Electronic Structure

Training models using forces computed by stochastic electronic structure methods

David M Ceperley[1,*] 🅒, Scott Jensen[1] 🅒, Yubo Yang[2] 🅒, Hongwei Niu[3], Carlo Pierleoni[4] 🅒 and Markus Holzmann[5]

[1] Department of Physics, University of Illinois, Urbana, IL 61801, United States of America
[2] Center for Computational Quantum Physics, Flatiron Institute, New York, NY 10010, United States of America
[3] Ningde Amperex Techology Ltd, Ningde, Fujian 352100, People's Republic of China
[4] Department of Physical and Chemical Sciences, University of L'Aquila, Via Vetoio 10, I-67010 L'Aquila, Italy
[5] Univ.Grenoble Alpes, CNRS, LPMMC, Grenoble 38000, France
[*] Author to whom any correspondence should be addressed.

E-mail: ceperley@illinois.edu

Keywords: quantum Monte Carlo, potential energy surface, machine learning

Abstract

Quantum Monte Carlo (QMC) can play a very important role in generating accurate data needed for constructing potential energy surfaces. We argue that QMC has advantages in terms of a smaller systematic bias and an ability to cover phase space more completely. The stochastic noise can ease the training of the machine learning model. We discuss how stochastic errors affect the generation of effective models by analyzing the errors within a linear least squares procedure, finding that there is an advantage to having many relatively imprecise data points for constructing models. We then analyze the effect of noise on a model of many-body silicon finding that noise in some situations improves the resulting model. We then study the effect of QMC noise on two machine learning models of dense hydrogen used in a recent study of its phase diagram. The noise enables us to estimate the errors in the model. We conclude with a discussion of future research problems.

1. Introduction

A key technology in the today's materials science and related fields is the ability to perform simulations of atoms and molecules using Molecular Dynamics (MD) or Monte Carlo (MC) methods. Because electrons are more than a thousand times lighter than the nuclei, at room temperatures we can often assume that electrons are in their ground state. We can then 'integrate them out' and deal with the motion of the nuclei alone by using the Born–Oppenheimer separation between electronic and nuclear degrees of freedom. Once the electrons have been eliminated, the resulting 'potential energy surface' (PES), $V(R)$, is a function of $3n_I$ coordinates (R) of the n_I nuclei. Traditionally this potential has been approximated by an empirical function, such as a pair potential, or calculated 'on-the-fly' using density functional theory [1].

Much progress has been made in approximating the PES using a neural network or machine learning (ML) model by Behler and Parrinello[2], Bartok *et al*[3], Zhang *et al*[4–7], Chmiela *et al*[8] and many others. In this approach, the ML model is trained (or fit) to a set of energies and forces of typical configurations of the modeled system. The data often comes from a density function theory (DFT) or other electronic structure method. However, such data can be biased by approximations used in those methods.

We examine in this paper whether stochastic methods, such as variational and diffusion quantum Monte Carlo (QMC) can be used to provide the training data so that the resulting model will be more accurate. Such an approach has been considered previously, for example in work by Alfe *et al*[9], Sorella *et al*[10, 11], Niu *et al*[12], Huang and Rubenstein[13], Nakano *et al*[14] and Cheng *et al*[15]. But several aspects of this approach, such as the effects of the noise on the model, have only been touched on [13].

In the following we introduce QMC methods, and discuss various errors that enter into the models and their relationships. We then discuss several examples such as those based on linear regression, a ML model for silicon and for dense hydrogen. We conclude with observations on future research questions.

2. Quantum Monte Carlo methods for electronic structure

We briefly review some QMC algorithms that estimate electronic properties using a stochastic process. We write $R = \{s_1, s_2 \ldots s_{n_I}\}$ for the n_I nuclear coordinates with charge Z_j on the jth ion and $r = \{r_1, r_2 \ldots r_{n_e}\}$ for the n_e electron coordinates. The non-relativistic electronic Hamiltonian in atomic units is:

$$\hat{H} = \sum_{i=1}^{n_e} \left[-\frac{1}{2} \hat{\nabla}_i^2 - \sum_{j=1}^{n_I} \frac{Z_j}{|r_i - s_j|} \right] + \sum_{i<j} \frac{1}{|r_i - r_j|}. \tag{1}$$

R_α will refer to the αth sampled configuration (a $3n_I$ vector) used for training or evaluation. By $\mathbf{F}_j(R) = -\nabla_j V(R)$ we mean the 3 vector of forces on the jth ion for a configuration R.

We mention the QMC methods that we think will be useful for generating ML data. For a detailed discussion of the various methods see [16].

Variational Monte Carlo (VMC) Metropolis (i.e. Markov Chain MC or MCMC) [17] is used to sample electronic coordinates from the square of a trial wavefunction $|\Psi_T(r|R)|^2$ and calculate its (variational) energy:

$$E_V(R) = \frac{\int dr \Psi_T^*(r|R) \hat{H} \Psi_T(r|R)}{\int dr |\Psi_T(r|R)|^2}. \tag{2}$$

The simplest correlated trial function used in QMC is the Slater-Jastrow form:

$$\Psi_T(r|R) = \exp\left(-U(r|R)\right) \det_{ik}\left(\phi_k(r_i|R)\right) \tag{3}$$

where $U(r|R)$ is a bosonic Jastrow factor, and $\phi_k(r|R)$ is a set of single-electron orbitals. Both of these terms are varied to minimize $E_V(R)$. There has been recent work on ML trial wave functions [18–23] which achieve excellent variational bounds. However, the current methods are quite time consuming, requiring very long optimizations. We argue that ML needs 'lightweight' QMC methods in order to provide sufficient training data. For example, in dense hydrogen we work with analytic forms requiring little optimization of trial functions [24]. Forces are then calculated as the derivative with respect to the ionic coordinates, invoking the Hellman–Feynman theorem [25].

Projector MC takes a trial wavefunction and uses the Hamiltonian to project out errors of the trial function:

$$\Phi = \lim_{t \to \infty} e^{-t\hat{H}} \Psi_T. \tag{4}$$

There are several methods by which this can be accomplished. Since electrons are fermions, the methods must employ an ansatz to control the stochastic error for many-electron systems such as the fixed-node [26] or fixed-phase [27] approximations. Fixed-node and fixed-phase methods solve exactly for the modulus of the wavefunction within statistical errors but the bias from the ansatz can be significant [16] if only a single Slater determinant is used for antisymmetry. One can use signful methods [28] to correct this error.

Diffusion MC (DMC) [26] accomplishes the projection by representing the system as an ensemble of many-body configurations that execute a drifting, branching random walk. DMC has a mixed estimator bias for forces; there are several procedures for correcting this bias.

Reptation quantum Monte Carlo (RQMC) holds a segment of a many-body path in the computer memory and adds or subtracts from the path using a modified MCMC [29, 30]. If the projection time is large enough, forces taken in the middle of the path do not have the mixed estimator bias but they will still have the fixed-node error.

Path integral methods [31] work at non-zero temperature and differ from RQMC in working with closed paths in configuration space. PIMC methods could address important problems for warm dense matter addressing a regime that is difficult for most electronic structure methods. However, PIMC has only been made practical for temperatures above some fraction (25%) of the Fermi temperature. Sampling is done with MCMC and must include sampling electron permutations. Early work calculated forces between protons in an electron bath [32].

For ML the ability to compute forces is crucial. There are several methods for computing forces with QMC [33, 34]. For generating ML data, the efficiency, namely how much computer resources are needed to

reach a given error on the forces is important. Huang and Rubenstein [13] have discussed recent methods to calculate forces using QMC. More research is needed on the relative efficiencies, the biases, and the appropriate domains of the QMC methods for the calculation of accurate forces.

In the past, many researchers concluded that the computer resource requirements to compute forces using QMC were excessive. However, available resources continue to grow year-by-year while the cost of human resources needed to come up with reliable predictions is not decreasing. A more reliable electronic structure method can reduce this human cost and thereby allow for more efficient simulations of materials. One of the main roadblocks in the past has been the availability of high quality QMC codes but this situation has improved with development of such packages as QMCPACK [35, 36], TURBORVB [37], and CASINO [38]. In addition to more plentiful computer resources, QMC techniques are improving. We believe their efficiency could be increased much further as we discuss below, especially for the problem of determining unbiased forces.

QMC, in general, is more accurate than other electronic structure methods. By that we mean that it has a lower bias with respect to the exact ground state than other methods such as those based on DFT where the choice of the functional is guided by external experimental knowledge. Experimental data of high quality is rarely available in many applications. Its low bias is the most important reason to use QMC to generate data. QMC is also a more general technique since it is able to handle highly correlated systems such as lattice models, superconductors, quantum crystals, and liquid helium.

The continuum basis set used in VMC, DMC, RQMC and PIMC is better suited for disordered systems, such as a liquid, since distortions caused by a finite basis set are absent. For deterministic methods, one needs to assure convergence in the basis set, e.g. the plane wave cutoff or the expansion in local basis functions.

A very important advantage of stochastic methods is that extra degrees of freedom that need to be integrated over do not necessarily increase the computational effort. For example, twist averaging [39], is used to eliminate some finite size effects by integrating over the Brillouin zone of the super-cell. If we compute QMC properties at P twist values, this does not increase the QMC computation effort since the P points all serve to reduce the statistical error. Instead, within deterministic methods such as DFT, this will increase the computer time by a factor of P. Another example of this advantage occurs in the path integral algorithm for quantum protons which requires the electronic energy of several replica of the protons to accept an update [24]. The same logic applies to getting data to train a ML model. The forces at a large set of configurations can be obtained using the same computer budget as a small set of configurations although the stochastic errors of the large set will be larger. But as we show below, the large set is better for constructing the ML model. Hence there is less importance to selecting an optimal set of configurations to use in training the model.

We expect that important applications of ML will be to complex, multi-component, many-body systems, for example a ternary liquid. For those systems, one does not know how many configurations will be required, how to place them, and how to assure that the energies are unbiased. With QMC, one can oversample the configuration space. Even if many of the points are clustered, the computational effort serves to reduce the error in the fitted potential energy surface. We do not wish to imply that methods to place the QMC points in an efficient manner will not improve the efficiency but that optimal placement is not always necessary.

Noisy data is also good for training neural networks and is often added to deterministic methods to aid in optimization. It can help to find the best model more easily [40]. With noisy data, one does not need to worry about data clustering: one will not have a problem with oversampling. Here we consider the case where the training data has stochastic errors.

The QMC computer time scales well with the number of electrons compared to other methods such as coupled cluster. For a recent discussion of the scaling and accuracy of various electronic structure methods see [41]. For systems with many hundreds of delocalized electrons the computation of determinants to enforce antisymmetry will cost n_e^3 operations to do a multiparticle move. This is the same scaling as in DFT algorithms. (Note than one is calculating $3n_I$ forces during this move.) In practice, because computing the determinant is a simple linear algebra operation, QMC time is typically dominated by other operations such as calculation of the charge-charge interaction and scales more slowly than n_e^3. QMC can easily profit from massively parallel computer environments since memory demands can be kept modest.

We will examine some of these issues in the rest of the article. We focus on methods that do not have basis set errors (i.e. work in the continuum) and work in the electronic ground state, that is the VMC, DMC and RQMC methods.

3. Errors in quantum Monte Carlo and machine learning models

Let $V(R)$ be the exact energy (i.e. the 'ground truth') and $\mathbf{F}_k(R) = -\nabla_k V(R)$ the corresponding force on the kth atom. The energy and forces will be estimated using a method for numerically solving the ground state Schrödinger equation for the nuclei in a given position obtaining for a set of M configurations $\{R_\alpha\}$ ($1 \leqslant \alpha \leqslant M$) with estimated values for the potential ($\hat{V}_\alpha$) and forces ($\hat{\mathbf{F}}_k$). Using this data we then make a model $V_m(R)$. To help with further discussion we explicitly list and discuss the errors and their relations. Although the listed formulas below are for energies, the analogous errors in the forces are more important in ML.

$$\vec{\sigma}_V \equiv \hat{V}_\alpha - V(R_\alpha) \tag{5}$$

$$\vec{\rho}_V \equiv \hat{V}_\alpha - V_m(R_\alpha) \tag{6}$$

$$\vec{\epsilon}_V \equiv V_m(R_\alpha) - V(R_\alpha) = \vec{\sigma}_V - \vec{\rho}_V \tag{7}$$

$$\sigma_V = \sqrt{\frac{1}{M}\sum_\alpha \left(\hat{V}_\alpha - V(R_\alpha)\right)^2} \tag{8}$$

$$\rho_V = \sqrt{\frac{1}{M}\sum_\alpha \left(\hat{V}_\alpha - V_m(R_\alpha)\right)^2} \tag{9}$$

$$\epsilon_V = \sqrt{\frac{1}{M}\sum_\alpha \left(V_m(R_\alpha) - V(R_\alpha)\right)^2} \tag{10}$$

$$\chi_V^2 = M\left[\frac{\rho_V}{\sigma_V}\right]^2 \qquad \chi_F^2 = 3n_I M\left[\frac{\rho_F}{\sigma_F}\right]^2. \tag{11}$$

The model error is the deviation of the model with respect to the ground truth: $\epsilon(R) = V_m(R) - V(R)$. Even with good data and training, unless the model is a sufficiently general function of $3n_I$ variables, the model error will be non-zero.

The stochastic error is the difference between the QMC value (with noise) and the exact value: $\sigma_\alpha = \hat{V}_\alpha - V(R_\alpha)$. We assume that QMC is unbiased so that $\langle \sigma_\alpha \rangle = 0$: after enough averaging one will recover the exact energies and forces. For convenience we will define the ground truth as the exact fixed-node (without stochastic noise) energies and forces. Note that this definition of the reference depends on the nodal surface. Estimates of the stochastic errors on energies and forces are determined from the fluctuations in the QMC run, from the same data as the mean values and will depend on the ionic coordinates. An example of QMC noise for hydrogen is shown in figure 4.

Fitting error tells us whether the model fits the noisy, but unbiased data: $\rho_\alpha = \hat{V}_\alpha - V_m(R_\alpha)$.
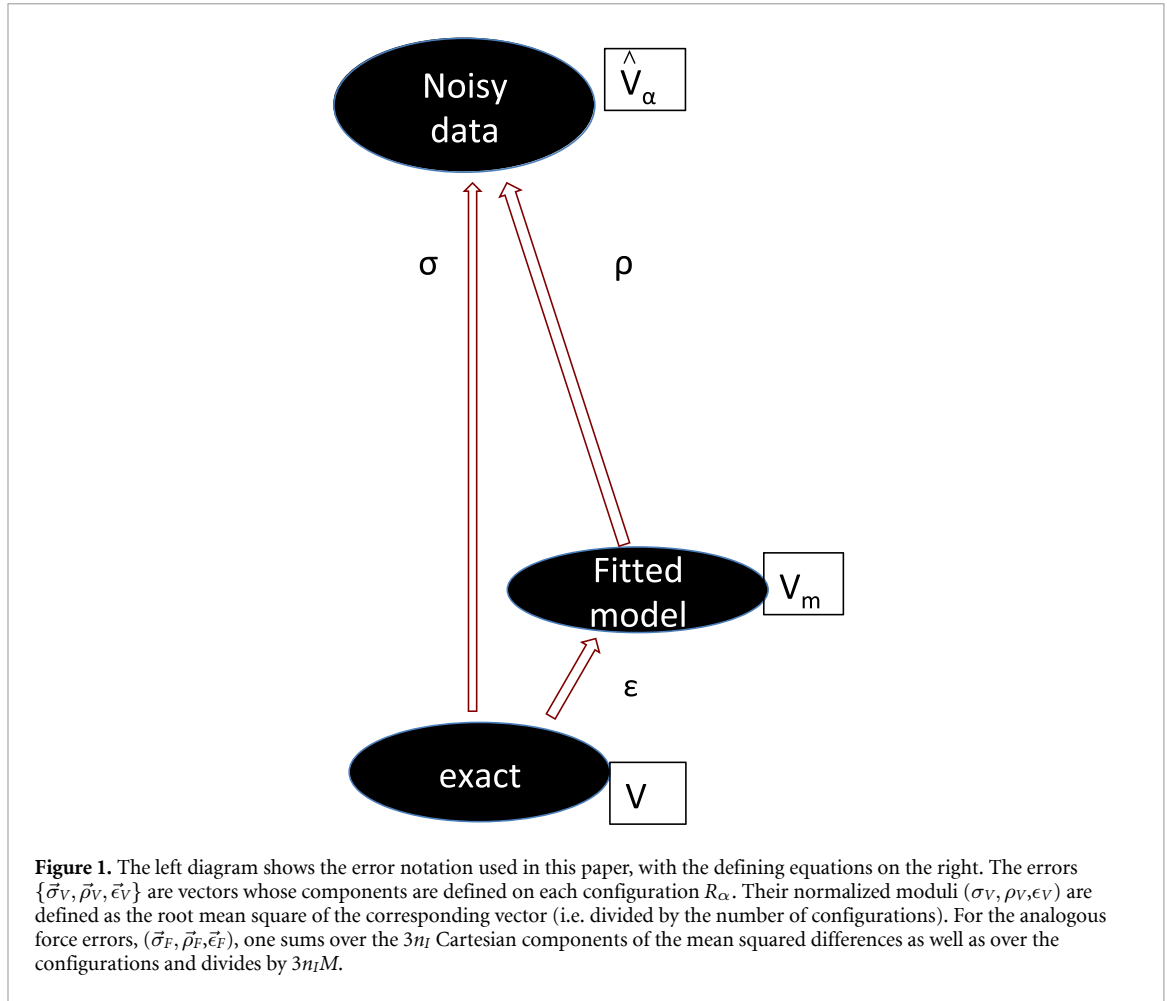
Chisquare is the negative logarithm of the likelihood that the model fits the QMC data: $\chi^2 = \rho^2/\sigma^2$.

Selection error. Since the assumed model is not perfect, it matters how many configurations M are selected for training and where they are placed. We will assume that they are selected independently from a probability density $F(R)$ by using a stochastic process such as MC or MD. How exactly they should be selected (e.g. the function $F(R)$) is an important question which we will not address in this paper. Note that probability distribution for assessing the fitting and model errors, the testing distribution, could be different from $F(R)$.

Inter-model or cross-mode validation error is the difference in the energies and forces between two fitted models ($m1$ and $m2$): $\delta(R) = V_{m1}(R) - V_{m2}(R)$. It gives an indication of the importance of the stochastic error, the selection error, the model error and the fitting robustness on the models.

Bias of the data could either be a result from controlled errors, those that can be studied and eliminated such as time step error, basis set error, convergence errors, or uncontrolled errors such as the fixed-node error of projection quantum Monte Carlo. Finite-size errors (FSEs) result from data being generated for small systems but used in simulations for larger systems. The QMC calculations are done on systems in periodic boundary conditions containing hundreds to thousands of electrons. Corrections may be needed [34, 39, 42] to use the data for models used for much larger systems. These FSE are not particular to QMC but are similar to FSE for any accurate electronic structure method that takes electron correlation into account. Because the FSE are generic to all methods, we will not consider them further in this paper.

One of the goals of Uncertainty Quantification is to estimate how accurate the model is. We can easily estimate σ and ρ but ϵ is more difficult since we do not have precise estimates of the energy or force for many-body systems. We can consider these 3 errors to be vectors with N_c components where $N_c = M$ for the energy errors and $N_c = 3n_I M$ for the force errors. In this paper we define the norm of an error vector as

Figure 1. The left diagram shows the error notation used in this paper, with the defining equations on the right. The errors $\{\vec{\sigma}_V, \vec{\rho}_V, \vec{\epsilon}_V\}$ are vectors whose components are defined on each configuration R_α. Their normalized moduli $(\sigma_V, \rho_V, \epsilon_V)$ are defined as the root mean square of the corresponding vector (i.e. divided by the number of configurations). For the analogous force errors, $(\vec{\sigma}_F, \vec{\rho}_F, \vec{\epsilon}_F)$, one sums over the $3n_I$ Cartesian components of the mean squared differences as well as over the configurations and divides by $3n_I M$.

$\sigma = \sqrt{\frac{1}{N_c}\sum_\alpha \sigma_\alpha^2}$. Thus we define $\{\sigma_V, \epsilon_V, \rho_V, \sigma_F, \epsilon_F, \rho_F\}$ as detailed in figure 1. If we define the vectors on a common set of configurations, the energy and force norms will obey the triangle inequalities:

$$|\rho - \sigma| \leqslant \epsilon \leqslant \rho + \sigma. \tag{12}$$

The norms should converge to a limit as $M \to \infty$ so that the inequalities will apply in that limit. In that limit we expect that ϵ_V will converge to

$$\lim_{M\to\infty} \epsilon_V = \sqrt{\int dR\, F(R)\,(V_m(R) - V(R))^2}. \tag{13}$$

A finite number of configurations is a stochastic (i.e. Monte Carlo) evaluation of this integral. We can construct other triangle inequalities using different ML models.

We can make tighter bounds on the model error, ϵ, by considering how the fluctuations in the QMC data could be correlated with the errors of the model. Suppose we assume $\vec{\epsilon} = \eta\vec{\sigma} + \vec{\epsilon}_0$ where $\vec{\epsilon}_0$ is the model error vector with zero noise ($\sigma = 0$). In the most extreme procedure the model could consist of taking the closest configuration so that $\eta = 1$. For a good model which is relatively unaffected by the QMC noise because the noise is filtered or averaged over, and for models such as those based on a DFT functional or an empirical potential, we expect $\eta \approx 0$. This is also the case when the test set on which the errors are evaluated is statistically uncorrelated with the training set. Because the vector of QMC errors, $\vec{\sigma}$, is uncorrelated with the ideal model errors $\vec{\epsilon}_0$:

$$\rho^2 = \epsilon^2 + \sigma^2 - \frac{2}{M}\sum_\alpha \sigma_\alpha \epsilon_\alpha = \epsilon^2 + (1 - 2\eta)\sigma^2 \pm 2\sigma\epsilon_0/\sqrt{N_c}. \tag{14}$$

In the case of plentiful data, we expect $\eta \to 0$, so that $\epsilon \to \sqrt{\rho^2 - \sigma^2} \pm \sigma/\sqrt{N_c}$. In general, we expect that $0 \leqslant \eta \leqslant 1$, leading to tighter bounds on ϵ in terms of σ and ρ:

$$\sqrt{\rho^2 - \sigma^2} \quad \leqslant \epsilon \leqslant \sqrt{\rho^2 + \sigma^2} \qquad \sigma \leqslant \rho$$

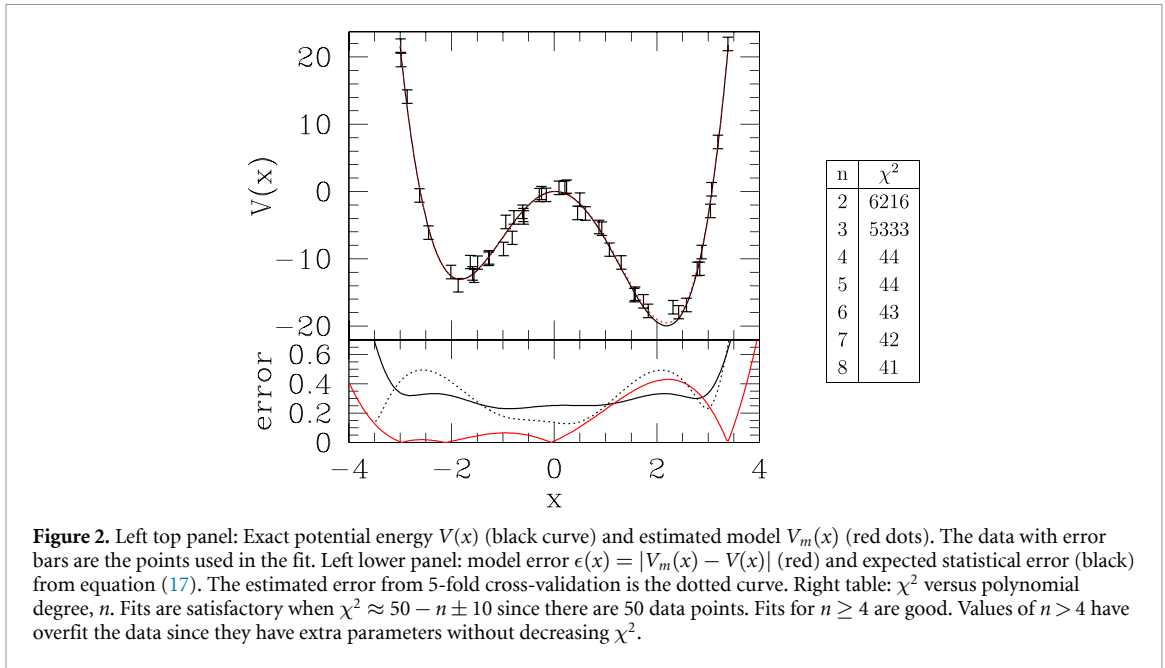| n	χ^2
2	6216
3	5333
4	44
5	44
6	43
7	42
8	41

Figure 2. Left top panel: Exact potential energy $V(x)$ (black curve) and estimated model $V_m(x)$ (red dots). The data with error bars are the points used in the fit. Left lower panel: model error $\epsilon(x) = |V_m(x) - V(x)|$ (red) and expected statistical error (black) from equation (17). The estimated error from 5-fold cross-validation is the dotted curve. Right table: χ^2 versus polynomial degree, *n*. Fits are satisfactory when $\chi^2 \approx 50 - n \pm 10$ since there are 50 data points. Fits for $n \geq 4$ are good. Values of $n > 4$ have overfit the data since they have extra parameters without decreasing χ^2.

$$|\rho - \sigma| \quad \leqslant \epsilon \leqslant \sqrt{\rho^2 + \sigma^2} \qquad \sigma \geqslant \rho. \tag{15}$$

To estimate the statistical uncertainties within a model, we can use a K-fold cross-validation, a variation of the jackknife procedure in statistics. One partitions the set of configurations into K 'folds' possibly stratifying the selection of the folds so that each fold contains roughly the same number of configurations at a given density and temperature. The number of folds may be limited by computer resources since each fold requires a model optimization. We used $K = 5$ in the estimates in section 6. From the folds we make K training sets by sequentially dropping a fold from the complete data set. Using those training sets one finds K models, $V_{mk}(R)$, and an average model, $V_{m0}(R) = \sum_k V_{mk}(R)/K$. Summed over a test set of configurations, one finds the rms differences in the energies and forces between the models and the average model. The jackknife estimated model error for the energy is:

$$\delta_V = \sqrt{\frac{K-1}{KM} \sum_{k,\alpha}^{K,M} \left(V_{mk}(R_\alpha) - V_{m0}(R_\alpha)\right)^2} \tag{16}$$

with a similar relation for δ_F but normalized by an additional factor $3n_I$. These errors estimate the overall effect of the stochastic QMC errors, the selection errors and optimization errors, i.e. how would the energies and forces change if one redid the complete analysis with M newly sampled configurations, new QMC calculations on those configurations and a new optimization. They do not include model errors or bias.

The stochastic and selection errors come from the finite sampling and can be reduced by more sampling. The model error can be estimated by showing that the deviation of the model from the data is larger than expected on the basis of equation (14). To quantify the bias, one needs to perform systematic studies of convergence, for example by changing the trial wavefunction or varying the number of atoms.

4. Least squares analysis

The simplest procedure for finding a model to represent the QMC data is to perform a linear least squares fit (LLSF), i.e. a regression. Some leading ML models such as Gaussian approximation potentials (GAPs) [43] are based on LLSF. Non-linear models can be approximated by linear models near their solution. LLSF is a standard numerical method with a guaranteed single solution and methods for assessing the resulting error bars on the model.

We illustrate how having stochastic error affects the results with a simple 1D quartic function, shown in figure 2, by using LLSF with a polynomial basis. We assume that the parameters of the quartic potential (the ground truth) are unknown. The 50 points with error bars represent QMC results for locations chosen at random and uniformly in the range $-3.5 < x < 3.5$. To the 'exact' but unknown potential, we added normally distributed noise with a standard deviation of $\sigma_V = 1$. The noisy potential energies were then fit to a polynomial of degree *n* using LLSF. The fit is the red dotted curve for a quartic polynomial, i.e. $n = 4$. The

IOP Publishing   *Electron. Struct.* **6** (2024) 015011   D M Ceperley *et al*

QMC derived potential energy surface is more accurate than the individual QMC energies and is only different from the exact PES around $x = 2$ by a maximum of 0.4 even though the data had a standard error of 1.

The average error defined in equation (13), computed by propagating errors in the data to errors in the solution (see equation (17)) is found to be $\epsilon_V = 0.326$. Using 5-fold cross-validation we found $\epsilon_V = 0.346$. By fitting to a quartic polynomial, we have assumed that the PES must be smooth. If we had fit to a higher degree polynomial as shown in table 2, the errors would not have decreased. Using the value of ρ, σ, ϵ we can compute the sensitivity parameter and find $\eta = 0.11$. This is reasonable since we have 5 free parameters in the fit with 50 data points so $\eta \approx 5/50 = 0.10$.

For LLSF we want the amount of data to be greater than the number of free parameters. To determine a quartic polynomial, exact energies at five distinct points would have been enough. A deterministic method on this particular low-dimensional problem would, of course, be preferred for generating data since one can saturate the configuration space with accurate noise-free data. However, one cannot saturate the space for realistic models of bulk materials.

The table in figure 2 shows χ_V^2 vs polynomial degree. One has extremely poor fits using quadratic or cubic polynomials, but using a quartic polynomial is excellent. Increasing the order beyond this does not improve the fit and is overfitting the data. The computed error, shown in the bottom graph, is excellent in the region $-3 < x < 3$ but it degrades outside of the region where data is present.

Now consider a general least squares fit (LLSF) in a space of arbitrary dimension, representative of a many-body system. Expand the model as $V_m(R) = \sum_1^P a_k f_k(R)$ where the $f_k(R)$ are P basis functions. Suppose we have unbiased energy data consisting of M configurations $\{R_\alpha\}$ sampled from a distribution $F(R)$ with estimated potential energies $\hat{V}_\alpha$ and standard errors σ_α. We determine a_k by minimizing the likelihood $\chi^2 = \sum_{\alpha=1}^M ((V_m(R_\alpha) - \hat{V}_\alpha)/\sigma_\alpha)^2$ with respect to a_k. The solution is determined by solving linear algebraic equations in the P unknowns [44].

The variance of the energy of the fitted model at a point R

$$\delta_V(R)^2 = \left\langle (V_m(R) - \langle V_m(R) \rangle)^2 \right\rangle = \sum_{kn} f_k(R) \left[\hat{A}^{-1} \right]_{kn} f_n(R) \tag{17}$$

is given in terms of the inverse of the matrix:

$$\hat{A}_{kn} = \sum_{\alpha=1}^M \frac{f_k(R_\alpha) f_n(R_\alpha)}{\sigma_\alpha^2}. \tag{18}$$

The above brackets $\langle \ldots \rangle$ mean that one averages over both the electronic structure process that determines the noise σ_α and the selection of points R_α sampled from $F(R)$.

Let us examine how $\delta(R)$ depends on the 'running' computer time T (i.e. the computer time when you are actually generating forces and energies. We exclude from T the start-up time for the QMC process). We assume that the variance of the QMC noise σ_α is independent of R_α, and is given by

$$\sigma_\alpha^2 = C/T_\alpha \tag{19}$$

for some coefficient C. T_α is the computer time expended for configuration R_α. This implies that the mean value of the matrix A,

$$\left\langle \hat{A}_{kn} \right\rangle = M \left\langle \frac{f_k(R_\alpha) f_n(R_\alpha)}{\sigma_\alpha^2} \right\rangle_F = \frac{\sum_\alpha T_\alpha}{C} \int dR F(R) f_k(R) f_n(R) \tag{20}$$

where $\langle \ldots \rangle_F$ means the average over the selection of points R_α. We assume that T_α is constant or not correlated with R_α. Then it follows from equation (20) that the mean value of $\hat{A}_{mn}$ depends only on the total computer time, $T = \sum_\alpha T_\alpha$. From equation (17), the error in the model will have the same property: the error in the fitted model is only proportional to $T^{-1/2}$: it does not depend on how many points we use to determine the data or how that time is allocated between the various configurations as long as the overall computer time is fixed. This conclusion holds when the configurations are independently sampled from some distribution, the model potential is determined by minimizing the mean squared deviation of the model energy from the noisy energy and the QMC errors are given by equation (19).

It is quite important to use forces in the training because they are much more plentiful than total energies by a factor of $3n_I$. Also forces give local information and integrate well with the ML models which are also local. Use of only the QMC energies to determine the ML model, as was done in [13], requires many more

QMC calculations and results in a much less accurate PES. The likelihood function for optimizing the model including both energies and forces is

$$\mathcal{O} = \chi_V^2 + \lambda \chi_F^2. \tag{21}$$

The likelihood function is the correct object to minimize if the noise is component-by-component independent and if the fluctuations are normally distributed. Both are reasonable assumptions for QMC data, though there can be deviations. The parameter λ should be set to one for maximum likelihood but it is often treated as a free parameter. Including the contribution of the force data, the matrix determining the errors gets an extra contribution:

$$\hat{A}_{kn} = \sum_{\alpha=1}^{M} \frac{f_k(R_\alpha) f_n(R_\alpha)}{\sigma_V^2} + \lambda \sum_{\alpha,\beta,i} \frac{\nabla_{\beta,i} f_k(R_\alpha) \nabla_{\beta,i} f_n(R_\alpha)}{\sigma_F^2} \tag{22}$$

where $\nabla_{\beta,i}$ is the derivative of the βth Cartesian component on the ith atom. The above discussion for the energy follows through. The values of the matrix elements in A have changed, becoming much larger, implying that the stochastic error on the model is much reduced, however the average matrix elements still only depend on the total computer time, not on how many configurations are present.

In the above discussion, we have made some assumptions that need to be tested to see if they apply to realistic problems. What follows are two examples that show this property. First, a model system of many-body silicon where we know the inter-atomic potential and, second, an actual QMC calculation of many-body hydrogen at high pressure.

5. Silicon example

In this section, we give an example to illustrate the effect of stochastic error in training a neural network potential where the exact potential is known but where the potential has more realistic features. We use the Tersoff potential [45] for Si, included in the LAMMPS program [46], as the ground truth. A MACE [47] model was trained on a set of configurations. We do not expect that the MACE model will be able to capture the Tersoff potential perfectly though it should not be too difficult.

The training set was generated by isothermally compressing 64 Si atoms at 3000 K from $0.467\rho_0$ to $1.282\rho_0$, where $\rho_0 = 4.994 \times 10^{-2}$ Å^3 is the ambient density of Si solid. This training set provides a diverse set of local environments over a wide range of densities. We used the MACE model [47] an equivariant message passing model that uses higher body order messages with 32 invariant messages, angular resolution $l_{\max} = 3$, and a cutoff radius of 3.2 Å, the same as the range of the Tersoff potential. Radial features are generated using 8 Bessel basis functions and a 5th order polynomial envelope for the cutoff. The loss function is a parameterized likelihood function with $\lambda = 1.3$ [6] Models were trained with AMSGrad variant of Adam, with default parameters, a learning rate of 0.01 and a batch size of 4.

We trained the model on different sizes of training sets ranging from 660 configurations to 4620 configurations. We performed 3 different optimizations on the same data with those sizes. The first set used the clean data, data without noise, to get a baseline. The trained model got better as more data was added, presumably by covering phase space better. Using the same configurations, we then added normally distributed noise to both the energies and forces but using a larger amount of noise as M increased as detailed in table 1. Because the statistical noise is inversely proportional to the square root of how long one samples in QMC, assuming a fixed total computation budget, the standard deviation of the noise on the energies and forces followed $\sigma \propto \sqrt{M}$. For each configuration we subtracted the mean value of the force. Without this modification, the potential energy would be inconsistent with the forces since any model that only contains differences of coordinates will have a zero mean force.

The values of σ, M and the resulting ϵ are shown in tables 1 and 2 and in figure 3. Table 1 gives errors as evaluated on an independent test set of 100 configurations (set Ω_1). This test set was generated with an MD trajectory obtained by heating a silicon crystal at ambient density from 1000 to 3000 K in 5 picoseconds. This set was not used in the training. Table 2 shows the errors on a different test set (denoted as Ω_2) which was a subset of 400 configuration of the training set with densities within 6% of ρ_0. Strikingly different errors are obtained depending on the number of configuration used for the training and the makeup of the test sets.

We see several things from these examples. The most accurate model was obtained for the largest data set ($M = 4620$) using the clean data for training, obtaining a model with $\epsilon_V^0 = 13$ meV and $\epsilon_F^0 = 1.2$ meV Å^{-1}. Clearly, MACE is able to represent accurately the Tersoff potential near ambient conditions of density if given

---

[6] In the MACE input $\alpha_F = 10$. To convert to the likelihood function $\lambda = \alpha_F(\sigma_F/\sigma_V)^2/(3n_I)$.

Table 1. Energy errors (in meV) and force errors (meV Å^{-1}) as a function of the number of training configurations, M, for the Tersoff model. σ_V is the standard deviation of the noise added to the energies and σ_F to the forces. ϵ is the rms deviation of total energy and force components with respect to the exact values as evaluated on the test set described in the text. ϵ_V^0 and ϵ_F^0 are evaluated on the model trained with noiseless data. Errors in this table were computed using the 100 configurations in the test set (Ω_1, see text).

M	σ_V	ϵ_V^0	ϵ_V	σ_F	ϵ_F^0	ϵ_F
4620	60	13	33	300	1.2	3.9
2310	42	31	36	212	3.4	4.1
1540	34	45	26	173	3.4	4.6
1155	30	1214	94	150	6.2	7.1
924	26	697	63	134	6.4	6.2
660	22	229	496	113	8.6	9.6

Table 2. Errors on a subset of 400 training configurations within 6% of ambient density, test set Ω_2. η is a measure of the sensitivity of the energies and forces to the noise. M is the number of configurations used in training. Numbers in $(\ldots)$ are the statistical errors in the last digits. Other notation and units as in table 1.

M	σ_V	ϵ_V^0	ϵ_V	ρ_V	η_V	σ_F	ϵ_F^0	ϵ_F	ρ_F	η_F
4620	60(2)	103	84(3)	106(4)	−0.06	297.3(8)	43	67.3(4)	301(1)	0.013
2310	43(2)	179	82(4)	95(4)	−0.11	211.5(8)	72	80.0(9)	223(1)	0.016
1540	35(2)	282	298(7)	300(7)	0.16	171.8(7)	69	80(1)	185(1)	0.031
1155	29(2)	520	557(18)	564(18)	−4.55	148.4(7)	94	101(2)	177(1)	0.025
924	29(2)	763	290(13)	286(12)	1.83	133.7(9)	96	100(3)	163(2)	0.040
660	24(2)	306	191(11)	193(12)	0.05	112.5(7)	103	109(2)	152(2)	0.060

enough data. Note that the highest precision on the clean data was only achieved with relatively large data sets; the errors on the energy for smaller data sets were quite erratic. This could be a consequence of the value of λ and the preponderance of force data. On the same data with 60 meV noise, the errors of the model trained with noise were about three times as large. The force errors showed a steady decrease as more data was added and accurate force models were obtained with fewer than one thousand configurations. This illustrates how much more information is provided by forces versus total energies and the insensitivity of the model to noise.

We also see that for the test set (Ω_1) with densities close to ambient pressure of silicon, the force errors were 20 to 30 times smaller than for the subset of the training set (Ω_2) that had a much greater variation in density. The model is much more accurate when tested over a very narrow range of densities even though it has been trained over a much wider range of densities.
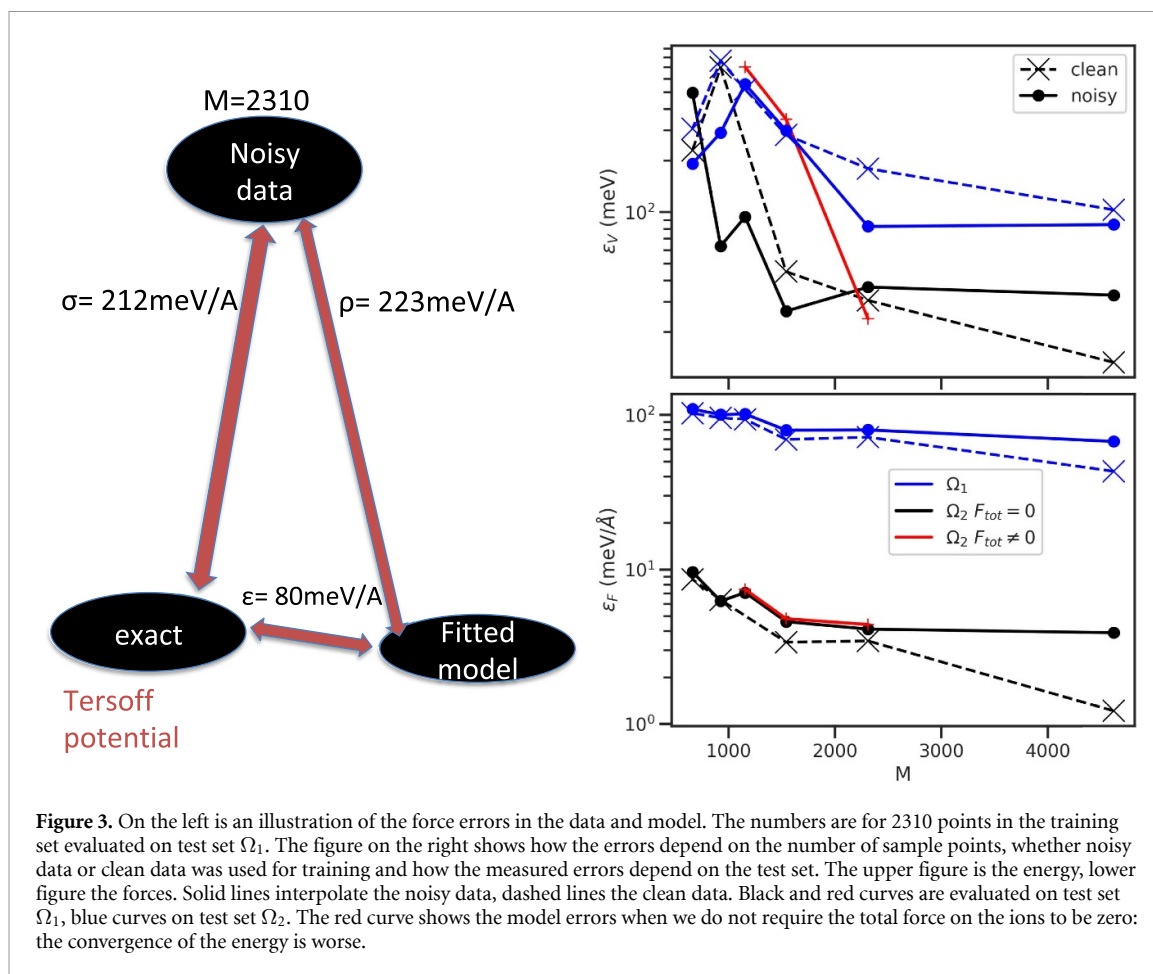
While the model energy was affected by the noise, except for the largest data set, the forces were not affected by the noise. Additional noisy data always improved the error in the forces even though the additional data had larger statistical errors. From the results of this example we conclude that one should always run with the maximum possible value of M even though the resulting statistical errors are increased. Our analysis in the previous section of LLSF remains valid for non-linear models. In practice, the best choice for M will be influenced by the startup time of the QMC relative to the overall computer budget and the time to train the model.

We see from table 2, that the sensitivity parameter η_F is small and decreases as M increases. This implies that we can estimate the model error directly as $\epsilon_F = \lim_{M \to \infty} \sqrt{\rho_F^2 - \sigma_F^2}$ based just on quantities that can be easily computed in a realistic model where we cannot compute the exact potential. However, the results for η_V, the energy value, are erratic and reinforce our finding that the energy model evaluated on test set Ω_2 is not well converged.

In a third set of optimizations we did not subtract out the mean force. The results are shown in figure 3, The force error was little affected but the energy error was increased when the training had fewer than 2000 configurations. It is possible that the choice of λ in the loss function did not put enough importance on the energy errors versus the force errors.
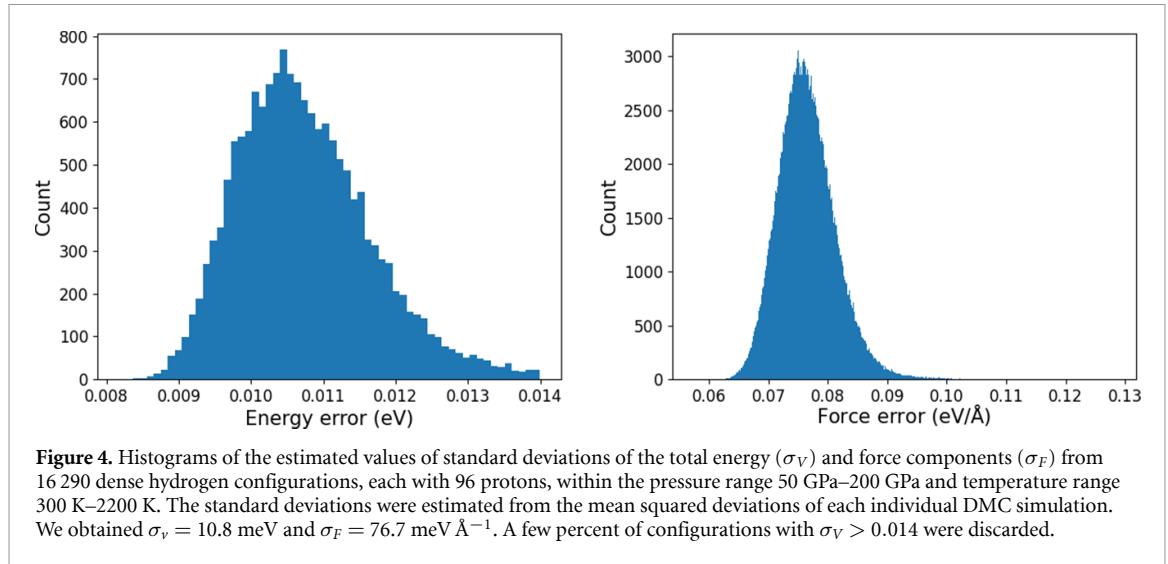
6. Dense hydrogen example

We now discuss a recent calculation [12] of many-body hydrogen under extreme conditions of temperature and pressure. Bulk hydrogen is important both because of its ubiquitous presence in the Universe and in technological applications [48]. Even though it is the 'holy grail' of high pressure studies, its phase diagram above 100 GPa is uncertain. Experiments at high pressure are very difficult so they cannot be used to validate the computational approaches. Hydrogen is a good test case for computer simulation, QMC and ML because

Figure 3. On the left is an illustration of the force errors in the data and model. The numbers are for 2310 points in the training set evaluated on test set Ω_1. The figure on the right shows how the errors depend on the number of sample points, whether noisy data or clean data was used for training and how the measured errors depend on the test set. The upper figure is the energy, lower figure the forces. Solid lines interpolate the noisy data, dashed lines the clean data. Black and red curves are evaluated on test set Ω_1, blue curves on test set Ω_2. The red curve shows the model errors when we do not require the total force on the ions to be zero: the convergence of the energy is worse.

one does not need to use a pseudopotential so that does not complicate the algorithm or cause errors. Although it does not have core electrons, its electronic structure has other complications such as a still elusive molecular-atomic transition that may coincide with its insulator-metal transition as well as effects caused by the very large quantum zero-point of the protons resulting in solid H_2 being disoriented at low temperatures. QMC has been used extensively on this system [48]. Statistical errors are small with only a single electron per atom. There is no fixed-node error for an isolated H_2 molecule if the 2 electrons have different spins. Within the molecular phase, wavefunction nodes are primarily in the low density region external to the molecules. Hence we find the biases from the fixed-node approximation is very small [49]. We assume that the converged fixed-node QMC energies are the ground truth in the following analysis.

Fixed-node DMC (FN-DMC) calculations were done with QMCPACK version 3.9.2 [35, 36] using a Slater-Jastrow trial wavefunction with PBE orbitals computed using a Troullier-Martins PBE pseudopotential with core radius $r_c = 0.37a_{\text{bohr}}$ and a planewave cutoff of 200 Ry. The Jastrow functions contain two-body correlations as a sum of short-range and long-range contributions, both fully optimized for each configuration. The FN-DMC calculations used > 2000 walkers and were projected for 100 ha^{-1} with a timestep of 0.02 ha^{-1}. Canonical twist-averaging was performed using a $4 \times 4 \times 4$ shifted twist grid. We used the Chiesa force estimator [50] correcting the mixed-estimator bias with linear extrapolation. Finite-size corrections were applied to the total energy [34, 42] using the computed structure factor. In contradiction to claims in the literature [13], the computational effort is not substantially increased to compute forces using the Chiesa estimator since all that is required are modified electrostatic forces (regularized around the protons to avoid large fluctuations), quantities that are already computed for the gradient of the trial function. Forces are obtained from the twist-averaged electronic densities. No further finite-size corrections were applied to the forces, as the leading order energy correction is a function only of the mean density and thus does not affect the force.

We generated a large set of hydrogen configurations, all containing 96 protons, using several different simulation methods: CEIMC, AIMD with DFT forces, both classical simulations and path integral simulations. The temperatures of the simulations ranged from 300 K to 2200 K, the pressures from 50 GPa to 200 GPa. The configurations included mostly molecular hydrogen, but for the higher temperatures and pressures configurations with dissociated molecules were present. The lower temperature simulations were of

Figure 4. Histograms of the estimated values of standard deviations of the total energy (σ_V) and force components (σ_F) from 16 290 dense hydrogen configurations, each with 96 protons, within the pressure range 50 GPa–200 GPa and temperature range 300 K–2200 K. The standard deviations were estimated from the mean squared deviations of each individual DMC simulation. We obtained $\sigma_v = 10.8$ meV and $\sigma_F = 76.7$ meV Å^{-1}. A few percent of configurations with $\sigma_V > 0.014$ were discarded.

disoriented h.c.p. crystals. Above 1000 K most configurations were of liquid H_2 or liquid H. The forces for 100 000 configurations were calculated using DFT (both PBE and vdW-DF1 functionals). We calculated QMC forces and energies for more than 16 000 configurations chosen over a range of temperatures and pressures. The QMC calculations used about 200 times more processor time than the PBE-DFT calculations. The data and information about the temperatures and pressures resides in a public data base [51].

Histograms of the potential and force errors are shown in figure 4 as determined using a standard error analysis of the QMC calculations on each ionic configuration. Statistical analysis shows that the stochastic errors of the estimated forces and energies are close to normal justifying the use of the χ^2 loss function. There were a few ($\approx 2\%$) configurations which had estimated energy errors twice the average which we discarded pending investigation into the cause of the larger errors. The variation of the remaining errors is less than 10%. The error in the energy and force is only weakly dependent on the proton configuration within the range of pressures and temperatures studied and have mean standard errors of $\sigma_V = 10.8$ meV on the total energy and $\sigma_F = 76.6$ meV Å^{-1} on each force component. The mean value of the ionic forces within each configuration was subtracted from the individual ionic forces as was done in the silicon example.
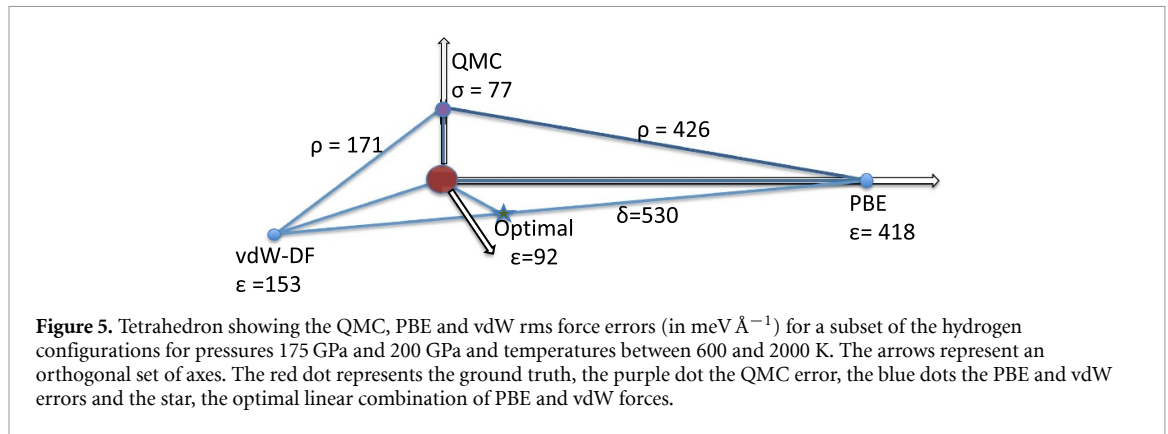
Using the QMC energies and forces we can make an estimate of the accuracy of DFT functionals and their error correlation. We have previously benchmarked DFT functionals on dense hydrogen [52, 53]. In that work, other methods were used to quantify the errors such as the mean absolute error. Using the property that the QMC errors are uncorrelated with the DFT errors, we can find an estimate of ϵ, the rms difference between DFT functionals and the ground truth, the noiseless QMC fixed-node energy; see table 3. We agree with the conclusion in our previous study: for dense molecular hydrogen, the vdW-DF functional [54] has errors about one third of that of the PBE functional [55]. The rms error of the total energy using the vdW functional is 1.24 eV. Because it is the variance of the energy fluctuations that is proportional to the size of the system, for a local energy error we divide by $\sqrt{n_I} = 9.8$. This gives a local energy error of 0.127 eV $= 1440$ K, reinforcing our conclusion that even vdW-DF is not accurate enough to predict the phase diagram of dense molecular hydrogen when energy scales of 1440 K are relevant. A diagram illustrating the magnitudes of the force errors in these 2 functionals is shown in figure 5. It shows that the PBE force errors are anticorrelated with the vdW errors, though with three times their magnitude, agreeing with our experience that PBE is a better description of the metallic-atomic phase and vdW of the insulating-molecular phase. By taking a linear combination of the two functionals (77% vdW, 23% PBE), one can get a significantly lower rms force error ($\epsilon_F = 92$ meV Å^{-1}) for this regime of temperature and pressure though with this combination the rms energy error is slightly higher.

6.1. DeepMD model

Our first ML model for dense hydrogen used a Δ-learning model containing three terms:

$$E = E_{\text{pair}} + \Delta E_{DFT} + \Delta E_{DMC} \qquad (23)$$

where E_{pair} is the contribution from an analytic potential using only the repulsive part of the exact BO energy of an isolated H_2 molecule [56]. This allows an accurate description of the internal motion of a single H_2 without having to learn it from the training data. The next two terms were determined by a ML model. ΔE_{DFT} is a model trained on DFT energies, forces, and stresses (either PBE or vdW functionals) minus the

Figure 5. Tetrahedron showing the QMC, PBE and vdW rms force errors (in meV Å^{-1}) for a subset of the hydrogen configurations for pressures 175 GPa and 200 GPa and temperatures between 600 and 2000 K. The arrows represent an orthogonal set of axes. The red dot represents the ground truth, the purple dot the QMC error, the blue dots the PBE and vdW errors and the star, the optimal linear combination of PBE and vdW forces.

pair potential contribution and was trained on roughly 100 000 hydrogen configurations. This model should capture the overall band structure and electron correlation effects as described by the density functional. Finally, ΔE_{DMC} is determined by fitting $E - E_{pair} - \Delta E_{DFT}$ to the subset of 20 000 configurations for which we have computed QMC forces. We expect ΔE_{DMC} to be a small correction: a smooth function of proton positions that can be described by a simpler ML model with fewer parameters, thus making good use of the computationally expensive and noisy QMC forces.

To model ΔE_{DFT} and ΔE_{DMC} we used the Deep Potential Molecular Dynamics (DeePMD) [4–6] version 1.3.3. Symmetry invariant feature vectors for atomic environments are constructed via the 'smooth edition' [4] descriptors with angular and radial information. The angular features are smoothly cut-off from 3.4 to 4.0 Å, and radial features from 5.1 to 6.0 Å. We used 8 neurons in the embedding layer, and 3 layers of 8 hidden neurons each for the descriptor network. The DeepMD training consisted of 10^6 iterations using 8 configurations to update the neural network parameters at each iteration. The starting and ending learning rates were set to 5×10^{-3} and 2×10^{-7}.

In [12] we used this model to perform very extensive MD and PIMD simulations over a wide range of temperatures and pressures, thereby determining solid-solid transitions, the melting temperature and making comparison with the experimental x-ray scattering determined structures.

6.2. MACE model

Our second optimization used the MACE model [47], an equivariant message passing model that uses higher-body order messages. We did not use Δ-learning model of equation (23) with MACE. Instead the QMC data were fit directly without a baseline or DFT layer. Optimization used 64 invariant messages, an angular resolution $l_{max} = 3$, and a cutoff radius of 5 Å. The loss function is the parameterized likelihood function with forces weighted with $\lambda = 3.5$.

6.3. Errors in the DeePMD and MACE models

Results in table 3 give the fitting error ρ (equation (9)) and cross-validation error δ (equation (16)) in the energies and forces for the two models used and for the clean DFT calculation. A visual depiction is shown in figure 6. We see for both models that $\rho > \sigma$. This means that the models have not exhausted the information in the QMC data, i.e. the values of χ^2 are large compared to the number of data points. In particular, the QMC energy uncertainty of 10 meV is two orders of magnitude smaller than the fitting error of the models to the QMC energies. (We remind the reader that what is quoted for σ_V is the standard deviation on the total energy of 96 protons in the supercell.) The energies are much worse in this regard than the forces[7]. The MACE model force errors are only 46% larger than the force QMC errors, so the force information from the QMC data has been reasonably well fit by the models. This indicates that more research is needed to find a more accurate model for dense hydrogen in this regime of temperature and pressure. We estimated the accuracy of the models with respect to the fixed-node energies and forces using equation (14) with $\eta = 0$.

The different performance of the energy and force errors is not understood. One might think that as a model approaches the exact potential, that the force and energy errors would decrease together. However, for the two DFT functionals $\epsilon_V/\epsilon_F = 8$ Å but for the 3 DeePMD models this ratio is about 4.5 Å while for the

---

[7] By decreasing the weight on forces compared to energy in the loss function, i.e. by decreasing the λ parameter, the energy error is reduced by a factor of 4 with an increase of force errors by less than 10%. A systematic study of these effects will be presented in a future work. For the current study neither the architecture nor λ were optimized.
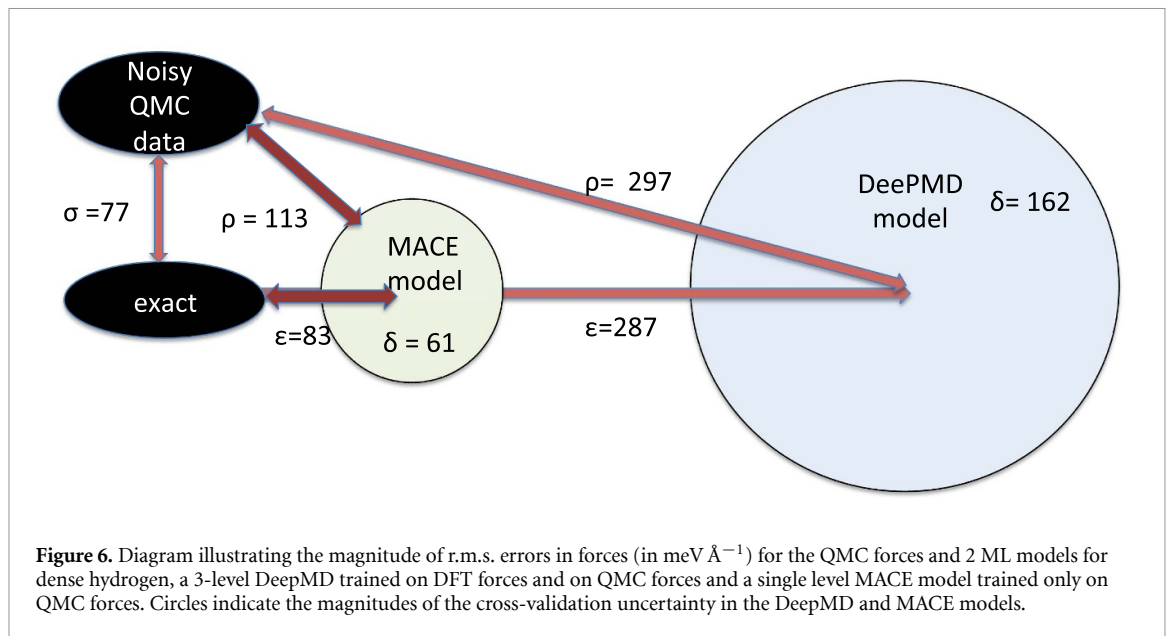
Figure 6. Diagram illustrating the magnitude of r.m.s. errors in forces (in meV Å^{-1}) for the QMC forces and 2 ML models for dense hydrogen, a 3-level DeepMD trained on DFT forces and on QMC forces and a single level MACE model trained only on QMC forces. Circles indicate the magnitudes of the cross-validation uncertainty in the DeepMD and MACE models.

Table 3. RMS errors in total energies (in meV) and forces (in meV Å^{-1}) with respect to QMC calculations using 16 290 configurations of 96 protons at pressures between 50GPa and 200 GPa. The first two columns abbreviate the construction of the models. For example 'PBE' means that the errors wrt QMC energies are compared to PBE energies, 'PBE/DeePMD' means that a DeePMD model is constructed using PBE forces and compared to QMC, and 'QMC/DeePMD' means a second level DeePMD model trained using QMC energies is added to the first level model. Symbols (ρ, ϵ, δ) are defined in figure 1. To compute ϵ we assumed $\eta = 0$ and used σ from figure 4. δ was estimated using a 5-fold cross-validation (jackknife) procedure.

1	2	ρ_V	ϵ_V	δ_V	ρ_F	ϵ_F	δ_F
PBE		3360	3360		426	418	
vdW		1240	1240		171	153	
PBE/DeePMD		1388	1388		306	296	
vdW/DeePMD		1412	1412		329	320	
vdW/DeePMD	QMC/DeePMD	1268	1268	312	297	287	162
QMC/MACE		1109	1109	226	113	83	61

MACE model it is 13 Å. Even though the forces drive the trajectory in MD, it is the potential that determines the equilibrium properties, for example the free energy.

There is a large uncertainty, δ, in the ML models as determined by the 5-fold cross validation also shown in figure 6. The uncertainty of the MACE model is less than half that of the DeePMD model, but it is of the same magnitude as the errors in the model forces. This indicates that the training would profit from more data; the existing ML models are still highly sensitive to the QMC noise and the selection of configurations.

The QMC/MACE model has the smallest errors in both the energy and forces from the other models including the clean vdW DFT model. However, the MACE model has not yet been tested with MD or PIMD simulations. It is not unusual for a MD or PIMD simulation to reveal hidden problems with an energy surface when a trajectory ventures into regions of configuration space that the model has not been trained for. It is clearly worthwhile to do such tests since the MACE scales better with the number of protons and is several orders of magnitude faster than using DFT functionals or Coupled Electron Ion MC (CEIMC) for studies of dense hydrogen.

7. Summary and future research

We have shown that it is feasible to use QMC forces to train state-of-the-art ML models. Such models trained on QMC energies and forces open the possibility to quantify the accuracy of models used in MC or MD studies. We have shown in the molecular hydrogen example, that the DeePMD and MACE models have not fully exploited the information contained in the QMC force and energy data.

Our analysis of the errors in the two analytic examples give some guidance on how to use QMC forces to improve ML models. Clearly one needs to calculate forces, not just energies because QMC forces have much more information to be exploited. It is better to have forces from many configurations even though the force errors would be larger than if fewer configurations had been used. A model is well converged if all of the important parameters are well trained. This can be judged by a cross-validation analysis. Models with fewer

parameters can become well-converged more easily. The ratios, ρ_V/σ_V and ρ_F/σ_F, can be used to judge whether models have extracted all the information from the data. These are some of the ways noisy data can help in validating ML models. They present great opportunities but there are many open research problems.

It is not clear which type of QMC will be most useful to generate data for ML: VMC, DMC, or RQMC? VMC can generate uncorrelated forces more efficiently. However, unless extensive careful optimization is done for each configuration, it can have a much larger systematic bias. DMC and RQMC are automatic procedures for eliminating much of this bias and should be equally accurate for the energy. A systematic examination of their efficiency and bias for forces has not been reported. We can also ask whether PIMC will be useful in the warm dense matter regime when the electronic excited states are populated.

There has been limited research on the efficiencies and biases of the suggested estimators for forces such as the Chiesa estimator [50] or zero-variance, zero-bias estimators [33]. Is extrapolation of the forces with DMC adequate or do we need to do forward walking, reptation or use unbiased estimators? What is systematic bias of the forces coming from estimators and from the fixed-node or fixed-phase approximations? Can second derivatives of the energy computed by QMC be useful to train the models? At the moment we do not know the statistical errors and biases of higher order derivatives.

An important practical question is how to speed up the QMC calculation so as to deliver data more efficiently. We found in the Silicon example that it is best to have as many points as possible even though the noise would be larger. If we minimize the start-up cost of a run, we can generate more data but we need robust but 'lightweight' QMC methods for this. It would be best to avoid a lengthy optimization of the trial function. Typically, QMC uses DFT orbitals for the trial wavefunction, however there are drawbacks to tying the QMC run to a fully converged DFT calculation as that would decrease its efficiency. One possibility is to develop a simplified single-particle theory, for example by using optimized effective interactions [24]. One gains with a compact expansion of the electron orbitals in terms of a basis, first, because determining the best expansion coefficients is more rapid, and, second, the evaluation of the trial wavefunction and its derivatives within the QMC calculation would be faster. Such a description is possible because the electron-ion Jastrow factor and backflow already contain the wavefunction cusps and the dynamic correlations of the electrons and do not need to be present in the orbitals. What is important is to find orbitals with a good representation of the nodal surfaces so that the fixed-node approximation is accurate. It has been found that analytic Jastrow functions, backflow functions and three-body functions can be used [57] and need little or no optimization. In CEIMC calculations of dense hydrogen, we made minor tweaks to the analytic functions for a given density and did not reoptimize for small changes in density or temperature [24].

What is the trade-off between high quality data and covering configuration space better? In the Tersoff Silicon example discussed above, we found that the best models come from the largest data set, even though the noise in forces was larger. What is the best way to select configurations? Can we formulate in terms of a sampling function F(R) or do learning procedures improve on this? For example, should one keep the configurations spaced out in configuration space.

What type of model should we to fit to? We have tried fitting directly to the QMC data and to fitting to a 3 level Δ-model containing molecular forces, DFT forces and QMC forces. It seems useful to start with a simple model that prevents atoms from getting too close to each other. But the MACE example for molecular hydrogen shows that hierarchical model may not be necessary. Having a model that needs accurate DFT calculated forces will make the data generation less efficient. Perhaps an embedded atom or tight-binding model will be a useful alternative baseline model. There is also an advantage to a model with fewer free parameters since having more parameters requires more QMC data for convergence. These issues will need careful examination on a many-body system.

Incorporating known physical constraints on the models can be important. We have already used short-ranged interactions such as the molecular H_2 binding curve in our model. However, long-ranged interactions are essential for calculations of the electrostatic properties such as the dielectric constant. But since forces on atoms are dominated by short-ranged interactions and ML models typically have a cutoff at a small multiple of the interatomic distance, long-range interactions may not be correctly determined in the ML model. In many cases, one is able to calculate parameters of the long-range interactions from first principles. For example, the dispersion interaction only depends on the multipole moments of the atoms or molecules present and their polarizabilities, and can be determined using QMC methods on small supercells. On the other hand in metals, the bare Coulomb interaction is screened by the mobile charges; models to describe the magnitude of the long-range interaction are more complex. A challenging problem is to simulate a metal–insulator transition such as occurs when hydrogen changes from the molecular phase to the atomic phase. The character of the long-range interaction changes during the transition and is correlated with the changes in the short-ranged structure of the system. It is important to be able to model the interactions at all distance scales accurately and self-consistently in order to make precise predictions of this transition using a ML model.

We have demonstrated that QMC is useful in generating data for ML potentials and assessing the error of those potentials. In principle, the data from QMC has a much smaller systematic bias. The noisy feature of QMC forces and energies can be an advantage both in allowing more complete sampling of phase space, in optimization of the model and assessing the model errors. There is much scope for improving the methodology to make QMC a useful tool for generating ML data.

Data availability statement

The data that support the findings of this study are openly available at the following URL/DOI: https://qmc-hamm.hub.yt/index.html.

Acknowledgments

ORCID iDs

David M Ceperley ● https://orcid.org/0000-0001-5082-6271
Scott Jensen ● https://orcid.org/0000-0002-6576-2204
Yubo Yang ● https://orcid.org/0000-0002-8800-9426
Carlo Pierleoni ● https://orcid.org/0000-0001-9188-3846

References

[1] Car R and Parrinello M 1985 *Phys. Rev. Lett.* **55** 2471–4
[2] Behler J and Parrinello M 2007 *Phys. Rev. Lett.* **98** 1–4
[3] Bartók A P, Payne M C, Kondor R and Csányi G 2010 *Phys. Rev. Lett.* **104** 136403
[4] Zhang L, Han J, Wang H, Saidi W A, Car R and Weinan E 2018 End-to-end symmetry preserving inter-atomic potential energy model for finite and extended systems *Proc. 32nd Int. Conf. on NeurIPS NIPS'18* (Curran Associates Inc.) pp 4441–51
[5] Zhang L, Han J, Wang H, Car R and Weinan E 2018 *Phys. Rev. Lett.* **120** 143001
[6] Han J, Zhang L, Car R and Weinan E 2018 *Commun. Comput. Phys.* **23** 629–39
[7] Jia W, Wang H, Chen M, Lu D, Lin L, Car R, Weinan E and Zhang L 2020 Pushing the limit of molecular dynamics with ab initio accuracy to 100 million atoms with machine learning *SC20: Int. Conf. for High Performance Computing, Networking, Storage and Analysis* pp 1–14
[8] Chmiela S, Tkatchenko A, Sauceda H E, Poltavsky I, Schütt K T and Müller K R 2017 *Sci. Adv.* **3** e1603015
[9] Alfé D, Bartók A P, Csányi G and Gillan M J 2013 *J. Chem. Phys.* **138** 221102
[10] Tirelli A, Tenti G, Nakano K and Sorella S 2022 *Phys. Rev. B* **106** L041105
[11] Tenti G, Tirelli A, Nakano K, Casula M and Sorella S 2023 arXiv:2301.03570
[12] Niu H, Yang Y, Jensen S, Holzmann M, Pierleoni C and Ceperley D M 2023 *Phys. Rev. Lett.* **130** 076102
[13] Huang C and Rubenstein B M 2023 *J. Phys. Chem.* A **127** 339–55
[14] Nakano K, Raghav A and Sorella S 2022 *J. Chem. Phys.* **156** 034101
[15] Cheng B, Mazzola G, Pickard C J and Ceriotti M 2020 *Nature* **585** 217–20
[16] Martin R M, Reining L and Ceperley D M 2016 *Interacting Electrons* (Cambridge University Press)
[17] Ceperley D, Chester G V and Kalos M H 1977 *Phys. Rev. B* **16** 3081–99
[18] Ruggeri M, Moroni S and Holzmann M 2018 *Phys. Rev. Lett.* **120** 205302
[19] Xie H, Zhang L and Wang L 2022 *J. Mach. Learn.* **1** 38–59
[20] Pescia G, Han J, Lovato A, Lu J and Carleo G 2022 *Phys. Rev. Res.* **4** 023138
[21] Li X, Li Z and Chen J 2022 *Nat. Commun.* **13** 7805
[22] Wilson M, Moroni S, Holzmann M, Gao N, Wudarski F, Vegge T and Bhowmik A 2023 *Phys. Rev. B* **107** 235139
[23] Xie H, Li Z-H, Wang H, Zhang L and Wang L 2023 *Phys. Rev. Lett.* **131** 126501
[24] Pierleoni C, Delaney K T, Morales M A, Ceperley D M and Holzmann M 2008 *Comput. Phys. Commun.* **179** 89–97
[25] Feynman R P 1939 *Phys. Rev.* **56** 340–3
[26] Ceperley D M and Alder B J 1980 *Phys. Rev. Lett.* **45** 566–9
[27] Ortiz G, Ceperley D M and Martin R M 1993 *Phys. Rev. Lett.* **71** 2777–80
[28] Ceperley D M and Alder B J 1984 *J. Chem. Phys.* **81** 5833–44
[29] Baroni S and Moroni S 1999 *Phys. Rev. Lett.* **82** 4745–8
[30] Pierleoni C and Ceperley D 2005 *ChemPhysChem* **6** 1872–8

[31] Ceperley D M 1996 Path integral Monte Carlo methods for fermions *Monte Carlo and Molecular Dynamics of Condensed Matter Systems* ed K Binder and G Ciccotti (Editrice Compositori)

[32] Zong F and Ceperley D M 1998 *Phys. Rev.* E **58** 5123–30

[33] Assaraf R and Caffarel M 2000 *J. Chem. Phys.* **113** 4028–34

[34] Chiesa S, Ceperley D M, Martin R M and Holzmann M 2006 *Phys. Rev. Lett.* **97** 076404

[35] Kim J *et al* 2018 *J. Phys.: Condens. Matter* **30** 195901

[36] Kent P R C *et al* 2020 *J. Chem. Phys.* **152** 174105

[37] Nakano K *et al* 2020 *J. Chem. Phys.* **152** 204121

[38] Needs R J, Towler M D, Drummond N D, Ríos P L and Trail J R 2020 *J. Chem. Phys.* **152** 154106

[39] Lin C, Zong F H and Ceperley D M 2001 *Phys. Rev.* E **64** 016702

[40] Li H, Xu Z, Taylor G, Studer C and Goldstein T 2018 Visualizing the loss landscape of neural nets *Advances in Neural Information Processing Systems* vol 31, ed S Bengio, H Wallach, H Larochelle, K Grauman, N Cesa-Bianchi and R Garnett (Curran Associates, Inc.)

[41] Motta M *et al* 2017 *Phys. Rev.* X **7** 031059

[42] Holzmann M, Clay R C, Morales M A, Tubman N M, Ceperley D M and Pierleoni C 2016 *Phys. Rev.* B **94** 035126

[43] Deringer V L, Bartok A P, Bernstein N, Wilkins D M, Ceriotti M and Csanyi G 2021 *Chem. Rev.* **121** 10073–141

[44] Press W H, Teukolsky S A, Vetterling W T and Flannery B P 2007 *Numerical Recipes 3rd edn: The Art of Scientific Computing* 3rd edn (Cambridge University Press)

[45] Tersoff J 1988 *Phys. Rev.* B **37** 6991–7000

[46] Thompson A P *et al* 2022 *Comput. Phys. Commun.* **271** 108171

[47] Batatia I, Péter Kovács D, Simm G N C, Ortner C and Csányi G 2022 arXiv:2206.07697

[48] McMahon J M, Morales M A, Pierleoni C and Ceperley D M 2012 *Rev. Mod. Phys.* **84** 1607–53

[49] Morales M A, Clay R, Pierleoni C and Ceperley D M 2014 *Entropy* **16** 287–321

[50] Chiesa S, Ceperley D M and Zhang S 2005 *Phys. Rev. Lett.* **94** 036404

[51] Yang Y, Jensen S, Ceperley D M, Kowalik K and Turk M 2022 yt hub qmc data browser (available at: https://qmc-hamm.hub.yt/index.html)

[52] Clay R C, Mcminis J, McMahon J M, Pierleoni C, Ceperley D M and Morales M A 2014 *Phys. Rev.* B **89** 184106

[53] Clay R C, Holzmann M, Ceperley D M and Morales M A 2016 *Phys. Rev.* B **93** 035121

[54] Dion M, Rydberg H, Schröder E, Langreth D C and Lundqvist B I 2004 *Phys. Rev. Lett.* **92** 246401

[55] Perdew J P, Burke K and Ernzerhof M 1996 *Phys. Rev. Lett.* **77** 3865–8

[56] Kołos W and Wolniewicz L 1964 *J. Chem. Phys.* **41** 3674–8

[57] Holzmann M, Ceperley D M, Pierleoni C and Esler K 2003 *Phys. Rev.* E **68** 046707