metal monoxides. In Fig. 3 we present a finer comparison between our method and CCSD(T) at the CBS limit [25], where the difference from the experimental data is compared for each system. For most of the monoxides, our results are close to the CCSD(T) ones, except for VO, NiO, and CuO. For VO and NiO, our results are very close to the experimental data. For CuO, both our result and the CCSD(T) one are quite far away from the experimental reference (greater than 10 millihartrees), while ours are farther away. We discuss more on our CoO, CuO, and ZnO results in the next section. Note that compared to CCSD(T), known as the golden standard in quantum chemistry, our method has not only comparable accuracy but also better computational scaling, namely, N^4, same as FermiNet, as opposed to CCSD(T)'s N^7, where N stands for the number of electrons considered in the computation.

Furthermore, we have carried out additional calculations for ScO with ECP used for both Sc and O. Our results are $E(\mathrm{O}) = -15.882\,62(4)$ hartree and $E(\mathrm{ScO}) = -62.6876(1)$ hartree. Note that the $E(\mathrm{ScO})$ here is calculated

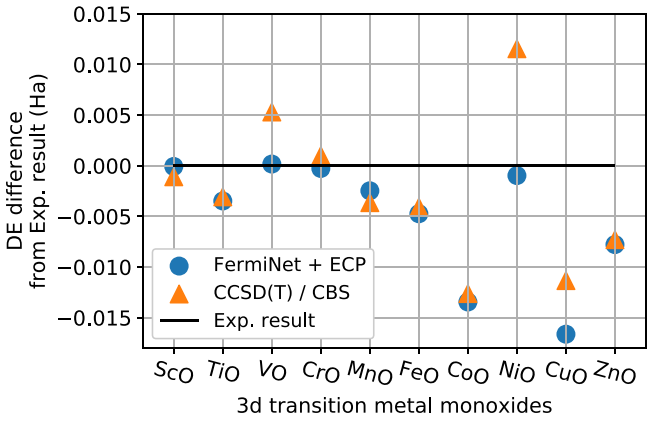

FIG. 3. DE of transition metal monoxides calculated using FermiNet + ECP (blue dots) compared to CCSD(T) (orange triangles) both with ccECP. We show the relative difference against the experimental data. Our results are quite close to the CCSD(T) ones, except for VO, NiO, and CuO, in which cases our results are closer to the experimental ones for VO and NiO.

with only 200 000 training iterations rather than 500 000 iterations in all-electron O simulation. The corresponding dissociation energy for ScO is 0.2514(1) hartree, which is slightly smaller than the dissociation energy with all-electron O 0.2551(1) hartree and 0.2564(2) hartree with 200 000 and 500 000 training iterations, respectively. Overall, we estimate that using ECP for O may lead to an underbinding of 3–4 millihartrees for the dissociation energy.

IV. DISCUSSION

In this section, we discuss some details of our calculations and model training.

A. Efficiency

Efficiency and scalability, as well as accuracy, are our main motivations for taking ECP into consideration. ECP reduces the number of electrons to deal with in FermiNet, but it also introduces the additional cost of numercial intergration of the ECP Hamiltonian. In neural networks, the cost of such numerical interaction has not been investigated. Whether the combination of FermiNet with ECP can have better computation efficiency and scalability than all-electron (AE) calculations needs to be examined.

Compared with the original FermiNet, the computational complexity for energy calculation is changed after ECP is employed, which reads

$$O((N_\mathrm{c}+N_\mathrm{v})^4) \xrightarrow{\text{Using ECP}} O\big(N_\mathrm{int}N_\mathrm{c}^4\big), \tag{14}$$

where N_c and N_v denote the number of core and valence electrons, and N_int denotes the number of integral quadrature points used in the ECP calculation. In our calculations, icosahedron quadrature is employed and $N_\mathrm{int} = 12$. We can see that the ECP method removes core electrons from the calculation at the cost of additional forward times for integrals.

TABLE IV. The number of electrons and runtimes of a single training iteration in AE and ECP calculations, respectively.

	AE		ECP	
Element	No. electrons	Single iteration runtime (s)	No. electrons	Single iteration runtime (s)
Ga	31	2.9	3	0.086
Kr	36	4.5	8	0.39
Sc	21	1.08	11	0.61
Ti	22	1.19	12	0.71
V	23	1.32	13	0.85
Cr	24	1.42	14	0.99
Mn	25	1.65	15	1.13
Fe	26	1.80	16	1.30
Co	27	2.00	17	1.55
Ni	28	2.11	18	1.73
Cu	29	2.32	19	1.99
Zn	30	2.59	20	2.16

Here we show the application of FermiNet + ECP on the elements Ga and Kr in which cases the ECP method has a significant advantage. With ccECP, we only need to consider 3 electrons for Ga and 8 electrons for Kr, while with all-electrons we have to deal with 31 and 36 electrons for Ga and Kr, respectively, which are already quite large systems for FermiNet to handle. The computation runtimes are listed in Table IV, where the data are obtained from 1000 training iterations by taking the average of their runtime, and we also ignore the first several iterations to avoid the effect of the initial warmup and/or the compilation process. It is apparent that, compared with the AE calculation, the simulation efficiency is highly improved with ECP employed.

In Table IV we also list the computation runtime of 3*d* transition metal atoms, for which 10 core electrons are considered in ccECP. Because of the fixed number of core electrons, the heavier the element, the smaller the difference between AE and ECP is. For Sc there is about a 50% reduction in the single-iteration runtime from ECP, whereas for Zn the reduction is less than 20%. Nonetheless, we find that our implementation of ECP in FermiNet has larger gains from electron reduction than losses due to integration of the ECP part, which suggests a promising future for applying various ECPs in fast-developing neural network electronic structure packages.

B. Optimizer comparison

Following the original work of FermiNet [13], we have tested both ADAM [42] and KFAC [37] optimizers when training neural networks with ECP. In terms of training runtime, we find that the choice of optimizer does not matter much, suggesting that the runtime is dominated by the forward pass, especially the numerical integration introduced by the ECP method. For instance, for the vanadium atom, one training step runs for around 0.8 s regardless of the optimizer. As for the model performance, KFAC can lead the neural network to a better state yielding lower energy. It also converges in much fewer iterations, as shown in Fig. 4 where we show atoms Sc, Ti, and V as examples. Energy optimized by KFAC approaches 1% of the correlation energy error after around 10 000 iterations, while the one optimized by ADAM is not close to that level of accuracy even after 1×10^6 iterations.

C. Model training details

Here we discuss a few technical issues occurred in calculating Co, Cu, Zn, and their monoxides, which may be related to the bad performances presented in the above sections. The first issue is that when training the neural network for these atoms, it may start to produce a Not-a-number (NAN) value after a number of iterations, which ruins the whole process. To resolve this issue, we removed outliers in terms of the local energy from the training process instead of simply doing clipping. Surprisingly, there was a magical threshold for outlier identification such that the training energy may be around 10 millihartrees lower than the result with a lower or higher threshold. In our case, such a threshold is 10 times the standard deviation of the local energy in one training batch. We did not observe such phenomena for monoxides and we speculate that it is because electrons are less likely to be too

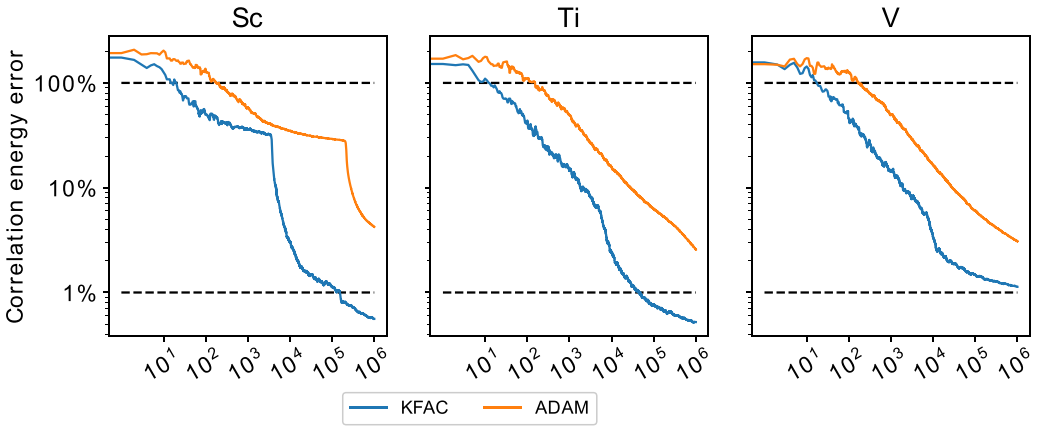

FIG. 4. The log-log plot of the optimization progress of FermiNet with ECP for Sc, Ti, and V using optimizers KFAC (blue lines) vs ADAM (orange lines). The horizontal axis is the number of training iterations. The vertical axis is the correlation energy error, calculated using CCSDT(Q)/CBS with ccECP provided in Ref. [26]. For clarity, we show the median energy over the last 10% of iterations.

close to the nuclei of transition metal atoms in molecules. Note that during the inference phase we do not do such outlier removal so that our calculated energy remains reliable and variational. Therefore, although this is an issue to be further investigated in future developments of FermiNet, we do not expect it is the main cause of the deviation of our results from experimental values.

Moreover, as mentioned in Sec. III B, our calculated DEs for CoO, CuO, and ZnO are underestimated when compared to the result of the most accurate methods and the experimental data. The CoO result is 13 millihartrees lower, the CuO result is 17 millihartrees lower, and the ZnO result is 8 millihartrees lower, in which cases the gap is more than 10% of their DE. Since our method is variational, it is more likely that there is an overestimation of the total energy of the monoxide. There are several potential reasons. First, we used the same bond length as CCSD(T) [25] for monoxides' equilibrium position, which may not be the optimal value for FermiNet + ECP. A fine search of the optimal bond length may improve the calculated DE by a few millihartrees. Second, those monoxides are notoriously strongly correlated and challenging systems for electronic structure calculations. Although we have made it possible with FermiNet + ECP, the system is at the limit of current computational capacity. Thus, it remains unclear whether a more powerful network with a larger number of layers and determinants can improve the description of such strongly correlated systems. Last but not least, we trained our neural networks up to 500 000 iterations, up to which the convergence of the training process becomes really slow. This maximum number of training iterations is chosen based on our affordability, but the slow convergence does indicate that the wave function represented by the neural network is still evolving slowly. For such challenging systems, we cannot rule out the possibility that further fine-tuning in the network can eventually lead to better results. How the network size and training process affect the performance of neural networks is certainly an interesting topic for future studies, especially for strongly correlated systems.

V. CONCLUSION

In this work, we have implemented the ECP method in the existing deep learning work, namely, FermiNet. Employing the ECP method pushes FermiNet towards studying larger systems by reducing the cost while retaining the accuracy to describe chemical bonding. Based on our implementation we carried out studies of 3*d* transition metal atoms and their monoxides, which are challenging systems for electronic structure methods. The calculated results are consistent with state-of-the-art computational methods and experimental data. Comparing to more widely used methods such as DFT and CCSD(T), FermiNet + ECP outperforms them in a broad range of systems, suggesting a bright future for deep learning techniques in molecules and materials modeling. Based on this study, we expect ECPs can also be successfully used in other neural networks, where benefits in computational efficiency are likely to be similar to our work. We also want to mention that investigating better network structures, such as the Transformer network structure [4], may further improve the accuracy and efficiency in solving electronic structures, and related works are under way.

ACKNOWLEDGMENTS

The authors thank Matthew Foulkes, David Ceperley, and Abdulgani Annaberdiyev for helpful discussions, Hang Li for support and guidance, the DeepMind FermiNet team for the open-sourced FermiNet software and prompt response on Github. Additional thanks to the ByteDance AML team for technical support, the ByteDance AI-Lab LIT Group for fruitful collaboration, and the rest of ByteDance AI-Lab Research team and collaborators for ideas and inspiration.

[1] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT, Cambridge, MA, 2016), http://www.deeplearningbook.org.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, Imagenet classification with deep convolutional neural networks, Commun. ACM **60**, 84 (2017).

[3] S. Ren, K. He, R. Girshick, and J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, in *Advances in Neural Information Processing Systems* (Curran Associates, Red Hook, NY, 2015), Vol. 28, pp. 91–99.

[4] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. U. Kaiser, and I. Polosukhin, Attention is all you need, in *Advances in Neural Information Processing Systems* (Curran Associates, Red Hook, NY, 2017), Vol 30.

[5] H. Li, *Learning to Rank for Information Retrieval and Natural Language Processing*, 2nd ed. (Morgan & Claypool, California, San Rafael, 2014).

[6] P. Covington, J. Adams, and E. Sargin, in *Deep Neural Networks for YouTube Recommendations* (ACM Press, New York, 2016), pp. 191–198.

[7] W. Gao, X. Fan, C. Wang, J. Sun, K. Jia, W. Xiao, R. Ding, X. Bin, H. Yang, and X. Liu, Deep retrieval: Learning a retrievable structure for large-scale recommendations, arXiv:2007.07203.

[8] J. Han, L. Zhang, R. Car *et al.*, Deep potential: A general representation of a many-body potential energy surface, Commun. Comput. Phys. **23**, 629 (2018).

[9] K. T. Schütt, P.-J. Kindermans, H. E. Sauceda, S. Chmiela, A. Tkatchenko, and K.-R. Müller, in *Schnet: A Continuous-Filter Convolutional Neural Network for Modeling Quantum Interactions* (Curran Associates, Red Hook, NY, 2017), pp. 992–1002.

[10] L. Zhang, J. Han, H. Wang, R. Car, and W. E, Deep Potential Molecular Dynamics: A Scalable Model with the Accuracy of Quantum Mechanics, Phys. Rev. Lett. **120**, 143001 (2018).

[11] J. Han, L. Zhang, and W. E, Solving many-electron Schrödinger equation using deep neural networks, J. Comput. Phys. **399**, 108929 (2019).

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

[13] 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. Res. **2**, 033429 (2020).

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

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

[16] J. Kessler, F. Calcavecchia, and T. D. Kühne, Artificial neural networks as trial wave functions for quantum Monte Carlo, Adv. Theor. Simul. **4**, 2000269 (2021).

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

[18] K. Choo, A. Mezzacapo, and G. Carleo, Fermionic neural-network states for ab-initio electronic structure, Nat. Commun. **11**, 2368 (2020).

[19] M. Wilson, N. Gao, F. Wudarski, E. Rieffel, and N. M. Tubman, Simulations of state-of-the-art fermionic neural network wave functions with diffusion Monte Carlo, arXiv:2103.12570.

[20] Y. Chen, L. Zhang, H. Wang, and W. E, Deepks: A comprehensive data-driven approach toward chemically accurate density functional theory, J. Chem. Theory Comput. **17**, 170 (2021).

[21] M. Dolg and X. Cao, Relativistic pseudopotentials: Their development and scope of applications, Chem. Rev. **112**, 403 (2012).

[22] M. Burkatzki, C. Filippi, and M. Dolg, Energy-consistent pseudopotentials for quantum Monte Carlo calculations, J. Chem. Phys. **126**, 234105 (2007).

[23] M. Dolg, U. Wedig, H. Stoll, and H. Preuss, Energy-adjusted *ab initio* pseudopotentials for the first row transition elements, J. Chem. Phys. **86**, 866 (1987).

[24] J. R. Trail and R. J. Needs, Shape and energy consistent pseudopotentials for correlated electron systems, J. Chem. Phys. **146**, 204107 (2017).

[25] A. Annaberdiyev, G. Wang, C. A. Melton, M. C. Bennett, L. Shulenburger, and L. Mitas, A new generation of effective core potentials from correlated calculations: 3d transition metal series, J. Chem. Phys. **149**, 134108 (2018).

[26] A. Annaberdiyev, C. A. Melton, M. C. Bennett, G. Wang, and L. Mitas, Accurate atomic correlation and total energies for correlation consistent effective core potentials, J. Chem. Theory Comput. **16**, 1482 (2020).

[27] M. Born and R. Oppenheimer, Zur quantentheorie der molekeln, Ann. Phys. **389**, 457 (1927).

[28] R. P. Feynman and M. Cohen, Energy spectrum of the excitations in liquid helium, Phys. Rev. **102**, 1189 (1956).

[29] Q. Sun, T. C. Berkelbach, N. S. Blunt, G. H. Booth, S. Guo, Z. Li, J. Liu, J. D. McClain, E. R. Sayfutyarova, S. Sharma, S. Wouters, and G. K.-L. Chan, PySCF: the python-based simulations of chemistry framework, WIREs Comput. Mol. Sci. **8**, e1340 (2018).

[30] W. M. C. Foulkes, L. Mitas, R. J. Needs, and G. Rajagopal, Quantum Monte Carlo simulations of solids, Rev. Mod. Phys. **73**, 33 (2001).

[31] S. Fahy, X. W. Wang, and S. G. Louie, Variational quantum Monte Carlo nonlocal pseudopotential approach to solids: Formulation and application to diamond, graphite, and silicon, Phys. Rev. B **42**, 3503 (1990).

[32] L. Mitáš, E. L. Shirley, and D. M. Ceperley, Nonlocal pseudopotentials and diffusion Monte Carlo, J. Chem. Phys. **95**, 3467 (1991).

[33] A. McLaren, Optimal numerical integration on a sphere, Math. Comp. **17**, 361 (1963).

[34] V. Lebedev, Values of the nodes and weights of ninth to seventeenth order Gauss-Markov quadrature formulae invariant under the octahedron group with inversion, USSR Comput. Math. Math. Phys. **15**, 44 (1975).

[35] V. Lebedev, Values of the base-points and weights of Gauss-Markov quadrature formulae for a sphere from the 9-th to the 17-th order of accuracy, invariant under the octahedron group with inversion, Zh. Vychisl. Mat. Mat. Fiz. **15**, 48 (1976).

[36] 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).

[37] J. Martens and R. Grosse, Optimizing neural networks with Kronecker-factored approximate curvature, in *Proceedings of the 32nd International Conference on International Conference on Machine Learning*, ICML'15 (JMLR.org, 2015), Vol. 37, pp. 2408–2417.

[38] T. Jiang, Y. Chen, N. A. Bogdanov, E. Wang, A. Alavi, and J. Chen, A full configuration interaction quantum Monte Carlo study of ScO, TiO, and VO molecules, J. Chem. Phys. **154**, 164302 (2021).

[39] K. T. Williams, Y. Yao, J. Li, L. Chen, H. Shi, M. Motta, C. Niu, U. Ray, S. Guo, R. J. Anderson, J. Li, L. N. Tran, C.-N. Yeh, B. Mussard, S. Sharma, F. Bruneval, M. van Schilfgaarde, G. H. Booth, G. K.-L. Chan, S. Zhang *et al.* (Simons Collaboration on the Many-Electron Problem), Direct Comparison of Many-Body Methods for Realistic Electronic Hamiltonians, Phys. Rev. X **10**, 011041 (2020).

[40] Z. SchÃ¤tzle, J. Hermann, and F. Noé, Convergence to the fixed-node limit in deep variational monte carlo, J. Chem. Phys. **154**, 124108 (2021).

[41] Y. A. Aoto, A. P. de Lima Batista, A. Köhn, and A. G. S. de Oliveira-Filho, How to arrive at accurate benchmark values for transition metal compounds: Computation or experiment?, J. Chem. Theory Comput. **13**, 5291 (2017).

[42] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, in *Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, May 7–9, 2015*.