Towards the ground state of molecules via diffusion Monte Carlo on neural networks

Weiluo Ren,1, * Weizhong Fu,1, 2, † Xiaojie Wu,1 and Ji Chen2,3, ‡
1 ByteDance Research, Zhonghang Plaza, No. 43, North 3rd Ring West Road, Haidian District, Beijing, People’s Republic of China
2 School of Physics, Peking University, Beijing 100871, People’s Republic of China
3 Interdisciplinary Institute of Light-Element Quantum Materials, Frontiers Science Center for Nano-Optoelectronics, Peking University, Beijing 100871, People’s Republic of China
(Dated: April 11, 2023)

Diffusion Monte Carlo (DMC) based on fixed-node approximation has enjoyed significant developments in the past decades and become one of the go-to methods when accurate ground state energy of molecules and materials is needed. However, the inaccurate nodal structure hinders the application of DMC for more challenging electronic correlation problems. In this work, we apply the neural-network based trial wavefunction in fixed-node DMC, which allows accurate calculations of a broad range of atomic and molecular systems of different electronic characteristics. Our method is superior in both accuracy and efficiency compared to state-of-the-art neural network methods using variational Monte Carlo (VMC). We also introduce an extrapolation scheme based on the empirical linearity between VMC and DMC energies, and significantly improve our binding energy calculation. Overall, this computational framework provides a benchmark for accurate solutions of correlated electronic wavefunction and also sheds light on the chemical understanding of molecules.

I. INTRODUCTION

Since the establishment of quantum wavefunction theory by Erwin Schrödinger, ab initio electronic structure calculation has become one of the holy grails in chemistry [1, 2]. Molecules generally consist of a set of nuclei bonded together via electrons through electrostatic interactions. Therefore, the ground state electronic structure, i.e. the many body electronic wavefunction, is very much the most fundamental property, based on which we form the basic understanding of molecules. On top of the ground state wavefunction solution, one may further study electronic excitation, calculate nuclear forces and vibrations, optimize molecular structures, model dynamics and reactions, etc. [3]. Approximated methods, such as density functional theory and post Hartree-Fock methods have been widely employed for these purposes, but challenges still exist when high accuracy is needed [4, 5]. For instance, the sub-chemical-accuracy is often desired to predict adsorption of molecules on surfaces, the packing order of organic chemicals, and the hydrogen bonding of water and biological molecules [6, 7]. Therefore, pushing the limit towards the exact ground state wavefunction of molecules is of both fundamental importance and practical relevance.

Stochastic approaches, i.e. quantum Monte Carlo (QMC) methods, have been a competitive rival of the deterministic methods in chasing the ground truth of many body electronic wavefunction of molecules [8–11]. In particular, diffusion Monte Carlo (DMC), an approach based on ground state projection, is capable of treating dynamic correlations and reaching sub-chemical-accuracy for molecules [12, 13]. However, effective DMC algorithms usually work together with the so-called fixed-node approximation [14, 15], and the accuracy is only assured when a good trial wavefunction containing the correct nodal structure is provided in advance [16]. Despite many progresses have been made to improve the trial wavefunction, e.g. using physically more meaningful ansatz or combined with multi-determinant post Hartree-Fock wavefunctions [13, 17, 18], the fixed-node approximation remains as the Achilles’ heel of DMC.

Recently, it has been shown that machine learning techniques such as neural networks can lend strong support to describe the electronic structure of molecular systems and provide a powerful way to reconstruct the many body wavefunction [19–26]. FermiNet is one of the notable examples, which has already shown promising results for small molecules consisting of typically less than 30 electrons [20, 21, 27]. In these neural network wavefunction methods, variational Monte Carlo (VMC) is often employed to train the network on the fly. Despite its effectiveness on small molecules, it remains to be challenging to apply neural-network based VMC on larger systems, due to required large computation resources and long converging time.

In this work, we integrate the FermiNet neural network wavefunction into DMC. This approach takes advantage of the accurate trial wavefunction of FermiNet and the efficient ground state projection of DMC, which allows calculations of a range of systems to unprecedented accuracy. We refer to the vanilla FermiNet approach as FermiNet-VMC, and refer to our FermiNet-based DMC approach as FermiNet-DMC. Compared to FermiNet-VMC, FermiNet-DMC is able to achieve lower variational ground state energy at reduced computational cost. We
carry out tests on atoms as well as molecules including \( \text{N}_2 \), cyclobutadiene, water dimer, benzene and benzene dimer. We also present the empirical linear relation between VMC and DMC energies in our calculations and introduce an extrapolation scheme accordingly. Insights to the electronic structure of these systems obtained from our calculations are also discussed.

II. RESULTS

A. Computational framework

As illustrated in Fig. 1a, in the traditional electronic structure approach, diffusion Monte Carlo is often used after optimization of trial wavefunction using VMC, which approaches the limit of a given wavefunction ansatz. DMC further purifies the true ground state out of other contaminating eigenstates, and it often allows the breaking through of the ansatz limit. However, to overcome the notorious sign problem, nodal points where the wavefunction is zero have to be fixed in DMC, and walkers are only allowed to evolve in each fixed nodal pocket. Here, the idea is to implement the recently developed neural network as an accurate wavefunction ansatz (Fig. 1b). On one hand, the wavefunction learned by the neural network automatically reproduces an accurate representation of the mysterious nodal structure of many electrons of molecules. The accurate nodal structure ensures that the subsequent DMC simulation with fixed nodes does not yield bias to the ground state. On the other hand, compared with neural networks based VMC, our scheme only requires the information of the nodal structure instead of the full wavefunction. It is reasonable to expect the nodal structure to be simpler characterized than the full wavefunction.

Our multi-walker DMC algorithm is implemented in a fully parallel manner, in which each walker independently simulates the stochastic dynamics of electrons (Fig. 1c). The three key steps in our DMC algorithm are diffusion, branching, and merging (Fig. 1d), and they ensure that the equilibrium is reached for each walker after simulation in terms of the probability distribution of different electronic configurations. The diffusion step changes the configuration of electrons from one to another, while the cross-node movement is forbidden. Branching and merging control the total population of walkers during the simulation. In this work, we have implemented a GPU and neural network friendly DMC algorithm, which can be easily scaled out to multiple computing nodes. The runtime for one step in FermiNet-DMC is almost identical to that in FermiNet-VMC. Therefore, to compare the efficiency or total runtime between FermiNet-DMC and FermiNet-VMC, we only need to compare the number of steps in those processes. More methodological and technical details are provided in the Method section and the Supplementary Note 1-5.

B. Single atoms

Neural network models are faced with the trade-off between model expressiveness and computational intensity. For powerful models like FermiNet, it may take hundreds of thousands iterations to converge in the training process even for small benchmark systems with just a few electrons. Fig. 2 shows calculations for single atoms with a shallow and narrow FermiNet ansatz with only 2 layers of rather small number of neurons (see Supplementary Table S3 for details). The network is designed to be restricted so that we can study FermiNet’s performance when it is not expressive enough for the considered systems. This situation is of practical importance especially when we are interested in applying neural network based QMC methods to larger systems of one hundred electrons or more. As shown in Fig. 2a, a common pattern of FermiNet’s training progress is that the energy curve drops to a fairly low level in a short amount of time and then slowly converges to its limit. Fig. 2a is a calculation on the Be atom with the mentioned small network, and after \( 5 \times 10^5 \) steps of training which ensures complete convergence, the systematic error still can not be reduced to within the chemical accuracy. In addition, the computational cost could scale up quickly for larger systems even on the most advanced modern computation platforms such as NVIDIA’s Tesla A100 GPU. This issue prevents accurate calculations for more than 30 correlated electrons [20, 21].

The combination of the FermiNet neural network wavefunction ansatz and DMC achieves a substantial improvement in both accuracy and efficiency. For Be atom and the same simple neural network, FermiNet-DMC energy drops to within 1 mHa with respect to the reference value of the total energy. The DMC data is obtained with \( 10^5 \) steps of simulation, and the variance of DMC is also significantly reduced. It is also encouraging to see that even when we start from the trial wavefunction after \( 10^4 \) steps of training, the DMC energy obtained subsequently is also converged within 1 mHa to the exact value. At the \( 10^5 \) step when the training has not yet completely converged, the DMC energy is already consistent with the result obtained at the \( 5 \times 10^5 \) step. The good performance of DMC based on undertrained trial wavefunctions suggests the nodal structure is well characterized before the wavefunction is fully trained in the neural network ansatz. In Fig. 2b-d, we present the three-dimensional cuts of the full 11-dimensional (11D) nodal structure of the FermiNet wavefunction at the initial, \( 10^4 \), and \( 10^5 \) step. The \( 10^5 \) step nodal structure is very well converged to the correct one obtained from CI calculations [30], and the nodal structure at \( 10^4 \) step is also qualitatively same, explaining the high accuracy obtained subsequently using DMC. For comparison, the nodal structure of the initial wavefunction is also shown. Because of the fact that only the nodal structure determines the accuracy of DMC, the training process of neural network functions can be significantly shortened. Overall, to reach chemi-
Fig. 1. **Computational framework.** a. A sketch of a brief overview on variational Monte Carlo (VMC) and diffusion Monte Carlo (DMC) from the perspective of eigenstates composition. Atomic orbitals represent different eigenstates, and histograms indicate the weight of each eigenstate in the state decomposition. Top: A randomly initialized state with no dominant eigenstate. Middle: The output state of VMC where the ground state dominates, but other eigenstates are still non-negligible due to ansatz limitations. Bottom: The output state of DMC, which surpasses ansatz limitations and reaches the ground state. b. Left: a neural network ansatz of wavefunction; right: one dimensional projection of a many electron wavefunction and its nodal surface. c. Left: parallelized diffusion Monte Carlo processes on GPU; right: zoom in to the stochastic dynamics of each walker containing configurations of all electrons in the system, while the nodal structure is fixed. This panel is inspired by and adapted with permission from the website of Quantum Monte Carlo for Chemistry @ Toulouse (http://qmcchem.ups-tlse.fr/index.php/Quantum_Monte_Carlo_for_Chemistry_@_Toulouse) [28]. d. Three key steps in diffusion Monte Carlo. Each walker is assigned with a weight which evolves every iteration. Diffuse: The stochastic propagation of walkers without crossing the nodal surface. Branch: Split one walker when its weight becomes too large. Merge: Merge two walkers when their weights become too small.

Cal accuracy for Be atom, the cost of FermiNet-DMC is only a fraction of the cost of FermiNet-VMC.

Fig. 2e further shows the energy of FermiNet-based VMC and DMC for different atoms in order of the number of electrons under the same 2-layer network. We try different learning rates and train enough iterations ($10^6$ for S, Cl, Ar and $5 \times 10^5$ for the other atoms) to ensure that we make full use of the expressive power of the network. As expected the error of VMC increases when the number of electrons in the system increases and the complexity of the system gradually exceeds the expressive limits of the neural network. With DMC the errors are reduced by more than half. The dashed lines are linear fittings of the VMC and DMC energy. And the deviation of the data points from the fitting curves indicates that there is a correlation between the DMC and the VMC energy: when the VMC energy is comparably better, the DMC error is also smaller. The linear rising of DMC error shows that the training of nodal structure also becomes increasingly difficult when system size increases, and the correlation between the VMC and the DMC error indicates the information of the nodal structure is closely entangled with the full wavefunction. Note that we use a 2-layer network here in order to examine the behavior of FermiNet VMC and DMC in the regime where the network ansatz is relatively restricted for the considered systems, while FermiNet-VMC can be more expressive to achieve high accuracy for those atoms with more layers and neurons, as shown in [20].

Moreover, the improvement of DMC suggests that it may take a smaller and hence more efficient network to represent the nodal surface, without affecting the DMC accuracy. In Fig. 2f, we present a set of such tests on Ne atom, where the complexity of the neural network is labeled as (L,D,W) to indicate the number of layers, the number of determinants, and the width of each layer in the network, respectively. Overall, when the expressiveness of the network is reduced both VMC and DMC are affected in terms of their accuracy. Therefore, all the calculations suggest that the VMC energy is a good indi-
FIG. 2. Accuracy and efficiency of FermiNet-DMC on single atoms. a. The blue line shows the energy of a 2-layer FermiNet as a function of training iteration for a Be atom. The orange symbols show the DMC energy obtained with the trial wavefunction at the 10\textsuperscript{4}, 10\textsuperscript{5}, and 5 \times 10\textsuperscript{4} training iteration, respectively. The red dashed line shows the final convergence of VMC energy. b-d. The three isosurfaces show the three-dimensional cuts of the full 11D nodal structure obtained at different training iterations (See Methods, section Nodal structure and wavefunction visualization for plotting details). e. The calculated energies of atoms with respect to the reference ground state energy as the number of electrons increases with a 2-layer network. The dashed lines are linear fittings on the second and the third period elements, respectively. f. The energy error with different settings of FermiNet for a Ne atom. Three variables are respectively the number of layers (L), the number of determinants (D), and the width (W) of each layer. All the energy reference values are from Chakravorty et al. (1993)\cite{29}. cator of not only how well the wavefunction is optimized but also the quality of its nodal structure. The behavior is also expected for other neural network wavefunction ansatz. Combined with the typical first-steep-then-flat optimization curve in neural networks, we can automate the switching-on of DMC and minimize the total cost of calculations at targeted accuracy.

Building upon the successful treatments of FermiNet-DMC on atoms, we now extend the approach to larger molecules.

C. Nitrogen molecule

The first example is the dissociation curve of N\textsubscript{2} molecule. At equilibrium N\textsubscript{2} forms a strong triple covalent bond at 2.1 a.u., and the dissociation is accompanied by a severe bond breaking process, which is strongly correlated in nature. Therefore, the dissociation curve of N\textsubscript{2} is often used to benchmark electronic structure methods’ description of strong correlation. In DMC, this is also highly relevant because the nodal structure is directly affected by electron correlation. Fig. 3a plots the relative energy of N\textsubscript{2} with respect to the experimental reference \cite{31} as a function of the bond length. The results from FermiNet-VMC and r12-MR-ACPF, a state-of-the-art traditional multi-reference approach \cite{32}, are also shown. We can see that our DMC calculations are consistently better than those references, with an error of less than 1 mHa in a wide range of bond length. The largest error comes, not surprisingly, around the dissociation point near 4 a.u., and yet the error is only 3 mHa. In fact, our results can be considered as the most accurate \textit{ab initio} one of N\textsubscript{2} dissociation curve reported so far. It is worth noting that the FermiNet-VMC results here have been remarkably accurate, whose deviation from experiment curve is within 2 mHa near equilibrium and 4 mHa in dissociation region. Yet our FermiNet-DMC results can improve averagely about 1 mHa. For comparison, CCSD(T) calculation (not plotted), which is known as the “golden standard” in quantum chemistry, have an error of 25 mHa around 4 a.u. \cite{20}. In terms of relative energy, the non-parallelity error (NPE) of FermiNet-DMC (3.28 mHa) is only slightly better than that of FermiNet-VMC (3.53 mHa), consistent with mild improvement on small systems reported in \cite{37}, and both are comparable to the state-of-the-art r12-MR-ACPF result (2.14 mHa).

The remaining error source of DMC is the nodal structure error produced in the training of neural network using VMC, which is fully reflected on the shape of the Fer-
FIG. 3. Calculations on N\textsubscript{2}, cyclobutadiene and water dimer. a, Main panel: calculated energy of N\textsubscript{2} at different bond length, plotted as the difference to the experimental data [31]. For comparison, the green line is the highly accurate r12-MR-ACPF results under a modified basis set based on aug-cc-pV5Z [32]. Inset: the dissociation curves from experimental data (black line) and FermiNet-DMC (orange squares). The negligible error bars (less than 0.1 mHa) are not plotted. The pink backgrounds highlight the dissociation region where correlations are strong. b, Molecular structures of cyclobutadiene’s equilibrium state (bottom) and transition state (top). c, Main panel: the ground state energy of cyclobutadiene’s equilibrium state as a function of the VMC training step. FermiNet-DMC energy is calculated using the trial wavefunction at the corresponding training steps. FermiNet-VMC* indicates the result from Spencer et al. [21]. Inset: the transition barrier of cyclobutadiene calculated with different methods. Pink background indicates the range of experimental estimates between 2.5 and 15.9 mHa, while we show only the part above 9 mHa to highlight differences between QMC results. Grey dashed lines indicate results from five multi-reference coupled cluster (MRCC) methods (top to bottom): MR-DI-EOMCCSD, RMRCCSD(T), Mk-MRCCSD(T), MRCISD+Q and BW-MRCCSD(T) [33]. d, Left: three of the relative energies of the 10 Smith stationary points SP\textsubscript{n} (n = 1, 2, ..., 10) [34]. For the other results, see Supplementary Fig. 6. The SP1 structure is the global minimum and is taken as the reference. The geometries of SP1 and SP3 are shown as insets while the others are included in Supplementary Fig. 4. Two neural networks have been trained for 10\textsuperscript{5} steps and 3 × 10\textsuperscript{5} steps, which are dubbed as “undertrained” and “well-trained”. All the geometries are optimized by CCSD(T) [35]. The CCSD(T) energies and the DMC results with the conventional Slater-Jastrow ansatz [36] are also plotted for comparison. The error bar of the energy difference is calculated as the square root of the sum of the squares of each energy estimator’s standard error. Right: mean absolute deviation from CCSD(T) results over all the relative energies.

A similar example is the structural transition of cyclobutadiene, which is also well known for its multi-referential nature. The neural network based VMC models [21, 22] have already shown promising results on cyclobutadiene. FermiNet-DMC can handle this system with higher accuracy and reduced computational cost.

In our experiments, VMC process takes around 3 × 10\textsuperscript{5}
steps to converge, though the converged result is still around 7 mHa higher than the reported value in ref. [21], which converges in $2 \times 10^5$ steps. This is probably because we use different training hyperparameters, or simply because our optimization process gets trapped in a bad local minimum. However, our final DMC result is around 4 mHa lower than the reference data [21]. This demonstrates the effectiveness of our DMC implementation as a seamless extension to VMC. Namely even if the optimization in VMC does not work well, the following DMC process can still bring the energy calculation to a highly accurate level. This is especially important for neural-network based VMC, because its optimization is significantly trickier to tune and requires a longer time to completely converge, compared to conventional VMC. Here the DMC finite time step error is negligible as illustrated in Supplementary Fig. 1, which guarantees the variational property of our FermiNet-DMC results.

With $10^5$ VMC and $10^5$ DMC steps, FermiNet-DMC’s energy result is 2 mHa lower than the reference data in ref. [21] produced from a training phase with $2 \times 10^5$ VMC steps. Note that in this case our number of total QMC steps is still slightly less than the ref. [21] due to the required inference phase in FermiNet-VMC. Therefore, FermiNet-DMC should be preferred for its lower variational energy at the same or less computational cost.

The automerization energy difference of cyclobutadiene is shown in the inset panel of Fig. 3c. Neural-network based VMC gives an accurate automerization energy difference of cyclobutadiene [21, 22]. It is consistent with the high-end of the experimental data. The results of FermiNet-DMC are also in the same region. See Supplementary Note 9 for more details, including the training curve for transition configuration and the DMC energy data for both equilibrium and transition configurations.

### E. Water dimer

In addition to the strong covalent bonding, where static correlation is more essential, molecular systems with weaker hydrogen bonding and non-covalent interactions can also be challenging because of dynamic correlations. To this end, we have carried out FermiNet-DMC calculations on the 10 Smith stationary point of water dimer [34]. The 10 structures, as illustrated in Fig. 3d and Supplementary Fig. 4, have different hydrogen bonding configurations and their relative energies are used to benchmark the performance of electronic structure methods and force field models on hydrogen bonding systems [38]. With 10 total energy results (plotted in Supplementary Fig. 5) and 9 relative energy results (plotted in Supplementary Fig. 6), we can thus have a rather credible investigation on the error cancellation performance of FermiNet-VMC and FermiNet-DMC. We compare the energy results of FermiNet-DMC with an undertrained network and a well-trained network as trial wavefunctions respectively. The undertrained network is trained by VMC in $10^6$ steps, while the well-trained network is trained by VMC in $3 \times 10^5$ steps. CCSD(T) results [35] are displayed as benchmarks for their high accuracy for such type of systems.

As shown in Fig. 3d, the undertrained FermiNet-VMC performs badly on SP3 and SP5, and so does the well-trained FermiNet-VMC on SP4, though some of the FermiNet-VMC results are quite close to the benchmark results (e.g. SP7 and SP8 in Supplementary Fig. 6). On the other hand, FermiNet-DMC performs consistently well no matter which network is used as trial wavefunction, undertrained or well-trained. Overall, the mean absolute deviations from the benchmark CCSD(T) results are also given in Fig. 3d, from which we can clearly tell the improvement of FermiNet-DMC on relative energy calculations. For comparison purpose, we also show DMC results with traditional Slater-Jastrow wavefunction ansatz [36], whose accuracy is at the same level with FermiNet-DMC as the difference is negligible compared to the statistical error. The inferior performance of FermiNet-VMC may be due to the different degree of convergence in different systems, while FermiNet-DMC provides a more efficient and practical solution than fully converged FermiNet-VMC.

### F. Benzene

To further illustrate the power of our approach, we have examined the benzene molecule and a benzene dimer. Benzene is one of the most fundamental organic molecules with a hexagonal ring of C-H (Fig. 4a). There have been challenges in understanding its electronic configuration, bonding order and obtaining the ground state energy. To understand the electronic structure of benzene molecule, we performed FermiNet based VMC and DMC simulations with 3-layer and 4-layer networks separately. Our best FermiNet-DMC result calculated with the 4-layer network coincides with the CCSD(T) result extrapolated to complete-basis-set (CBS) limit. The comparison is shown in Fig. 4d. The CCSD(T) result is carried out with Psi4 [39] and the CBS result is extrapolated using cc-pCVXZ (X=3,4,5) basis, which is much larger than the ones reported in [40] and used by others as the state-of-the-art electronic structure methods in Refs. [41]. The energy from our CCSD(T) / CBS calculation is also much lower than those references. See Supplementary Note 14 for more details on the CCSD(T) / CBS calculation and CBS extrapolation.

The 3-layer FermiNet here is much smaller than the 4-layer one. Besides being one layer shallower, the number of neurons on each layer is also significantly less. See Supplementary Tables 8-10 for the related hyperparameters. Fig. 4d shows that the 3-layer FermiNet-DMC’s energy is lower than the 4-layer VMC result by around 10 mHa, which demonstrates one of the main benefits of FermiNet-DMC that it can achieve better accuracy with smaller network. This is especially important when we
are dealing with large systems. In our calculations, FermiNet-DMC is able to achieve lower variational energy results with an order of magnitude better efficiency. With a total of $4 \times 10^5$ QMC steps ($2 \times 10^5$ VMC training steps and $2 \times 10^5$ DMC steps), the 3-layer FermiNet-DMC’s energy result (-232.225 Ha) is slightly better than the 4-layer FermiNet-VMC’s energy result (-232.223 Ha) at $10^6$ VMC training step. Moreover, the runtime of a single VMC step for the 4-layer network is approximately 4 times that of a single VMC or DMC step for the 3-layer network under the same computation resources. Therefore, in this case, the 3-layer FermiNet-DMC can achieve a better energy result at only a tenth of the total computation cost compared to the 4-layer FermiNet-VMC. Similarly, compared to the 3-layer FermiNet-VMC at $2 \times 10^6$ VMC training step, the 3-layer FermiNet-DMC with $4 \times 10^5$ QMC steps can achieve more than 10 mHa better energy result at only a fifth of the total computation cost.

Furthermore, the energy difference between the FermiNet-DMC results in Fig. 4d is only around 3 to 4 mHa, suggesting the closeness between the node structure of the two trial wavefunctions. To confirm this statement, we visualized 2-dimensional slices of those trial wavefunctions in Fig. 4b-c. The slices are generated by moving a single spin-up electron inside a two dimensional box while fixing all other electrons at representative positions suggested by Liu et al. [42] and illustrated in Fig. 4a. See Section IV E and Supplementary Note 11 for more visualization details. Comparing Fig. 4b (4-layer FermiNet VMC) and Fig. 4c (3-layer FermiNet), we find that the nodes, represented by the dark pixels, do share the same pattern. Moreover, the parts of nodal surface in lighter areas, namely with larger wavefunction value, are very close to each other in Fig. 4b-c, and they are the most important parts of nodal surface in the DMC process since walkers are more likely to visit its neighbourhood. The closeness of those parts is consistent with the fact that the FermiNet-DMC energies are close.

To track how nodal surface evolves along the training process, we propose a divergence $D(S,T)$ measuring the difference between two nodal surfaces $S$ and $T$. The definition and algorithmic details are described in section IV F and Supplementary Note 15, and the definition is also related to the intuition mentioned above that nodes in the neighbourhood with larger wavefunction value are more important in the QMC calculation. For the 3-layer FermiNet, we calculated

$$D(S_{\text{final}}, S_k),$$

where $S_{\text{final}}$ and $S_k$ are the nodal surface corresponding to the final VMC training step and the intermediate training steps $k$, respectively. The result is shown in Fig. 4e together with the VMC and DMC energy, where the trend of the divergence correlates well with energies. As a matter of fact, there is a linear relation between the divergence and DMC energy, as shown in Fig. 4f, indicating that the proposed divergence successfully captures the essential information of the difference between nodal surfaces. Here the divergence converges to around 0.005 instead of 0 because of the large learning rate used when training the 3-layer FermiNet for benzene.

We have also trained a neural network for a benzene dimer, which is a prototypical system to further test non-covalent interactions. Benzene dimer, which has 84 electrons in total, is a much larger system than the ones considered in previous neural-network based VMC works [19–26]. We elaborate the challenges and tricks dealing with large systems using FermiNet based QMC methods in Supplementary Note 5. We consider a T-shaped structure with an edge-to-face arrangement, as illustrated in Fig. 5a, specifically the equilibrium configuration with a center-to-center distance of 4.95 Å [43]. Fig. 5a also shows the VMC and DMC energies as functions of the VMC training step, which are both over 200 mHa lower than the CCSD(T) result with cc-pCVTZ basis. The converged FermiNet-DMC energy is over 50 mHa lower than both FermiNet-VMC result and the CCSD(T) result with cc-pCVQZ basis. It echos statements made in above sections that FermiNet-DMC can achieve significantly higher accuracy for larger systems or cases where the neural network ansatz is not powerful enough to characterize the ground state wavefunction well. For comparison, the FermiNet-VMC energy has not fully converged even after four million training steps. Schätzle et al. [44] shows that neural-network based VMC, in particular, PauliNet, can achieve variational energies at the fixed-node limit in certain circumstances, while in our calculations, one can clearly see that it is not the case for FermiNet especially when its expressive power is limited compared to the size of the system. On the other hand, our DMC result is 15 mHa higher than the CCSD(T) / CBS result. Note that CCSD(T) is not a variational method, hence the relatively lower CCSD(T) / CBS result may indicate similar accuracy compared to our DMC result. To achieve more accurate FermiNet-DMC result, we can use a better neural-network trial wavefunction with a larger network or a better network architecture.

In addition to the total energy at the equilibrium configuration, binding energy is also of great interest when studying a benzene dimer [43, 45, 47–49], and classical methods, such as CCSD(T) and MP2, can produce results agreeing with experimental data well. However, for neural-network based QMC, the binding energy calculation is more subtle and challenging due to the lack of systematic error cancellation. Using the same network structure handling both monomer and dimer would introduce additional size-inconsistency-like bias because of the more severe expressiveness limitation on benzene dimer than monomer. For the benzene dimer, we find such an estimate would predict a severe underbinding with both VMC and DMC. Another way to estimate the binding energy is to take the difference between a separated dimer configuration (10 Å) [45] and the equilibrium configuration, shown in Fig. 5b, which turns out to be systematically overbinding. With an empirical linear relation
between VMC and DMC energy in the training process, we developed a simple VMC-DMC hybrid extrapolation scheme, which leads to an accurate estimate of the binding energy, well-agreed with the experimental measurements [46], also displayed in Fig. 5b. We will elaborate more on this extrapolation scheme in Section II G. In order to systematically improve the binding energy calculation, the most straightforward way is to adopt better neural-network ansatz as the trial wavefunction for better accuracy. Adding regularization mechanism in the optimization processes is another possible option so that the model variance can be reduced for better error cancellation. Note that in the case of DMC with pseudopotential, binding energy calculation can be also improved with certain deterministic approximation [50]. We will leave it as a future study apply those ideas to improve the binding energy calculation.

G. Linear relation between VMC-DMC energy

Quite consistently, we find linear relation between VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.

We find that, for molecular systems, such as cyclobutadiene, benzene monomer and dimers, there’s a linear trend between the VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.

We find that, for molecular systems, such as cyclobutadiene, benzene monomer and dimers, there’s a linear trend between the VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.

We find that, for molecular systems, such as cyclobutadiene, benzene monomer and dimers, there’s a linear trend between the VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.

We find that, for molecular systems, such as cyclobutadiene, benzene monomer and dimers, there’s a linear trend between the VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.

We find that, for molecular systems, such as cyclobutadiene, benzene monomer and dimers, there’s a linear trend between the VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.

We find that, for molecular systems, such as cyclobutadiene, benzene monomer and dimers, there’s a linear trend between the VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.

We find that, for molecular systems, such as cyclobutadiene, benzene monomer and dimers, there’s a linear trend between the VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.

We find that, for molecular systems, such as cyclobutadiene, benzene monomer and dimers, there’s a linear trend between the VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.

We find that, for molecular systems, such as cyclobutadiene, benzene monomer and dimers, there’s a linear trend between the VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.

We find that, for molecular systems, such as cyclobutadiene, benzene monomer and dimers, there’s a linear trend between the VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.

We find that, for molecular systems, such as cyclobutadiene, benzene monomer and dimers, there’s a linear trend between the VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.

We find that, for molecular systems, such as cyclobutadiene, benzene monomer and dimers, there’s a linear trend between the VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.

We find that, for molecular systems, such as cyclobutadiene, benzene monomer and dimers, there’s a linear trend between the VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.

We find that, for molecular systems, such as cyclobutadiene, benzene monomer and dimers, there’s a linear trend between the VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.

We find that, for molecular systems, such as cyclobutadiene, benzene monomer and dimers, there’s a linear trend between the VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.

We find that, for molecular systems, such as cyclobutadiene, benzene monomer and dimers, there’s a linear trend between the VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.

We find that, for molecular systems, such as cyclobutadiene, benzene monomer and dimers, there’s a linear trend between the VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.

We find that, for molecular systems, such as cyclobutadiene, benzene monomer and dimers, there’s a linear trend between the VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.

We find that, for molecular systems, such as cyclobutadiene, benzene monomer and dimers, there’s a linear trend between the VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.

We find that, for molecular systems, such as cyclobutadiene, benzene monomer and dimers, there’s a linear trend between the VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.

We find that, for molecular systems, such as cyclobutadiene, benzene monomer and dimers, there’s a linear trend between the VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.

We find that, for molecular systems, such as cyclobutadiene, benzene monomer and dimers, there’s a linear trend between the VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.

We find that, for molecular systems, such as cyclobutadiene, benzene monomer and dimers, there’s a linear trend between the VMC and DMC energies in our calculation. We have encountered two types of linear relation. One is about intermediate energies calculated along the training process for a given system, while another one is about the converged energies from different systems. We take advantage of the first type of linearity and develop a simple but effective extrapolation scheme accordingly.
FIG. 5. Benzene dimer calculated energy and the extrapolation based on VMC-DMC linear relation. a, The energy of the T-shaped benzene dimer with a center-to-center distance of 4.95 Å as well as the CCSD(T) results as baselines. The experimental range is from Grover et al. [46]. The fitted distribution of binding energy from VMC, DMC and extrapolation results, where the solid lines inside each violin represent the actual data points. The energy for two separated benzene molecules is calculated using a benzene dimer of 4.95 Å (0.70), benzene dimer 10 Å (0.61), benzene (0.47), cyclobutadiene transition (0.53), cyclobutadiene equilibrium (0.64) and FermiNet VMC and DMC change along the optimization process. b, The fitted distribution of binding energy from VMC, DMC and extrapolation results, where the solid lines inside each violin represent the actual data points. c, Linear fitting of $E_{	ext{VMC}} - E_{	ext{DMC}}$ with respect to $E_{	ext{DMC}} - E_{\text{final}}$ on various molecular systems, where $E_{\text{final}}$ represents the DMC energy at final VMC training step. The number within the parenthesis in the legend is the fitted slope for each system. d, Linear fitting of $E_{	ext{VMC}} - E_{	ext{DMC}}$ with respect to $E_{\text{DMC}} - E_{\text{exact}}$ on atoms, where the energy data are the same as Fig. 2e. Note that the data in d is fundamentally different from c in the sense that the linearity in c is about different training steps for the same system while in d the linearity is measured across different systems.

the DMC energy at the final VMC training step, namely a constant for one training process. Such relation is shown in Fig. 5c. Based on this empirical linear relation, we propose an extrapolation scheme

$$E_{\text{DMC}}^{(k)} - E_{\text{ex}} = w \cdot (E_{\text{VMC}}^{(k)} - E_{\text{DMC}}^{(k)}) + b$$  (1)

where $E_{\text{ex}}$ is the extrapolated energy, and $w$ and $b$ are two parameters to be determined. Here slope $w$ can be fitted using $E_{\text{VMC}}^{(k)}$ and $E_{\text{DMC}}^{(k)}$ along the training process, but the intercept $b$ cannot be inferred from those data. Therefore it is difficult to use this scheme to extrapolate absolute energy unless we have extra information on intercept $b$. On the other hand, when calculating relative energy, we may simply assume the intercept $b$ between different configurations are the same so that it can be canceled out in the calculation. Namely for relative energy, we have

$$\Delta E_{\text{ex}} = (1 + w) \cdot \Delta E_{\text{DMC}} - w \cdot \Delta E_{\text{VMC}}$$  (2)

Note that the calculation of relative energy is especially troublesome for neural-network based QMC methods, due to the strong dependence on the number of training steps and the long converging period. See Supplementary Figure 8b for how binding energies calculated with FermiNet VMC and DMC change along the optimization process. With our scheme, the binding energy results calculated from different VMC training steps would be the same, modulo the fitting error of the linear relation, which means we can circumvent the dependence of the binding energy result on the number of training steps. In practice, the extrapolated binding energies form a well concentrated distribution, and doing an extra average using different VMC training steps can eliminate the linear fitting error and provide an accurate estimate. Moreover, it also suggests that we can calculate the extrapolated binding energy with data collected in the early phase of the training process, avoiding the long converging period.
of VMC optimization.

Applying this scheme to binding energy calculation of a benzene dimer, the result is significantly improved and the distribution fitted from energy difference of different VMC training steps is concentrated around the experimental range, as shown in Fig. 5b. The estimate of extrapolated binding energy by averaging the energy difference is 3.60 mHa, within the experimental range. See Supplementary Note 13 for more extrapolation-related details for benzene dimer.

We have discussed the relation of VMC and DMC energy for elements on the second and third rows in Section II.B. For each atom, we have a reference energy data $E_{\text{exact}}$ to be compared with converged VMC energy ($E_{\text{VMC}}$) and DMC energy ($E_{\text{DMC}}$). As shown in Fig. 2e, both $E_{\text{DMC}} - E_{\text{exact}}$ and $E_{\text{VMC}} - E_{\text{exact}}$ grow linearly as the atomic number increases, though the slope changes when switching from the second row elements to the third row. However, if we instead compare

$$E_{\text{DMC}} - E_{\text{exact}} \text{ v.s. } E_{\text{VMC}} - E_{\text{DMC}},$$

then we have a single linear relation across all elements on both second and third rows, as shown in Fig. 5d.

Interestingly, the slope of fitted lines in both Fig. 5c and 5d are all quite close. We will leave further study on those two types of linearity as future work.

### III. DISCUSSION

FermiNet-DMC is able to achieve accurate *ab initio* calculations for various systems, obtaining ground state of 16 atoms, $N_2$ along the bonding curve, 2 cyclobutadiene configurations, 10 hydrogen bonded water dimers, benzene monomer and dimer. These systems include bond breaking structures where strong static correlation exists and weakly bonded dimers where dynamic correlation dominates, and FermiNet-DMC performs consistently well. FermiNet-DMC leverages the expressive power of neural network to provide well-behaved trial wavefunctions. Neural network based VMC has claimed success in small systems when the network can be sufficiently trained. However, it is not able to provide satisfactory ground state wavefunction and energy when the expressiveness of the neural network is limited. Compared to VMC, the combination of neural network with DMC provides a powerful solution, in the sense that it can achieve more accurate result with simpler network and better efficiency. The improvement of FermiNet-DMC in efficiency can be up to 1 or 2 orders of magnitude in the large systems tested in order to reach the same accuracy level as FermiNet-VMC, which can become increasingly more important when dealing with even larger molecules.

There is an interesting linear relation between VMC and DMC energy observed during the training process as well as across different systems. We develop an extrapolation scheme accordingly, which greatly improves the accuracy of relative energy calculation as shown in the benzene dimer case and overcome the issue that the relative energy calculation greatly depends on the different training steps in the QMC process. We also design a divergence measuring the difference between nodal surfaces of two wavefunctions, which correlates well with the corresponding DMC energies in numerical experiments. Namely the proposed divergence successfully captures the essence of nodal surface differences.

It is worth pointing out that a similar idea to this work was proposed in a preprint by Wilson et al., where they have performed preliminary tests on the second row elements [37]. However, only minor improvements in accuracy were observed accompanied by an increased cost of DMC, since the FermiNet used there was powerful enough to achieve high accuracy for the tested small systems and leave little room for further improvement. By comparison, our approach, being more sophisticated and efficient, achieves significant accuracy boost when dealing with more challenging molecular systems, which FermiNet alone cannot handle well. We have also shown that even for small systems, FermiNet-DMC should still be preferred for the fact that it can achieve comparable or even better accuracy with a smaller network and much less computation resources compared with FermiNet-VMC. Our work, therefore, eliminates the negative concerns of going from VMC to DMC with neural network wavefunction ansatz. Moreover, the DMC method can be further integrated with other powerful molecular neural networks [22, 25], periodic neural network for solids [51], neural networks with effective core potential [52], which has the potential to catalyze a paradigm shift in the application of stochastic electronic structure methods.

### IV. METHODS

#### A. Basic theory

To study a many-body system from first principles, we always consider solving the well-known Schrödinger equation for electrons and nuclei. When we work in the Born-Oppenheimer approximation [53], and further consider a fixed set of nuclear positions, the problem is simplified to the solution of the ground state many-electron wavefunction.

\[
\hat{H}\psi(x_1, \cdots, x_n) = E\psi(x_1, \cdots, x_n),
\]

\[
\hat{H} = -\frac{1}{2} \sum_i \nabla_i^2 - \sum_i \sum_j \frac{Z_i}{|\mathbf{r}_i - \mathbf{r}_j|} + \sum_{i<j} \frac{1}{|\mathbf{r}_i - \mathbf{r}_j|} + \sum_i Z_i Z_j,
\]

where $x_i = (r_i, \sigma_i)$ denotes the spatial and spin coordinates of electron $i$, and $\mathbf{r}_i, Z_i$ respectively denote the spatial coordinates and the charge of nucleus $i$. The wavefunction of electrons obeys Fermi-Dirac statistics.
thus should be antisymmetric with respect to the interchange of both the spatial coordinates and the spins of any two electrons, namely the following equality of wavefunction should hold: $\psi(\cdots, x_i, \cdots, x_j, \cdots) = -\psi(\cdots, x_j, \cdots, x_i, \cdots)$.

Unlike most methods that use variational principle to approach the ground state wavefunction, DMC is a stochastic projection method. A given antisymmetric wavefunction $\psi_T$ can always be represented as a linear combination of a set of eigenfunctions $\psi_k$ of the corresponding Hamiltonian operator,

$$\psi_T(x_1, \cdots, x_n) = \sum_{k=0}^{\infty} c_k \psi_k(x_1, \cdots, x_n), \quad (4)$$

$$\hat{H} \psi_k(x_1, \cdots, x_n) = E_k \psi_k(x_1, \cdots, x_n),$$

When an imaginary time evolution operator acts on $\psi_T$,

$$e^{-\tau(\hat{H} - E_T)} \psi_T = \sum_{k=0}^{\infty} c_k e^{-\tau(E_k - E_T)} \psi_k, \quad (5)$$

where $E_T$ is the trial energy as an offset, there will be a decay coefficient added to each expansion term, and the decay rate is proportional to state energy $E_k$. After a long enough imaginary time evolution, $\psi_T$ can reach the ground state $\psi_0$, whereas contributions from all other eigenfunctions vanish. If we define a time-dependent wavefunction and look at the imaginary-time Schrödinger equation:

$$\psi(x_1, \cdots, x_n, \tau) = e^{-\tau(\hat{H} - E_T)} \psi_T(x_1, \cdots, x_n),$$

$$-\partial_\tau \psi(x_1, \cdots, x_n, \tau) = (\hat{H} - E_T) \psi(x_1, \cdots, x_n, \tau). \quad (6)$$

Without the potential energy terms, it resembles a standard diffusion equation,

$$\partial_\tau \psi(x_1, \cdots, x_n, \tau) = \frac{1}{2} \sum_i \nabla_i^2 \psi(x_1, \cdots, x_n, \tau). \quad (7)$$

The diffusion equation defines the master equation of stochastic processes, hence we can solve the diffusion equation of wavefunction by simulating the stochastic processes [54]. With potential terms, additional processes are required to bind the diffusion equation in simulation (see, e.g., Refs. [16, 55, 56] for more details).

B. Trial wavefunction

In this work we use FermiNet neural network ansatz as our trial wavefunction. Due to the huge number of parameters, it is challenging to converge the training process of FermiNet unless the system is small enough. After many tests, we identified a common training pattern of FermiNet, which consists of two stages: a relatively short sharp-adjustment stage and a lengthy fine-tuning one. We propose to use the FermiNet wavefunction right after the sharp-adjustment stage as the trial wavefunction in DMC, which maximizes the efficiency of the entire simulation protocol. In this way we can also achieve more accurate results than a better converged FermiNet model after the lengthy fine-tuning stage. Comparing to the gain, the cost of performing DMC on the long-trained FermiNet is rather minor in most of the systems tested.

C. DMC implementation

We have developed a GPU-friendly DMC software in JAX [57], which can be seamlessly integrated with FermiNet [27], developed in the same programming framework. Our DMC software can also be integrated with other trial wavefunctions implemented in JAX and it has been open sourced in order to accelerate further combination of QMC methods with neural networks. See Algorithm 1 for a brief workflow of one DMC iteration, beyond which various of modifications are implemented, including those proposed by Umrigar et al. to reduce time-step error [55] and by Zen et al. to keep size consistency [58].

Random walkers’ branching and merging change the total number of walkers, which cause efficiency issue for JAX program and is also not friendly to distributed computing especially when load balancing is involved. We devised a new branching-merging strategy to overcome these issues. Whenever we need to branch certain random walker due to its overly large weight, we also merge two walkers on the same computing node with the smallest weight. No merging is executed if no branching happens. In this way, we keep the number of walkers on each computing node unchanged. We did thorough numerical verification of this strategy and found that the introduced bias is negligible.

The most time-consuming module in our DMC implementation is to calculate the local energy. In our optimized program, the computational cost for each local energy estimation is almost same as a VMC inference step of the original FermiNet. Therefore, the total cost depends solely on the number of iterations performed in DMC and VMC.

D. Energy calculation

For FermiNet-VMC, we always perform a separate inference simulation for energy estimate, where we fix all the parameters of FermiNet after training and do a number of Markov Chain Monte Carlo (MCMC) steps to sample batches of random walkers accordingly. We calculate the average local energy for each batch, and use reblock analysis to determine the mean value of the set of averaged energy as well as the standard deviation. For FermiNet-DMC, we use the mixed estimator of energy [55] and treat the first 10% of MC steps as the equilibrating phase and only use the steps afterwards for energy production. See Supplementary Table 6-12 for the hyperparameters of all our calculations. We also use reblock analysis to determine the mean of the averaged
Algorithm 1: Simplified Diffusion Monte Carlo algorithm pseudocode.

Data: number of walkers $M$, walkers’ positions $r$, walkers’ weight $\omega$, value and sign of the trial wavefunction $\Psi_T(r)$, drift velocity $v_D(r)$, local energy $E_L(r)$.

Output: accept ratio of trial moves $p_{move}$, trial energy $E_T$, mixed estimated energy $E_D$.

1. for $N_{max}$ do
   2. for each walker do
      3. \[ \xi \sim N(0, \tau) \]
      4. \[ r' = r + \tau v_D(r) + \xi \]
      5. compute $\Psi_T(r'), \text{sign}(\Psi_T(r'))$, $v_D(r')$
      6. \[ G(r \rightarrow r') = \exp \left[ -(r' - r - \tau v_D(r))^2 / 2\tau \right] \]
      7. \[ G(r' \rightarrow r) = \exp \left[ -(r - r' - \tau v_D(r'))^2 / 2\tau \right] \]
      8. \[ p = \min \left\{ 1, \frac{G(r' \rightarrow r)\psi_T(r')}{G(r \rightarrow r')\psi_T(r)} \right\} \]
      9. if $\text{sign}(\Psi_T(r')) \neq \text{sign}(\Psi_T(r))$ then
         10. $p = 0$
      end
   11. end
   12. $\alpha \sim U(0, 1)$
   13. if $p \geq \alpha$ then
      14. // Diffuse
      15. $w \leftarrow w \times \exp[-\tau(E_L(r) + E_L(r') - 2E_T)/2]$
      16. $r \leftarrow r'$
      17. $\Psi_T(r) \leftarrow \Psi_T(r')$
      18. $\text{sign}(\Psi_T(r)) \leftarrow \text{sign}(\Psi_T(r'))$
      19. $v_D(r) \leftarrow v_D(r')$
      20. $E_L(r) \leftarrow E_L(r')$
   end
   21. end
   22. compute $p_{move}$
   23. $E_T = E_T - \log \sum_w w$
   24. $E_D = \frac{\sum_w wE_L(r)}{\sum_w w}$
26. end

energy and its standard error. In our plots, error bars represent one standard error for energy estimates, unless otherwise specified.

Note that walkers in DMC are more auto-correlated than the ones in VMC inference phase especially when the time step used in DMC is set to be small to avoid bias. Therefore more batches of random walkers are needed to reduce the statistical error to a given level in DMC than in VMC. However in practice, we found that the number of the required extra batches of walkers in DMC is usually much fewer than the number of steps in VMC training phase for full convergence.

E. Nodal structure and wavefunction visualization

The three dimensional cuts of the full 11D nodal structure of Be in Fig. 2b-d is plotted according to the rules of Bressanini et al. [30]. The four electrons’ spherical coordinates are respectively

$$
\begin{align*}
&\{ r_1 \in [0.1, 2.1] \text{ a.u.}, \phi_1 = 0, \theta_1 \in [-1, 1], \\
&\{ r_2 = 1.1 \text{ a.u.}, \phi_2 = 0, \theta_2 = \pi/2, \\
&\{ r_3 \in [0.1, 2.1] \text{ a.u.}, \phi_3 = \pi/2, \theta_3 = \pi/2, \\
&\{ r_4 = 1.1 \text{ a.u.}, \phi_4 = 3\pi/2, \theta_4 = \pi/2,
\end{align*}
$$

fixing all the degrees of freedom except $r_1$, $\theta_1$ and $r_3$.

The green surfaces in the plots show the nodal surfaces, i.e. the places where the value of wavefunction is zero.

To visualize the nodal surface of benzene, we calculated the wavefunction value on 2-dimensional slices of the 126-dimensional space. We first fixed a 126-dimensional electron configuration at the representative position of benzene electronic structure from Liu et al. [42], and perturb it slightly for the visualization purpose. To construct one slice of the 126-dimensional space, we move a single spin-up electron in a 2-dimensional square with all other 41 electrons fixed. Then we apply FermiNet to points on each slice and display the log-scaled magnitude of the evaluated wavefunction value, where the points with small value stand for the nodes on each slice. Since the FermiNet output is unnormalized, diagrams for different FermiNet may have drastically different range of displayed value.

F. Divergence measuring nodal surface difference

We define a divergence measuring the difference between two sets $S_1$ and $S_2$ in any metric space as follows

$$
D(S_1, S_2) = E_Y \sim P_1 d(Y, S_2) \approx \sum_i d(Y_i, S_2) \quad (8)
$$

where $P_1$ is a probability measure on $S_1$, and $\{Y_i\}_{i=1}^K$ are sampled from $P_1$. The distance $d(Y, S)$ between a single point $Y$ and a set $S$ is defined as the smallest distance between $Y$ and any point in $S$, namely

$$
 d(Y, S) = \min_{Z \in S} d(Y, Z) \quad (9)
$$

For a nodal surface $S$ corresponding to an unnormalized wavefunction $\Psi$, we would like to define a measure on $S$ such that a small area on $S$ is assigned larger weight if its neighbourhood has larger $\Psi^2$ value, namely larger probability to be visited by walkers in DMC. Therefore, we consider a neighbourhood

$$
S_\epsilon = \{ x | d(x, S) < \epsilon \} \quad (10)
$$

around $S$ and a mapping

$$
\phi: S_\epsilon \rightarrow S,
$$

then "push forward" the probability density $m_{\Psi^2}$ (corresponding to $\Psi^2$) from $S_\epsilon$ to $S$ via $\phi$, namely

$$
\phi \circ m_{\Psi^2}(n) := m_{\Psi^2}(\phi^{-1}(s)) = \frac{\int_{\phi^{-1}(s)} \Psi^2}{\int_{S_\epsilon} \Psi^2}, \ \forall \text{ set } s \subset S
$$

(11)
Intuitively, for any point $y$ in $S$, we may simply choose $\phi(y)$ to be the point on $N$ that is closest to $y$.

However, it’s quite difficult to determine both $S$ and $\phi$ mentioned above algorithmically, and thus, in practice, we use some approximate alternatives that are much easier to compute. See Supplementary Note 15 for the algorithmic detail.

**Data Availability**

All data supporting the findings of this study are provided in Supplementary Information.

**Code Availability**

We have released our DMC software at [https://github.com/bytedance/jaqmc](https://github.com/bytedance/jaqmc).

**REFERENCES**

[1] J. A. Pople, Nobel lecture: Quantum chemical models, Rev. Mod. Phys. **71**, 1267 (1999).
[2] W. Kohn, Nobel lecture: Electronic structure of matter—wave functions and density functionals, Rev. Mod. Phys. **71**, 1253 (1999).
[3] T. Helgaker et al., Recent advances in wave function-based methods of molecular-property calculations, Chemical Reviews **112**, 543 (2012).
[4] Y. Cao et al., Quantum chemistry in the age of quantum computing, Chemical Reviews **119**, 10856 (2019).
[5] J. Kirkpatrick et al., Pushing the frontiers of density functionals by solving the fractional electron problem, Science **374**, 1385 (2021).
[6] J. G. Brandenburg, A. Zen, D. Alfè, and A. Michaelides, Interaction between water and carbon nanostructures: How good are current density functional approximations?, The Journal of Chemical Physics **151**, 164702 (2019).
[7] Y. S. Al-Hamdani et al., Interactions between large molecules pose a puzzle for reference quantum mechanical methods, Nature Communications **12**, 1 (2021).
[8] J. J. Eriksen et al., The Ground State Electronic Energy of Benzene, The Journal of Physical Chemistry Letters **11**, 8922 (2020).
[9] K. T. Williams et al., Direct Comparison of Many-Body Methods for Realistic Electronic Hamiltonians, Physical Review X **10**, 011041 (2020).
[10] G. H. Booth, A. J. Thom, and A. Alavi, Fermion monte carlo without fixed nodes: A game of life, death, and annihilation in slater determinant space, The Journal of chemical physics **131**, 054106 (2009).
[11] C. J. Umrigar, J. Toulouse, C. Filippi, S. Sorella, and R. G. Hennig, Alleviation of the fermion-sign problem by optimization of many-body wave functions, Phys. Rev. Lett. **98**, 110201 (2007).
[12] P. R. C. Kent et al., QMCPACK: Advances in the development, efficiency, and application of auxiliary field and real-space variational and diffusion quantum Monte Carlo, The Journal of Chemical Physics **152**, 174105 (2020).
[13] R. J. Needs, M. D. Towler, N. D. Drummond, P. López Rios, and J. R. Trail, Variational and diffusion quantum Monte Carlo calculations with the CASINO code, The Journal of Chemical Physics **152**, 154106 (2020).
[14] J. B. Anderson, A random-walk simulation of the schrödinger equation: $H^+_t$, The Journal of Chemical Physics **63**, 1499 (1975).
[15] J. B. Anderson, Quantum chemistry by random walk. $H^2 P$, $H^+_4 D_{sh}$ $A_1$, $H_2 \Sigma_{g}^+$, $H_4 \Sigma_{u}^+$, $Be \Sigma_{g}^+$, The Journal of Chemical Physics **65**, 4121 (1976).
[16] W. M. C. Foulkes, L. Mitas, R. J. Needs, and G. Rajagopal, Quantum Monte Carlo simulations of solids, Reviews of Modern Physics **73**, 33 (2001).
[17] M. Bajdich, L. Mitas, L. K. Wagner, and K. E. Schmidt, Pfaffian pairing and backflow wavefunctions for electronic structure quantum monte carlo methods, Phys. Rev. B **77**, 115112 (2008).
[18] P. López Rios, A. Ma, N. D. Drummond, M. D. Towler, and R. J. Needs, Inhomogeneous backflow transformations in quantum monte carlo calculations, Phys. Rev. E **74**, 066701 (2006).
[19] J. Han, L. Zhang, and W. E, Solving many-electron schrödinger equation using deep neural networks, Journal of Computational Physics **399**, 108929 (2019).
[20] D. Pfau, J. Spencer, A. de G. Matthews, and W. Foulkes, Ab-initio solution of the many-electron schrödinger equation with deep neural networks, Phys. Rev. Research **2**, 033429 (2020).
[21] J. S. Spencer, D. Pfau, A. Botev, and W. M. C. Foulkes, Better, Faster Fermionic Neural Networks, arXiv:2011.07125 [physics] (2020).
[22] J. Hermann, Z. Schätzle, and F. Noé, Deep-neural-network solution of the electronic Schrödinger equation, Nature Chemistry **12**, 891 (2020).
[23] M. Entwistle, Z. Schätzle, P. A. Erdman, J. Hermann, and F. Noé, Electronic excited states in deep variational monte carlo, arXiv preprint arXiv:2203.09472 (2022).
[24] J. Lin, G. Goldshlager, and L. Lin, Explicitly antisymmetrized neural network layers for variational monte carlo simulation (2021), arXiv:2112.03491 [physics.compph].
[25] M. Scherbelina, R. Reisenofer, L. Gerard, P. Marque-tand, and P. Grohs, Solving the electronic schrödinger equation for multiple nuclear geometries with weight-sharing deep neural networks, Nature Computational Science **10.1038/s43588-022-00228-x** (2022).
[26] N. Gao and S. Günemann, Ab-Initio Potential Energy Surfaces by Pairing GNNs with Neural Wave Functions, arXiv preprint arXiv:2110.05064 (2021).
[27] D. P. James, S. Spencer and F. Contributors, FermiNet (2020).
[28] M. Caffarel, Quantum monte carlo for chemistry @ toulouse, [http://qmchem.ups-tlse.fr/index.php/Quantum_Monte_Carlo_for_Chemistry_@_Toulouse/](http://qmchem.ups-tlse.fr/index.php/Quantum_Monte_Carlo_for_Chemistry_@_Toulouse/)
[29] S. J. Chakravorty, S. R. Gwaltney, E. R. Davidson, F. A. Parpia, and C. F. p Fischer, Ground-state correlation energies for atomic ions with 3 to 18 electrons, Physical Review X **10**, 011041 (2020).
[30] D. Bressanini, Implications of the two nodal domains conjecture for ground state fermionic wave functions, Phys-
ical Review B 86, 115120 (2012).
[31] R. J. Le Roy, Y. Huang, and C. Jary, An accurate analytic potential function for ground-state n2 from a direct-potential-fit analysis of spectroscopic data, The Journal of chemical physics 125, 164310 (2006).
[32] R. J. Gdanitz, Accurately solving the electronic schrödinger equation of atoms and molecules using explicitly correlated (r12-) mr-ci: the ground state potential energy curve of n2, Chemical physics letters 283, 253 (1998).
[33] D. I. Lyakh, M. Musial, V. F. Lotrich, and R. J. Bartlett, Multireference nature of chemistry: The coupled-cluster view, Chemical reviews 112, 182 (2012).
[34] B. J. Smith, D. J. Swanton, J. A. Pople, H. F. Schaefer III, and L. Radom, Transition structures for the interchange of hydrogen atoms within the water dimer, The Journal of chemical physics 92, 1240 (1990).
[35] G. S. Tschumper et al., Anchoring the water dimer potential energy surface with explicitly correlated computations and focal point analyses, The Journal of chemical physics 116, 690 (2002).
[36] M. Gillan, F. Manby, M. Towler, and D. Alfè, Assessing the accuracy of quantum mono Carlo and density functional theory for energetics of small water clusters, The Journal of chemical physics 136, 244105 (2012).
[37] 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 mono Carlo, arXiv preprint arXiv:2103.12570 (2021).
[38] M. J. Gillan, D. Alfè, and A. Michaelides, Perspective: How good is DFT for water?, The Journal of Chemical Physics 144, 130901 (2016).
[39] D. G. A. Smith et al., Psi4 1.4: Open-source software for high-throughput quantum chemistry, The Journal of Chemical Physics 152, 184108 (2020), https://doi.org/10.1063/5.0006002.
[40] R. D. J. III, Nist computational chemistry comparison and benchmark database, nist standard reference database number 101 (2021).
[41] J. J. Eriksen et al., The ground state electronic energy of benzene, The Journal of Physical Chemistry Letters 11, 8922 (2020).
[42] Y. Liu, P. Kilby, T. J. Frankcombe, and T. W. Schmidt, The electronic structure of benzene from a tiling of the correlated 126-dimensional wavefunction, Nature Communications 11, 1210 (2020).
[43] M. Pitonak et al., Benzene dimer: high-level wave function and density functional theory calculations, Journal of chemical theory and computation 4, 1829 (2008).
[44] Z. Schätzle, J. Hermann, and F. Noé, Convergence to the fixed-node limit in deep variational mono Carlo, The Journal of Chemical Physics 154, 124108 (2021).
[45] S. Azadi and R. Cohen, Chemical accuracy from quantum mono Carlo for the benzene dimer, The Journal of chemical physics 143, 104301 (2015).
[46] J. R. Grover, E. A. Walters, and E. T. Hui, Dissociation energies of the benzene dimer and dimer cation, The Journal of Physical Chemistry 91, 3233 (1987).
[47] Y. S. Al-Hamdani et al., Interactions between large molecules pose a puzzle for reference quantum mechanical methods, Nature Communications 12, 3927 (2021).
[48] S. Tsuzuki, K. Honda, T. Uchimaru, M. Mikami, and K. Tanabe, Origin of attraction and directionality of the π/π interaction: Model chemistry calculations of benzene dimer interaction, Journal of the American Chemical Society 124, 104 (2002), pMID: 11772067, https://doi.org/10.1021/ja0105212.
[49] M. O. Sinnokrot, E. F. Valeev, and C. D. Sherrill, Estimates of the ab initio limit for π-π interactions: The benzene dimer, Journal of the American Chemical Society 124, 10887 (2002), pMID: 12207544, https://doi.org/10.1021/ja025896h.
[50] A. Zen, J. G. Brandenburg, A. Michaelides, and D. Alfè, A new scheme for fixed node diffusion quantum mono Carlo with pseudopotentials: Improving reproducibility and reducing the trial-wave-function bias, The Journal of Chemical Physics 151, 134105 (2019).
[51] X. Li, Z. Li, and J. Chen, Ab initio calculation of real solids via neural network ansatz, Nature Communications 13, 7895 (2022).
[52] X. Li, C. Fan, W. Ren, and J. Chen, Fermionic neural network with effective core potential, Phys. Rev. Research 4, 013021 (2022).
[53] M. Born and R. Oppenheimer, Zur quantentheorie der molekül, Annalen der physik 389, 457 (1927).
[54] S. Karlin and H. E. Taylor, A second course in stochastic processes (Elsevier, 1981).
[55] C. Umrigar, M. Nightingale, and K. Runge, A diffusion mono Carlo algorithm with very small time-step errors, The Journal of chemical physics 99, 2865 (1993).
[56] P. J. Reynolds, J. Tsibouchnik, and H. Gould, Diffusion quantum mono Carlo, Computers in Physics 4, 662 (1990).
[57] J. Bradbury et al., JAX: composable transformations of Python+NumPy programs (2018).
[58] A. Zen, S. Sorella, M. J. Gillan, A. Michaelides, and D. Alfè, Boosting the accuracy and speed of quantum mono Carlo: Size consistency and time step, Physical Review B 93, 241118 (2016).

Acknowledgments

This work is directed and supported by Hang Li and ByteDance Research. We thank Yubing Qian for providing preliminary settings for water dimer simulations. We thank Weinan E, Xiang Li, Kai Zheng, Ke Liao and Yu Liu for fruitful discussions. We thank Mike Entwistle and James Spencer for sharing data and results. We thank Michel Caffarel for allowing us to use an adapted version of the figure on his website. We thank Shaochen Shi, Xiaoying Jia, Xin Liu and Chenlin Chai for engineering improvement on our DMC software. We thank the rest of ByteDance Research team for inspiration and encouragement. J.C. is supported by the National Key R&D Program of China under Grant No. 2021YFA1400500, the National Natural Science Foundation of China under Grant No. 92165101 and No. 11974024, and the Strategic Priority Research Program of Chinese Academy of Sciences under Grant No. XDB33000000.

Author contributions

W.R. and J.C. conceived the study; W.R. implemented the main code with important contributions from W.F.; W.R. and W.F. performed simulations, data analyses, and figure designing. X.W. performed CCSD(T) related calculation. J.C. supervised the project. W.R., W.F., X.W., and J.C. wrote the paper.
Competing interests

The authors declare no competing interests.
Supplementary Information: Towards the ground state of molecules via diffusion Monte Carlo on neural networks

Weiluo Ren\textsuperscript{1}, Weizhong Fu\textsuperscript{1,2}, Xiaojie Wu\textsuperscript{1}, and Ji Chen\textsuperscript{2,3}

\textsuperscript{1}ByteDance Research, Zhonghang Plaza, No. 43, North 3rd Ring West Road, Haidian District, Beijing, People’s Republic of China
\textsuperscript{2}School of Physics, Peking University, Beijing 100871, People’s Republic of China
\textsuperscript{3}Interdisciplinary Institute of Light-Element Quantum Materials, Frontiers Science Center for Nano-Optoelectronics, Peking University, Beijing 100871, People’s Republic of China

Supplementary Note 1. Additional software detail

In the method section, we have briefly described the DMC software developed for this work, including key methodological details. Here we add a few more points, and a complete description of our software will be published elsewhere.

Our implementation of DMC is scaling-out friendly. We have tested our software on a GPU cluster with 16 computing nodes and 128 Nvidia V100 GPU cards, and obtained near-linear speed-up. Our software saves checkpoint for DMC periodically and uploads to a remote storage cluster so that it can be resilient to occasional preemption due to cluster scheduling and device failure, and it is able to continue the process with the most recent checkpoint downloaded when a rescheduling is triggered.

In order to support multi-node computation efficiently, we have to minimize the inter-nodes communication, listed below:

- We calculate the total energy with local energy and weights for all the walkers across all the nodes.
- We also calculate several metrics with across-node aggregation, such as acceptance rate, number of outliers and etc.
- We synchronize the action of uploading and downloading checkpoints.

We have not yet supported across-node load balancing in terms of walkers, which could cause large communication overhead. As a matter of fact, we have devised a branching-merging algorithm which fixes the walker size per computing node so that we do not need to apply across-node walker load balancing at all. Every time when we do branching at the presence of a large-weight walker, we simultaneously find two walkers on the same node but with the smallest weight and merge them. Note that if the trial wavefunction has good quality and we have a decent number of walkers on each node, then such an action does not introduce much bias, confirmed in our calculations and shown in Supplementary Note 3. We only use the fixed-size branching scheme for relatively large systems, such as cyclobutadiene, benzene molecule and benzene dimers, in which cases efficiency is crucial due to the computation power limit and the fixed-node error dominates the extra bias introduced by this branching scheme.

In order to revert the effect of changing trial energy, the energy produced from each DMC step should be corrected accordingly before averaged. However, as pointed in \cite{1}, when the batch size is large enough, the effect of such correction is negligible, which is consistent with our experience. Furthermore, it is easier to do reblocking analysis \cite{2} to assess statistical error without such correction. Therefore, for all the results reported here, we do not include such energy corrections in our computation.

We have also implemented the electron-by-electron scheme suggested by CASINO \cite{3}, where each walker moves electrons one by one rather than all at once. Our tests show that the electron-by-electron scheme leads to a significant increase in the acceptance ratio of walkers with large time-steps, but we do not see much improvement on the time-step error or auto-correlation period. Besides, our electron-by-electron implementation sequentially goes over all the electrons in one configuration, leading to a decline in computing efficiency when dealing with large systems. So this option is also switched off by default.

Supplementary Note 2. Finite time-step error

We have implemented the ZSGMA approach proposed by Zen et al. to reduce the finite time-step error \cite{4}. As tested in various systems in literature our default setting $10^{-3}$ is usually small enough to reach convergence. Here we show a test on the cyclobutadiene with equilibrium structure as an example. The results are displayed in Supplementary Fig. 1, from which we can conclude that the finite time-step error for our default setting $10^{-3}$ is well below 1 mHa, comparable to the statistical error. Based on this observation as well as the efficiency concern, we use time-step $10^{-3}$ by default and use it in all of our FermiNet-DMC calculations.

To obtain comparable statistical error of the reported energy, we use different number of DMC steps for each time-step to compensate different level of auto-correlation in the energy time series. For time-step $10^{-3}$, we run DMC for $10^5$ steps. For large time-steps like 0.01 or 0.03, we only run DMC for $10^4$ steps while for small time-step like $10^{-4}$, we run DMC for $10^6$ steps.
**Supplementary Figure 1** | Time-step dependence of the total energy of cyclobutadiene with equilibrium structure. The pink curve is generated by doing a quadratic fitting on the collected DMC energy data points.

**Supplementary Note 3. DMC options**

To investigate the effect of different hyperparameters of our DMC software, especially the bias introduced in different settings, we compare the DMC energy calculated with the settings listed in Supplementary Table 1, using cyclobutadiene with equilibrium structure as an example. The trial wavefunction is a FermiNet trained with $10^5$ steps. See Note 9 for more training details. The result is displayed in Supplementary Fig. 2. Taking the statistical error into account, all settings agree with each other well except the most inaccurate one with time-step $10^{-2}$.

**Supplementary Table 1** | We test our DMC software with different hyperparameters, listed as follows. “time step” is the DMC time-step. “equilibrium phase length” is the length of phase in the beginning of the whole DMC process during which we do not collect statistics for report purpose. “total number of steps” is the number of steps in the whole DMC process. “energy window size” is the length of the rolling window for trial-energy update. “fix size” indicates whether we fix the number of DMC walkers on each computing node or not. “energy cutoff” is the parameter $\alpha$ in the cutoff scheme proposed in [4] and indicates the level of energy clipping.

| Identifier         | time step | equilibrium phase length | total number of steps | energy window size | fix size | energy cutoff |
|--------------------|-----------|----------------------------|-----------------------|-------------------|----------|--------------|
| time step $10^{-2}$| $10^{-2}$  | $2 \times 10^3$             | $10^4$                | $10^3$            | True     | 2            |
| time step $10^{-3}$ (default) | $10^{-3}$  | $2 \times 10^4$             | $10^5$                | $10^4$            | True     | 2            |
| time step $10^{-4}$| $10^{-4}$  | $2 \times 10^5$             | $10^6$                | $10^5$            | True     | 2            |
| window $5 \times 10^4$ | $10^{-3}$  | $10^5$                      | $2 \times 10^5$       | $5 \times 10^4$   | True     | 2            |
| vary size          | $10^{-3}$  | $2 \times 10^4$             | $10^5$                | $10^4$            | False    | 2            |
| energy cutoff 4    | $10^{-3}$  | $2 \times 10^4$             | $10^5$                | $10^4$            | True     | 4            |
Supplementary Figure 2 | DMC energy and statistical error for different settings listed in Supplementary Table 1. The pink patch indicates the one-standard-error interval of energy in the most accurate setting with time step $10^{-4}$. All the error bars are overlapped with this interval except the most inaccurate setting with time-step $10^{-2}$.

Supplementary Note 4. Efficiency

In this section, we compare the efficiency between FermiNet-VMC [5–7] and FermiNet-DMC. The evaluation of FermiNet’s local energy dominates the whole FermiNet-DMC process in terms of computation time. In other words, for one step, the difference between runtime of FermiNet-DMC and the inference phase of FermiNet-VMC is negligible.

However, DMC usually requires more steps to reduce the statistical error to a given level compared to VMC inference phase, because DMC has to use small time-step to avoid finite time-step error while VMC can use large time-step in Markov chain Monte Carlo (MCMC) to obtain less correlated samples. To cut down the statistical error more effectively in DMC, we can increase the batch size so that each batch contains more independent walkers. In practice, we start FermiNet-DMC procedure from the FermiNet-VMC walkers. To increase the DMC batch size, we duplicate the VMC walkers and then do MCMC for a period of time to get a larger batch of uncorrelated walkers, used as the starting walkers in DMC. Even if we do not enlarge the batch size, we still do a number of MCMC steps using fixed FermiNet before DMC so that the walkers is distributed closer to the square of FermiNet’s wavefunction, which serve better starting point for DMC. Note that doing MCMC with FermiNet is much faster than doing DMC since no local energy calculation is involved. In our experiments, we use larger DMC batch size only when handling benzene dimers, in which case DMC uses batch size 16384 and VMC uses batch size 4096.

Overall FermiNet-DMC is still a much more efficient solution than FermiNet-VMC, because it can drastically cut down the length of the training phase of VMC which is significantly longer than the DMC itself. As a matter of fact, in the calculations that we performed, FermiNet-DMC can reach the same or even better accuracy level with a FermiNet trained with only one tenth or even a smaller fraction of total number of steps compare to the fully converged FermiNet-VMC.

Supplementary Note 5. Details on large systems

When dealing with relatively large systems containing dozens if not hundreds of electrons, chances are that we can not afford a large enough network to well represent the ground state wavefunction. Several issues emerge in this regime when training FermiNet. Firstly, the under-fit FermiNet may hit outliers hurting the convergence process. What’s worse, the training process may hit NAN issue occasionally. We removed the outliers in terms of local energy when calculating the gradient in the FermiNet’s training process. This helps to stabilize the calculation of the average and standard deviation of local energies in a batch, which benefits the gradient clipping process. Without such outlier-removal operation, at the presence of one or more outliers, the gradient clipping is not effective since we are not able to determine the clipping level correctly.

Secondly, the determinant calculation of large-dimension matrix is numerically troublesome. Potentially, there could be overflow and underflow issues especially when using single precision. Besides that, there is usually a dominating determinant among multiple determinants, which makes the value of other determinants effectively zero due to float number rounding. Therefore, we only use a single determinant when dealing with large systems so that the process becomes more efficient and stable. Admittedly, this limits the network expressiveness, and further improvements are left for future work.

Thirdly, the under-fitted decaying behavior of electron density at a distance may cause serious issues, introducing outliers in the calculation of total energy. For instance, there could be walkers occasionally diffusing away from atoms and reaching areas where the
neural network is not well trained, which causes trouble for the calculation of total energy and its gradient. We use isotropic envelope [6] to maintain the exponentially decaying behavior of the wavefunction when the electron is away from all the atoms, which also helps to improve the efficiency of both FermiNet-VMC and FermiNet-DMC.

**Supplementary Note 6. Common Hyperparameters**

All of our calculations follow the same procedure. First we train FermiNet with VMC[5–7], then we use the trained FermiNet as trial wavefunction in DMC. To obtain reliable energy from VMC, we run a separate inference process with the trained FermiNet. There are some hyperparameters shared by all the calculations in this work, as listed in Supplementary Table 2.

**Supplementary Table 2 | Common hyperparameters**

| Hyperparameter               | Value | Hyperparameter               | Value |
|------------------------------|-------|------------------------------|-------|
| VMC optimizer                | KFAC  | Batch size                  | 4096  |
| Precision                    | Float32| Full determinant            | True  |
| DMC time-step                | $10^{-3}$|                            |       |

**Supplementary Note 7. Atoms**

A rather small neural network is used in calculations of single atoms (see Fig. 1e of main text), and all the training processes have well converged. The corresponding hyperparameters are listed in Supplementary Table 3.

**Supplementary Table 3 | Hyperparameters for single atoms calculations in Fig. 2e.**

| Hyperparameter                           | Value | Hyperparameter                           | Value |
|------------------------------------------|-------|------------------------------------------|-------|
| Dimension of one electron layer          | 32    | Dimension of two electron layer          | 4     |
| Number of layers                         | 2     | Number of determinants                   | 1     |
| Envelope type                            | full  | VMC learning rate                        | $5 \times 10^{-4}$ |
| Number of training steps                 | $\geq 5 \times 10^5$| Number of inference steps             | $10^5$ |
| Number of DMC steps                      | $2 \times 10^5$| MCMC steps between each iterations       | 100   |
| Outlier removal                          | false | Fixed-size branching                     | false |

**Supplementary Note 8. Nitrogen Molecule**

Note that our FermiNet-VMC result shown in Fig. 3a of main text is better than the reported FermiNet result in [5]. The main difference is that our calculation uses the full determinant mode, which boosts the accuracy of FermiNet-VMC, as pointed out in [6, 10].

**Supplementary Table 4 | Hyperparameters for nitrogen dissociation curve calculations in Fig. 3a of main text.**

| Hyperparameter                           | Value | Hyperparameter                           | Value |
|------------------------------------------|-------|------------------------------------------|-------|
| Dimension of one electron layer          | 256   | Dimension of two electron layer          | 32    |
| Number of layers                         | 4     | Number of determinants                   | 16    |
| Envelope type                            | full  | VMC learning rate                        | $10^{-4}$ |
| Number of training steps                 | $2 \times 10^5$| Number of inference steps             | $10^5$ |
| Number of DMC steps                      | $2 \times 10^5$| MCMC steps between each iterations       | 10    |
| Outlier removal                          | false | Fixed-size branching                     | false |

**Supplementary Note 9. Cyclobutadiene**

We use the default network structure in the open-source FermiNet repository for cyclobutadiene, and train two neural networks for the equilibrium and transition state configurations separately. The training process was converged after $5 \times 10^5$ steps, as shown in Supplementary Fig. 3. We use FermiNet trained with $10^5$ and $5 \times 10^5$ steps, and run FermiNet-DMC and FermiNet-VMC inference processes accordingly. The results are listed in Supplementary Table 6, where we also include the results from [6] for comparison.
**Supplementary Table 5** | Hyperparameters for cyclobutadiene related calculations

| Hyperparameter                        | Value | Hyperparameter                        | Value |
|---------------------------------------|-------|---------------------------------------|-------|
| Dimension of one electron layer       | 256   | Dimension of two electron layer       | 32    |
| Number of layers                      | 4     | Number of determinants                | 16    |
| Envelope type                         | full  | VMC learning rate                     | $10^{-4}$ |
| Number of training steps              | $5 \times 10^5$ | Number of inference steps           | $10^5$ |
| Number of DMC steps                   | $10^5$ | MCMC steps between each iterations   | 10    |
| Outlier removal                       | false | Fixed-size branching                  | true  |

**Supplementary Table 6** | Cyclobutadiene ground state energy in Hartree. “VMC” and “DMC” indicates our FermiNet-VMC and FermiNet-DMC results respectively. For FermiNet-DMC, we include results corresponding to FermiNet trained with $10^5$ and $5 \times 10^5$ steps. “VMC∗” indicates results from [6].

| Configuration | VMC $10^5$ step | VMC $5 \times 10^5$ step | DMC $10^5$ step | DMC $5 \times 10^5$ step | VMC∗ |
|---------------|----------------|-------------------------|----------------|-------------------------|------|
| Equilibrium   | -154.6577(1)   | -154.6655(1)            | -154.6740(3)   | -154.6770(2)            | -154.6719(1) |
| Transition    | -154.6430(1)   | -154.6505(1)            | -154.6585(3)   | -154.6611(2)            | -154.6554(1) |

**Supplementary Figure 3** | The FermiNet-VMC training curve and FermiNet-DMC energy for cyclobutadiene equilibrium and transition configurations. “VMC” and “DMC” indicates our FermiNet-VMC and FermiNet-DMC results respectively. For FermiNet-DMC, we show results corresponding to FermiNet trained with $10^5$ and $5 \times 10^5$ steps. “VMC∗” indicates results from [6].
Supplementary Note 10. Water Dimer

The 10 structures of water dimer and their point-group symmetries are shown in Supplementary Fig. 4. The energy curves of FermiNet-VMC and corresponding FermiNet-DMC energy are shown in Supplementary Fig. 5, where all the VMC training curves show a clear “elbow” pattern and transition to the slow-converging phase around $10^5$ step. The FermiNet-DMC energy at $10^5$ step is already lower than the corresponding FermiNet-VMC energy at $3 \times 10^5$ step for all structures. Actually, for most structures the VMC energy curve is still not converged at $3 \times 10^5$ step. In principle, convergence in VMC training stage is necessary to investigate the energy difference. All the relative energy results are shown here in Supplementary Fig. 6. In Fig. 3d of main text we see that the mean absolute deviation of the relative energy results of FermiNet-DMC are much better than VMC’s, especially at $10^5$ step when the networks are undertrained. This is reasonable as the nodal surface, which contains the zero order information of wave function, naturally requires fewer training steps to converge. In conclusion, to calculate the relative energy, DMC is a much more efficient and reliable choice compared with VMC.

Supplementary Table 7 | Hyperparameters for water dimer calculations in Fig. 3d of the main text.

| Hyperparameter                  | Value     | Hyperparameter                  | Value     |
|--------------------------------|-----------|--------------------------------|-----------|
| Dimension of one electron layer | 256       | Dimension of two electron layer | 32        |
| Number of layers               | 4         | Number of determinants          | 16        |
| Envelope type                   | full      | VMC learning rate               | $10^{-4}$ |
| Number of training steps       | $3 \times 10^5$ | Number of inference steps      | $10^5$    |
| Number of DMC steps            | $2 \times 10^5$ | MCMC steps between each iterations | 100 |
| Outlier removal                 | false     | Fixed-size branching            | false     |

Supplementary Figure 4 | Geometries of the SP$n$ ($n = 1, 2, ..., 10$) structures of water dimer, with their point-group symmetry labels.

Supplementary Figure 5 | Total energy results of the SP$n$ ($n = 1, 2, ..., 10$) structures of water dimer.
Supplementary Figure 6 | Relative energy results of the $SP_n$ ($n = 2, 3, \ldots, 10$) structures of water dimer to the SP1 structure.

Supplementary Note 11. Benzene

We trained a 3-layer and a 4-layer FermiNet for the benzene molecule. The common hyperparameters for both networks are listed in Supplementary Table 8. The hyperparameters indicating the network quality are described in Supplementary Table 9 and 10, respectively. The energy calculated are listed in Supplementary Table 11.

**Supplementary Table 8** | Common hyperparameters for benzene related calculations

| Hyperparameter                  | Value       | Hyperparameter                  | Value       |
|---------------------------------|-------------|---------------------------------|-------------|
| Number of determinants          | 1           | Envelope type                   | isotropic   |
| Number of VMC inference steps   | $10^5$      | Number of DMC steps             | $2 \times 10^5$ |
| Learning rate                   | 0.1         | MCMC steps between each iterations | 100         |
| Outlier removal                 | True        | Fixed-size branching            | true        |

**Supplementary Table 9** | Hyperparameters for 3-layer FermiNet for the benzene molecule

| Hyperparameter                  | Value       | Hyperparameter                  | Value       |
|---------------------------------|-------------|---------------------------------|-------------|
| Dimension of one electron layer | 128         | Dimension of two electron layer | 8           |
| Number of layers                | 3           | Number of VMC training steps    | $2 \times 10^6$ |

In Supplementary Fig. 7, we provide more visualization of the log scaled magnitude of wavefunction on slices of the electronic configuration space as supplement to the main text Fig. 4. As described in main text Section II F, the slices are generated by moving a single spin-up electron while keeping all others fixed in their representative positions. In Supplementary Fig. 7, the moving electrons are corresponding to the C-H bonds in a benzene. Note that there is no symmetry built in the neural network, while we can still see clear patterns across different subplots in Supplementary Fig. 7, indicating that FermiNet successfully learned the symmetric property of this system. Moreover, the node structure near the light area, namely the one with larger wavefunction magnitude and larger probability to be sampled in the VMC process, shares higher level of similarity. Intuitively, this part of nodes plays more important roles in our fixed-node DMC compared to the rest of nodes which are less likely to be visited in the DMC process, further explaining why the 3-layer and 4-layer FermiNet-DMC results are so close.
Supplementary Table 10 | Hyperparameters for 4-layer FermiNet for the benzeno module

| Hyperparameter                      | Value | Hyperparameter                      | Value |
|-------------------------------------|-------|-------------------------------------|-------|
| Dimension of one electron layer     | 256   | Dimension of two electron layer     | 32    |
| Number of layers                    | 4     | Number of VMC training steps        | $10^6$|

Supplementary Table 11 | Calculated benzene energy with FermiNet based VMC and DMC in Hartree. The 3-layer neural network is trained with $2 \times 10^6$ steps and the 4-layer one is trained with $10^6$ steps. The DMC energy has well converged.

| Network Structure | FermiNet-VMC | FermiNet-DMC |
|-------------------|--------------|-------------|
| 3-layer           | -232.2143(1) | -232.2330(3)|
| 4-layer           | -232.2233(1) | -232.2370(3)|

Supplementary Figure 7 | The 2-dimensional slices of log scaled magnitude of wavefunctions for benzene. The first and third column of figures are for a 4-layer FermiNet and the rest is for a 3-layer FermiNet. Each figure corresponds to a spin-up electron on a C-H bond, whose position is indicated by the lightest area in the figure. The dark pixels correspond to nodes.
Supplementary Note 12. Benzene dimer

We use the same 3-layer neural network for the benzene molecule to handle benzene dimers. The network for a benzene dimer is significantly more difficult to train than the benzene molecule. One of the tricky hyperparameters is the learning rate. Small learning rates lead to extremely slow convergence as well as trapping in local minimum, which was also mentioned in Lin et al. [10], while large learning rates may cause instability or even NaN issues ruining the whole process. In our numerous trials, we found that an initial learning rate of 0.01 with a slightly larger decay rate (5000 as opposed to the default 10000) hits a nice balance. It took millions of training steps to finally obtain a converge energy in FermiNet-DMC. In comparison, FermiNet-VMC has not converged even after four million steps. Our calculated energy for a T-shaped benzene dimer with bond length 4.95 Å is listed in Supplementary Table 12. The DMC energy of dimers are still higher than the doubled monomer DMC energy by 2 mHa, indicating that a 3-layer network is too restricted to handle a benzene dimer, which would lead to a bias when estimating binding energies.

Supplementary Table 12 | The ground state energy in Hartree for a T-shaped benzene dimer with bond length 4.95 Å. VMC and DMC results are based on a FermiNet trained for $4 \times 10^6$ steps.

| Methods       | Energy ($E_h$) |
|---------------|----------------|
| FermiNet-VMC  | -464.4067(3)   |
| FermiNet-DMC  | -464.4640(2)   |

In principle, the binding energy can be accurately determined when the ground state energy of the dimer is calculated using a sufficiently large network, which however is beyond the capability of our available computational resource. In order to reduce the error in estimating the binding energy, we train a separate FermiNet for a configuration with two benzene molecules separated at 10 Å, which we dub as “long-distance”. The training curve is also displayed in Supplementary Fig. 8.a. The calculated binding energy results, taken as the energy difference between the long-distance configuration and the equilibrium configuration, are shown in Fig. 8b. Although FermiNet-DMC improves the convergence and the final results of FermiNet-VMC significantly, it still shows a slight overbinding compared with experimental measured range. From the energy convergence pattern, the DMC energy for the long-distance configuration converges earlier than the one for the equilibrium configuration, resulting in an overly-large binding energy as the energy for equilibrium configuration gets smaller. The reason still lies in the inconsistency of two neural networks in describing two different configurations. To begin with, a long-distance configuration may be more difficult for FermiNet to handle, especially when its expressive power is limited. Similar phenomenon also shows up in [11] where DMC energy is too high for the long-distance configuration due to the limitation of the trial wavefunction. Moreover, the neural networks for different configurations are trained separately and therefore they may converge to local minima of different quality introducing additional bias in the binding energy calculation.
Supplementary Figure 8 | a. The energy curve for both equilibrium configurations with bond length 4.95 Å and long-distance configurations with bond length 10 Å calculated with FermiNet-VMC and FermiNet-DMC. b. Benzene dimer binding energy calculated with both FermiNet-VMC and FermiNet-DMC. Both results exceed the experimental range while FermiNet-DMC result is closer to the experimental upper limit.

Supplementary Note 13. Binding energy extrapolation

In this note we provide details on the binding energy extrapolation for benzene dimer discussed in main text Section II G.

We first show how to fit the slope parameter \( w \) in main text Eq. 1. For a given VMC training process, we do VMC inference and DMC process at a sequence of steps \( k \) and collect VMC and DMC energy estimate \( \{ E^{(k)}_{\text{VMC}} \} \) and \( \{ E^{(k)}_{\text{DMC}} \} \). Following main text Eq. (1), we assume

\[
E^{(k)}_{\text{DMC}} - E^{(k)}_{\text{VMC}} = w \cdot (E^{(k)}_{\text{VMC}} - E^{(k)}_{\text{DMC}}) + b + \epsilon^{(k)}
\]

where \( \epsilon^{(k)} \) corresponds to 0-centered random noise.

A simple manipulation of Supplementary Eq. (1) leads to

\[
E^{(i)}_{\text{DMC}} - E^{(j)}_{\text{DMC}} = w \cdot [(E^{(i)}_{\text{VMC}} - E^{(j)}_{\text{VMC}}) - (E^{(i)}_{\text{DMC}} - E^{(j)}_{\text{DMC}})] + (\epsilon^{(i)} - \epsilon^{(j)}) \quad \forall i, j
\]

which can be used to fit for slope \( w \) with data \( \{ E^{(k)}_{\text{VMC}} \} \) and \( \{ E^{(k)}_{\text{DMC}} \} \) using least squares.

From main text Fig. 5c, we can tell that the slope of fitted line for benzene dimers with different bond length are very close. Therefore we choose to fit a single slope \( w \) with the data from two configurations combined, as shown in Supplementary Fig. 9.

With the fitted slope \( w_0 \), we have extrapolation from Supplementary Eq. (1)

\[
E_{\text{ex}, s} = (1 + w_0) \cdot E^{(i)}_{\text{DMC}, s} - w_0 \cdot E^{(i)}_{\text{VMC}, s} - b - \epsilon^{(i)}
\]

where \( s \) denotes the bond length for the benzene dimer and can choose value from 4.95 and 10 in this case. Therefore,

\[
\Delta E_{\text{ex}} = E_{\text{ex}, 10} - E_{\text{ex}, 4.95} = (1 + w_0) \cdot (E^{(i)}_{\text{DMC}, 10} - E^{(i)}_{\text{DMC}, 4.95}) - w_0 \cdot (E^{(i)}_{\text{VMC}, 10} - E^{(i)}_{\text{VMC}, 4.95}) - (\epsilon^{(i)} - \epsilon^{(4.95)})
\]

Namely, with this scheme, we can get the extrapolated binding energy \( \Delta E_{\text{ex}} \) from energy difference from any pair of training steps, modulo the fitting error. And this explains why the distribution of our extrapolated result concentrates so well in main text Fig. 5b. By averaging the Supplementary Eq. (3) over all chosen steps \( i \) and \( j \), we significantly reduce the fitting error and thus get a much more accurate binding energy estimate, also shown in main text Fig. 5b.
Supplementary Figure 9. For a pair of steps $i$ and $j$ in VMC training for benzene dimer, we plot two quantities showing up in Supplementary Eq. (2), namely $[(E_{\text{VMC}}^{(i)} - E_{\text{VMC}}^{(j)}) - (E_{\text{DMC}}^{(i)} - E_{\text{DMC}}^{(j)})]$ vs. $E_{\text{DMC}}^{(i)} - E_{\text{DMC}}^{(j)}$. The data for benzene dimer with bond length 4.95 Å and 10 Å have very close slope. With those two set of data combined, we fit a straight line with least squares.

Supplementary Note 14. CCSD(T) calculations and their extrapolations

All CCSD(T) results in this paper are obtained with Psi4 [12]. Four-center electron repulsions are approximated with density fitting. Instead of the usual Dunning basis sets (cc-pVXZ) for valence-only calculation, we used Dunning basis sets (cc-pCVXZ) for valence-core correlations which approximate the exact solution to the all-electron problem. Since the corresponded auxiliary basis set for cc-pCV5Z is not available in the current version of Psi4 (v1.6), we assigned cc-pV5Z-jkfit as the auxiliary basis for cc-pCV5Z basis.

For the extrapolation of complete basis set (CBS) limit, we follow the exponential form [13]

$$E(X) = E_{\infty} + B e^{-aX}$$

(4)

where $X$ is the cardinal number corresponding to the number of basis functions for each atomic orbital (e.g. 2 for cc-pCVDZ, 3 for cc-pCVTZ). $E_{\infty}$ is the energy at CBS limit. $B$ and $a$ are parameters needed to fit. Hartree-Fock energy, CCSD correlation energy, and (T) contribution are fitted separately using the same formula. Then three extrapolated contributions are added to the CCSD(T)/CBS result. We denote CBS(2-4) and CBS(3-5) as the extrapolations using cc-pCVXZ (X=2,3,4), and cc-pCVXZ (X=3,4,5) respectively. The detailed calculation for benzene is summarized in Table 13.

Supplementary Table 13 | Extrapolation of CBS limit for benzene molecule

| basis     | HF     | CCSD   | CCSD(T) | $E_{corr}$ | (T) contribution |
|-----------|--------|--------|---------|------------|-----------------|
| cc-pCVDZ  | -230.724368 | -231.777896 | -231.815816 | -1.053528 | -0.037920       |
| cc-pCVTZ  | -230.779735 | -232.057476 | -232.124466 | -1.277741 | -0.054970       |
| cc-pCVQZ  | -230.793378 | -232.143423 | -232.202980 | -1.350045 | -0.059557       |
| cc-pCV5Z  | -230.796283 | -232.167077 | -232.227835 | -1.370788 | -0.060758       |
| CBS(2-4)  | -230.797839 | -232.243544 | -1.384460 | -0.061245 |
| CBS(3-5)  | -230.797069 | -232.237836 | -1.379133 | -0.061184 |

Our best estimate of CCSD(T)/CBS for benzene is $-232.237386$ Ha, given by CBS(3-5). The difference between CCSD(T)/CBS(2-4) and CCSD(T)/CBS(3-5) is about 6 mHa. We expect the extrapolation error of CBS(3-5) for Benzene molecule to be of the order of 1 mHa. The uncertainty of the extrapolation scheme is mostly contributed by the extrapolation of $E_{corr}$. Hartree-Fock energy and (T) contribution converge fast with respect to the basis set. Their uncertainties (difference between CBS(2-4) and CBS(3-5)) are less than 1 mHa, and almost negligible.

For Benzene dimer, the largest basis set we calculated is cc-pCVQZ. Hence, our best result for Benzene dimer is obtained by CBS(3-5), which gives $-464.486530$ Ha. We expect the extrapolation error of CBS(2-4) for Benzene dimer is of the order of 10 mHa, due to the uncertainty of CBS(2-4) and the doubled system size compared to a single Benzene molecule. We also calculated CCSD(T)/CBS(2-4) binding energy $E_{\text{binding}} = 2 \times E_{\text{mono}} - E_{4.95}$ where $E_{\text{mono}}$ is the energy of benzene molecule, and $E_{4.95}$ is the energy of benzene dimer at 4.95 Å. The binding energy is calculated for each finite basis set first. Then, they are extrapolated with Supplementary Eq. (4). The binding energy converges with respect to the basis set much faster than the absolute energy does (Table 14). The CBS(2-4) estimate for Benzene dimer is 2.66 kcal/mol which is already consistent with the existing result [14, 15]. Leveraging this fact, we are able to calculate the Benzene dimer CBS(3-5) result with Benzene monomer CBS(3-5) result and binding energy (2-4) result, which gives $-464.479015$ Ha. Namely $E_{4.95}(3-5) = 2 \times E_{\text{mono}}(3-5) - E_{\text{binding}}(2-4)$. We believe this is more reliable.
### Supplementary Table 14 | Extrapolation of CBS limit for binding energy of benzene dimer

| basis     | CCSD(T) binding energy (hartree) | (kcal/mol) |
|-----------|----------------------------------|------------|
| cc-pCVDZ  | 0.004855                         | 3.046686   |
| cc-pCVTZ  | 0.004659                         | 2.923544   |
| cc-pCVQZ  | 0.004526                         | 2.839844   |
| CBS(2-4)  | 0.004243                         | 2.662232   |

than the CBS(2-4) result and report it in the main text Figure 5a. Here we don’t consider the Basis Set Superposition Error (BSSE) because the difference between uncorrected and BSSE-corrected CBS estimates is negligible, according to [14].
Supplementary Note 15. Details on the divergence measuring nodal surface difference

The neighborhood $S_\epsilon$ and mapping $\phi$ introduced in main text Section IV F for the wavefunction $\Psi$ and nodal surface $S$ is difficult to compute in practice. Instead we propose easier-to-compute alternatives as approximation.

The major difficulty here is that given a point $y$ and a set $S$, it’s hard to find the point $z \in S$ that is closest to $y$. We propose an approximation as follows. Note that the set $S$ of our interest is always a nodal surface associated with certain wavefunction $\Psi$. Instead of looking for $z \in S$ that is closest to $y$, we move $y$ in the direction in which the value of $\log(|\Psi|)$ decreases the fastest, namely we form the integral curve of the vector field $-\nabla \log(|\Psi|)$ starting from $y$. And we use the intersection of this integral curve and $S$ as the target point $z$, signaled by the change of sign of $\Psi$, and we approximate the distance from $y$ to $S$ as

$$d(y, S) = d(y, z)$$

In other words, we are looking for $z \in S$ such that there exists curve $l(t)$ satisfying

$$\begin{cases} \frac{dl}{dt} = -\nabla \log(|\Psi(l)|) \\ l(0) = y \\ l(t_0) = z \in S \end{cases}$$

where $t_0$ could be an arbitrary number, either positive or negative. Denote the function mapping $y$ to $z \in S$ as $\xi$, and $\xi$ is used as our practical substitute for function $\phi$ mentioned in main text Section IV F looking for the closest point to $y$ in $S$. Function $\xi$ is well-defined in the neighbourhood of $S$, guaranteed by the local existence and uniqueness of ODE.

Similarly, we propose the alternative of $S_\epsilon = \{x|d(x, S) < \epsilon\}$ as

$$\bar{S}_\epsilon = \{x|\Psi(x) \cdot \Psi(x - \epsilon \cdot \frac{\nabla \log(|\Psi(x)|)}{||\nabla \log(|\Psi(x)|)||} < 0\}$$

where $S$ is the nodal surface for wavefunction $\Psi$.

Our algorithm to compute the approximated divergence is in Supplementary Algo. 1.
**Supplementary Algorithm 1**: Divergence for nodal surface difference pseudocode.

**Data**: wavefunction \( \Psi \) for nodal surface \( S \). wavefunction \( \Phi \) for nodal surface \( T \). A batch of walkers \( W \) following the distribution of normalized \( \Psi^2 \). The number of MCMC steps \( M \) to run in each iteration. \( \epsilon \) controlling the size of neighborhood of nodal surfaces. \( \eta \) controlling the discretization of integral curve when moving towards the nodal surface.

**Output**: Divergence value \( D \) measuring difference between \( S \) and \( T \).

1. Initialize set \( L \) as an empty set.  \( \triangleright \) Sample points from \( S_\epsilon \) according to \( \Psi^2 \)
2. While \( \text{length}(L) < K \)
   3. For each walker \( r_i \) in \( W \)
      4. Run MCMC procedure from \( r_i \) for \( M \) steps to get a new walker \( q_i \).
      5. \( u_i = q_i - \epsilon \cdot \frac{\nabla \log(|\Psi(q_i)|)}{||\nabla \log(|\Psi(q_i)|)||} \)
      6. If \( \Psi(u_i) \cdot \Psi(q_i) < 0 \)
         7. Insert \( q_i \) into \( L \)
     8. \( r_i \leftarrow q_i \)
   9. End
10. End
11. Initialize set \( O \) as an empty set. \( \triangleright \) Move samples in \( L \) closer to node Surface \( S \) of \( \Psi \)
12. For each walker in \( L \)
   13. Move along the vector field \( \frac{\nabla \log(|\Psi|)}{||\nabla \log(|\Psi|)||} \) with tiny step \( \eta \) until the sign of \( \Psi \) changes
   14. \( o_i \leftarrow \) the point right before the sign changes
   15. Insert \( o_i \) into \( O \)
16. End
17. For each walker \( o_j \) in \( O \) \( \triangleright \) Move samples in \( O \) closer to node surface \( T \) of \( \Phi \)
18. Move along the vector field \( \frac{\nabla \log(|\Phi|)}{||\nabla \log(|\Phi|)||} \) with tiny step \( \eta \) until the sign of \( \Phi \) changes
19. \( p_i \leftarrow \) the point right before the sign changes
20. \( d(o_i, p_i) \)
21. End
22. Compute \( D = \text{Avg}(d(o_i, p_i)) \)
**Supplementary References**

[1] Richard Needs, Mike Towler, Neil Drummond, and Pablo López Ríos. User’s guide version 2.13 (2019), 2019.

[2] Henrik Flyvbjerg and Henrik Gordon Petersen. Error estimates on averages of correlated data. *The Journal of Chemical Physics*, 91(1):461–466, 1989.

[3] RJ Needs, MD Towler, ND Drummond, Pablo Lopez Rios, and JR Trail. Variational and diffusion quantum monte carlo calculations with the casino code. *The Journal of chemical physics*, 152(15):154106, 2020.

[4] Andrea Zen, Sandro Sorella, Michael J Gillan, Angelos Michaelides, and Dario Alfe. Boosting the accuracy and speed of quantum monte carlo: Size consistency and time step. *Physical Review B*, 93(24):241118, 2016.

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

[6] James S. Spencer, David Pfau, Aleksandar Botev, and W. M. C. Foulkes. Better, Faster Fermionic Neural Networks. *arXiv:2011.07125 [physics]*, November 2020. arXiv: 2011.07125.

[7] James Spencer and David Pfau. FermiNet: Fermionic Neural Networks, 2020.

[8] James Martens and Roger Grosse. Optimizing neural networks with kronecker-factored approximate curvature. In *International conference on machine learning*, pages 2408–2417. PMLR, 2015.

[9] Aleksandar Botev and James Martens. KFAC-JAX, 2022.

[10] Jeffmin Lin, Gil Goldshlager, and Lin Lin. Explicitly antisymmetrized neural network layers for variational monte carlo simulation, 2021.

[11] Arne Lüchow and James B Anderson. Accurate quantum monte carlo calculations for hydrogen fluoride and the fluorine atom. *The Journal of chemical physics*, 105(11):4636–4640, 1996.

[12] Daniel G. A. Smith et al. Psi4 1.4: Open-source software for high-throughput quantum chemistry. *The Journal of Chemical Physics*, 152(18):184108, 2020.

[13] Asger Halkier, Trygve Helgaker, Poul Jørgensen, Wim Klopper, and Jeppe Olsen. Basis-set convergence of the energy in molecular hartree–fock calculations. *Chemical Physics Letters*, 302(5-6):437–446, 1999.

[14] Evangelos Miliordos, Edoardo Aprà, and Sotiris S Xantheas. Benchmark theoretical study of the π−π binding energy in the benzene dimer. *The Journal of Physical Chemistry A*, 118(35):7568–7578, 2014.

[15] J. R. Grover, E. A. Walters, and E. T. Hui. Dissociation energies of the benzene dimer and dimer cation. *The Journal of Physical Chemistry*, 91(12):3233–3237, 1987.