

RESEARCH ARTICLE | NOVEMBER 17 2022

Extending the reach of quantum computing for materials science with machine learning potentials

Special Collection: [2022 Machine Learning](#)

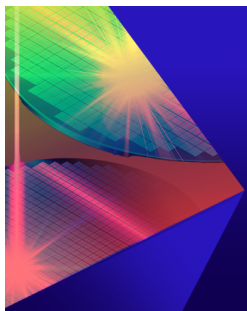
Julian Schuhmacher  ; Guglielmo Mazzola  ; Francesco Tacchino  ; Olga Dmitriyeva; Tai Bui; Shanshan Huang; Ivano Tavernelli  




AIP Advances 12, 115321 (2022)
<https://doi.org/10.1063/5.0099469>



CrossMark



AIP Advances
Special Topic: Novel Applications of Focused Ion Beams — Beyond Milling
Submit Today



Extending the reach of quantum computing for materials science with machine learning potentials

Cite as: AIP Advances 12, 115321 (2022); doi: 10.1063/5.0099469

Submitted: 17 May 2022 • Accepted: 26 October 2022 •

Published Online: 17 November 2022



View Online



Export Citation



CrossMark

Julian Schuhmacher,^{1,2}  Guglielmo Mazzola,¹  Francesco Tacchino,¹  Olga Dmitriyeva,³
Tai Bui,⁴ Shanshan Huang,⁴ and Ivano Tavernelli^{1,a)} 

AFFILIATIONS

¹ IBM Quantum, IBM Research–Zurich, 8803 Rüschlikon, Switzerland

² Institute for Theoretical Physics, ETH Zürich, 8093 Zürich, Switzerland

³ IBM Quantum, Yorktown Heights, New York 10598, USA

⁴ Applied Sciences, Innovation and Engineering, BP International Ltd., London, United Kingdom

^{a)} Author to whom correspondence should be addressed: ita@zurich.ibm.com

ABSTRACT

Solving electronic structure problems represents a promising field of applications for quantum computers. Currently, much effort is spent in devising and optimizing quantum algorithms for near-term quantum processors, with the aim of outperforming classical counterparts on selected problem instances using limited quantum resources. These methods are still expected to feature a runtime preventing quantum simulations of large scale and bulk systems. In this work, we propose a strategy to extend the scope of quantum computational methods to large scale simulations using a machine learning potential trained on quantum simulation data. The challenge of applying machine learning potentials in today's quantum setting arises from the several sources of noise affecting the quantum computations of electronic energies and forces. We investigate the trainability of a machine learning potential selecting various sources of noise: statistical, optimization, and hardware noise. Finally, we construct the first machine learning potential from data computed on actual IBM Quantum processors for a hydrogen molecule. This already would allow us to perform arbitrarily long and stable molecular dynamics simulations, outperforming all current quantum approaches to molecular dynamics and structure optimization.

© 2022 Author(s). All article content, except where otherwise noted, is licensed under a Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>). <https://doi.org/10.1063/5.0099469>

I. INTRODUCTION

Quantum computers have the potential to revolutionize the field of quantum chemistry by providing significant speedups in the solution of electronic structure problems.^{1–4} Substantial experimental and theoretical advances have been made in the last few years, both concerning the realization of quantum computing platforms^{5–8} and the development of new generations of quantum algorithms.^{9–12}

Current quantum processors are subject to different forms of hardware noise, such as finite gate fidelity and qubit coherence times, which limit the size and depth of the quantum circuits that can reliably be executed. As a consequence, it is generally believed that the most powerful but resource-demanding algorithms for

quantum chemistry calculations, such as those based on Quantum Phase Estimation (QPE), will only become accessible in combination with error correction protocols in the so-called fault-tolerant regime. At present, the most viable alternative options—spearheaded by the well-known variational quantum eigensolver (VQE)^{13,14}—can only offer noisy sampling access to families of quantum states prepared via shallow circuits. This introduces the further complication of statistical uncertainties on the reconstruction of the relevant physical observables. Despite these limitations and with the help of advanced readout and error mitigation techniques, noisy quantum computers could still challenge classical counterparts by effectively tackling intermediate-sized molecular systems.

In this regime, the most affordable representations of the electronic Hamiltonian feature a moderate yet challenging polynomial

number of terms: for example, in the second quantization formulation using Gaussian basis sets, this scales as $O(M^4)$, where M is the number of basis functions used to describe the system.⁴ This scaling is then reflected in most quantum algorithms that try to solve for the ground state of the Hamiltonian. While there exist other representations of the Hamiltonian, for example, with different basis sets (e.g., plane waves), which leads to a better asymptotic scaling, they require a larger number of basis functions for small systems, which renders them unpractical to implement on near-term quantum devices.¹⁵

Such asymptotic scaling of $O(M^4)$ may still represent a substantial practical barrier for simulations of extended systems. This is particularly significant in view of possible quantum-powered molecular dynamics (MD) simulations. In fact, even typical classical approximate electronic solvers, such as Density Function Theory (DFT)¹⁶ or Quantum Monte Carlo^{17,18}—featuring a $O(N^{2-3})$ scaling in terms of the number N of electrons in the system—can only allow for system sizes [$O(10^3)$ electrons] and timescales (few picoseconds) that are often not sufficient to study many physical phenomena, such as chemical reactions in solutions, nucleation processes, or phase transitions. If one also takes into account typical gate speeds of current quantum computers,^{19,20} it seems unlikely that quantum electronic structure calculations will directly achieve large scale simulations of, e.g., bulk systems. Similar conclusions also remain essentially valid for future QPE-based protocols.

Recently, machine learning (ML) approaches have been put forward to overcome such size and timescale barriers in first-principle simulations driven by DFT.^{21–25} The last few years witnessed an exceptional increase in quantity and quality of ML-powered numerical experiments such that this approach is on the path to become a standard in materials science.^{26–29}

In this article, we support the idea that first-principle quantum methods should follow the same approach, i.e., that data coming from quantum hardware should be used to harvest electronic structure datasets to generate a machine learning potential (MLP), rather than to drive an MD trajectory directly. However, the combination between quantum electronic structure calculations and MLPs is less straightforward compared to the case where the dataset is generated through DFT calculations due to the different forms of noise that currently affect quantum computation, degrading the quality of the corresponding training labels.

We focus on a standard implementation of the VQE algorithm, for which we identify three main sources of errors: statistical (measurement) noise, variational optimization noise, and hardware noise. For each of these sources, we assess the impact on the reference labels in a dataset and on the quality of the resulting MLP trained on this dataset. The size of the systems for which we can simulate the different noise sources (or running hardware experiments) is constrained, which limits most of the simulations to single molecule systems. While small systems—such as single water or hydrogen molecules—in principle, do not require powerful machine learning models, such as an MLP, our investigation is aimed at identifying the general behavior of MLPs trained on noisy reference labels generated with quantum computers. In this way, we derive a set of general principles that we expect to also observe in more complex systems.

It is also worth noting that our considerations may also apply to a more general class of quantum algorithms featuring variable

combinations of hardware noise (closely related to circuit depth) and sample complexity, which can be traded for one another.^{30,31}

This article is structured as follows. Section II reviews the general idea of MLPs. Section III introduces the general concepts for quantum electronic structure calculations. In Sec. IV, we discuss the effect of the different noise sources on quantum computations. Section V provides a brief description of the applied neural-network force-field approach. We present and discuss the results in Sec. VI and conclude in Sec. VII.

II. MACHINE LEARNING POTENTIALS

Machine learning approaches are the cornerstone in many technological applications, ranging from image/speech recognition, search and recommendation engines, to data detection and filtering.³²

While in the past, ML methods, due to their power of compressing high-dimensional data into low-dimensional representations,^{33,34} have been mostly applied to data science, we have recently witnessed an increased interest for applications in the physical sciences and particularly in quantum mechanics.^{25,35} For instance, several ML methods have been put forward to solve the many-body Schrödinger equation (in a reinforcement learning fashion),^{36–39} to learn quantum states from measurements (unsupervised learning),^{40–43} or to learn materials or chemical properties from datasets (supervised learning).^{25,26} The general idea in these approaches is to search large databases for non-trivial relationships between the molecular or crystal structures (i.e., atomic positions and nuclear charges $\{\mathbf{R}, \mathbf{Z}\}$) and several properties of interest. These include, for instance, semiconductor's bandgaps and dielectric constants,⁴⁴ atomization energies of organic molecules,^{45,46} energy formations of crystals,⁴⁷ or the thermodynamic stability of solids and liquids.^{48,49} To do so, one first performs the training of the ML algorithm on a finite subset of known solutions ($\{\mathbf{R}, \mathbf{Z}\} \rightarrow p$) and then predicts the properties p of interest for new, unseen structures, which differ in composition and geometry.

The construction of MLPs, first pioneered by Blank *et al.*⁵⁰ and later by Behler and Parrinello,^{24,51} falls within this class.

In essence, the approach works as follows: (i) one generates a training dataset of M configurations $\{\mathbf{R}, \mathbf{Z}\}$. Each sample contains atomic positions (the number of atoms in the sample can be limited, provided that interactions are local in space), charges, and the energy E (or forces \mathbf{f}) calculated with an *ab initio* electronic structure method, such as DFT. (ii) The learning process consists of generalizing the mapping $\{\mathbf{R}, \mathbf{Z}\} \rightarrow E$ to out-of-sample configurations and bypassing the need for solving the electronic structure problem at each MD iteration, thus achieving a considerable speedup in simulations.

As a result, first-principle modeling can now reach size and time scales, which were previously possible only with computationally cost-effective, but approximated, empirical force-fields. The technique has already been applied to several long-standing problems in materials modeling, such as the phase diagram of liquid and solid water,^{28,49} silicon,^{29,52} and dense hydrogen,²⁷ to name a few.

In this work, we adopt the MLP approach based on a neural-network architecture,⁵¹ as implemented in the software package n2p2 (version: 2.1.1),⁵³ in combination with a training based on a quantum computing evaluation of the electronic structure. Details

of the setup will be outlined in Sec. V. Before moving already to this rather technical part, let us first discuss the quantum computing aspect of this work.

III. QUANTUM COMPUTING FOR ELECTRONIC STRUCTURE PROBLEMS

The starting point of most electronic structure problems in chemistry or materials science is the electronic Hamiltonian written in the second quantized representation,^{3,4,54}

$$\hat{H}(\mathbf{R}) = \sum_{rs} h_{rs}(\mathbf{R}) \hat{a}_r^\dagger \hat{a}_s + \frac{1}{2} \sum_{pqrs} g_{pqrs}(\mathbf{R}) \hat{a}_p^\dagger \hat{a}_q^\dagger \hat{a}_r \hat{a}_s + E_{nn}(\mathbf{R}), \quad (1)$$

with $h_{rs}(\mathbf{R})$ denoting the one-electron integrals and $g_{pqrs}(\mathbf{R})$ denoting the two-electron integrals, respectively. The vector of nuclear coordinates $\mathbf{R} = (\mathbf{R}_1, \mathbf{R}_2, \dots, \mathbf{R}_{N_I}) \in \mathbb{R}^{3N_I}$ of N_I nuclei parameterizes the electronic Hamiltonian. The operators \hat{a}_r^\dagger (\hat{a}_r) represent the fermionic creation (annihilation) operators for electrons in N molecular spin-orbitals (MOs). The term $E_{nn}(\mathbf{R})$ represents the classical nuclear repulsion energy.

The implementation of Eq. (1) requires the translation of each fermionic operator into a qubit operator, which can be interpreted by using a quantum computer. This can be achieved by several fermion-to-qubit mappings, such as the Jordan–Wigner or the Bravyi–Kitaev mapping (we refer to standard reviews, such as Refs. 3, 4, and 54 for more details). After this mapping, the Hamiltonian operator has the following form:

$$\hat{H} = \sum_{k=1}^K c_k \hat{P}_k, \quad c_k \in \mathbb{R}, \quad (2)$$

where each N -qubit Pauli string \hat{P}_k is an element of the set $\mathcal{P}_N = \{\hat{p}_1 \otimes \hat{p}_2 \otimes \dots \otimes \hat{p}_N \mid \hat{p}_i \in \{\hat{I}, \hat{X}, \hat{Y}, \hat{Z}\}\}$ (tensor products of N single qubit Pauli operators). As discussed above, the total number of Pauli terms scales as $\mathcal{O}(N^4)$.

There exist several methods to solve for the ground state of Eq. (1) using a quantum computer. The method of choice, when fault-tolerant quantum computers will be available, is to perform quantum phase estimation to project onto eigenstates of the Hamiltonian.^{4,55,56} Another strategy is to obtain a variational approximation of the ground state using the variational quantum eigensolver (VQE).^{4,13} This heuristic method features parameterized quantum circuits, defined in terms of parametric gates. This generates a variational quantum state $|\Psi(\boldsymbol{\theta})\rangle$, often called *trial state*, defined by the array of parameters $\boldsymbol{\theta}$. The parameters are then optimized *classically* to reach the minimum for the energy,

$$E(\boldsymbol{\theta}) = \langle \Psi(\boldsymbol{\theta}) | \hat{H}(\mathbf{R}) | \Psi(\boldsymbol{\theta}) \rangle. \quad (3)$$

The expectation value in Eq. (3) is calculated as the sum of the expectation values $\langle \hat{P}_k \rangle$ of the single Pauli operators, multiplied by the respective scalar coefficients c_k . Each $\langle \hat{P}_k \rangle$ value is obtained through sampling from the prepared state $|\Psi\rangle$ using S_k measurements, hence S_k repetitions of the same circuit (see Ref. 58 for details). The statistical error associated with the evaluation of $\langle \hat{P}_k \rangle$ decreases as $1/\sqrt{S_k}$.

Finally, to construct a training dataset using the VQE algorithm, one just needs to create the second quantized Hamiltonian Eq. (1) for a set of generated atomic structures $\{\mathbf{R}, \mathbf{Z}\}$, perform the fermion-to-qubit mapping to generate the qubit Hamiltonian Eq. (2), and finally optimize a parameterized circuit $|\Psi(\boldsymbol{\theta}(\{\mathbf{R}, \mathbf{Z}\}))\rangle$ to obtain the energies of the atomic structures.

IV. ERROR SOURCES AND TRAINING WITH NOISY DATASETS

The combination of VQE with ML for force-field generation would be trivial if not for the presence of error sources that are absent in classical DFT calculations. These errors pertain only to the quantum nature of the hardware. While they can be systematically reduced, some of them will remain finite, assuming practical setups.

In this article, we consider three main noise sources that can affect the quality of the dataset generated with a quantum computer.

A. Statistical noise in expectation values

This type of noise is linked with the way observables are computed in the quantum setting. As discussed above for the case of the energy, the expectation value of an operator is computed as the sum of the expectation values of its Pauli terms. The variance becomes

$$\text{Var}[H] = \sum_{k=1}^K |c_k|^2 \text{Var}[\hat{P}_k], \quad (4)$$

where $\text{Var}[\hat{P}_k] = \langle \hat{P}_k^2 \rangle - \langle \hat{P}_k \rangle^2$ is the variance of the Pauli string \hat{P}_k . It is easy to see that the variance is always finite even if we consider the exact ground state of H . Since the ground state is an eigenstate of the sum, but not of each single Pauli operator P_k , the total variance is always positive.⁵⁷ The error in the estimation is, therefore, given by

$$\epsilon_{\text{stat}} = \sqrt{\sum_k |c_k|^2 \text{Var}(\hat{P}_k) / S_k}, \quad (5)$$

where S_k is the number of measurements, used to estimate the k th term, with $\sum_{k=1}^K S_k = M$, and M is the total number of measurements. For instance, for an eight qubit Hamiltonian operator representing the H_2 molecule at equilibrium bond distance in the 6–31g basis set, the number of shots M required to compute the energy within chemical accuracy (1.6 mHa) is on the order of 10^8 .⁴³

As of today, many strategies have been put forward to at least mitigate this issue.^{43,58–67} To the best of our knowledge, these methods can save at most three orders of magnitude in the number of shots,⁴³ but cannot entirely remove the problem.

Without loss of generality, we can, therefore, assume that the expectation value of an operator O decomposed as a sum of Pauli terms [Eq. (2)] will always take the form

$$\langle O \rangle = \langle O \rangle_{\text{true}} + \epsilon_{\text{stat}} \quad (6)$$

even if the exact ground state can be represented by the quantum circuit. The operators of our interest are the energy E and the set of N_I atomic forces $\mathbf{F} = (\mathbf{F}_1, \mathbf{F}_2, \dots, \mathbf{F}_{N_I})$, which can also be

decomposed into Pauli strings and measured alongside the energy.⁶⁸ This labeling error needs to be taken into account when training an ML model.

Finally, we note that statistical errors in the expectation values of energy and forces are also present in some classical electronic solvers, such as quantum Monte Carlo.⁶⁹

B. Variational and optimization error

The second type of error we consider is the variational error. This happens because the exact ground state generally lies outside the region of the Hilbert state that can be represented by the variational ansatz.

So far, different types of circuits have been employed in quantum computing calculations. They can be roughly divided into two classes. The first class contains *chemically inspired* circuits. These circuits generally feature few variational parameters, but a fairly large depth, and therefore, these are still unsuited for present day's hardware. The most popular of them is the unitary coupled cluster (UCC) circuit,⁴ with a depth growing as $\mathcal{O}(N^4)$, in the commonly used version where the excitations are truncated at the second order (UCCSD). Alternatives for chemically inspired ansätze would be the Adaptive Derivative-Assembled Pseudo-Trotter (ADAPT)-VQE⁷⁰ and k-UpCCGSD,⁷¹ which tend to have a better performance than UCC on strongly correlated systems.⁷² However, due to simplicity, we decided to use the UCCSD ansatz.

The second class consists of *hardware efficient* ansätze.¹⁴ These circuits prepare entangled states while minimizing the circuit depth. They usually feature many more variational parameters; therefore, they offload part of the computational burden to the classical optimizer.

Indeed, even in the case when the exact ground state, or a close approximation of it, is theoretically within the representability range of the ansatz, suboptimal energy minimization can lead to poor results. Equation (6) is thus modified as

$$\langle O \rangle = \langle O \rangle_{\text{true}} + \epsilon_{\text{stat}} + \epsilon_{\text{var}}, \quad (7)$$

where ϵ_{var} is the error coming from a non-ideal variational optimization. In this work, we will study how this error depends on the chosen circuit and how it impacts the training of an MLP.

C. Hardware noise

In the era of noisy quantum devices, errors occur in the execution of a quantum circuit on actual quantum processors. As a result, datasets prepared via quantum computing methods will be affected by inaccuracies even in the ideal case of a perfect choice of the ansatz (see Sec. IV B). It is, therefore, important to assess the possibility of successfully training a good MLP even in the presence of these effects. Incoherent errors and readout noise, which may increase fluctuations and the bias in the energy evaluations and can even hinder the optimization of VQE ansätze,⁷³ are particularly important in this context.

Errors belonging to the first class, namely, incoherent noise, are primarily due to unwanted and uncontrolled interactions between qubits and their environment throughout the whole computation. These formally translate into finite relaxation and coherence times, named T_1 and T_2 , respectively, which essentially correspond to amplitude and phase damping effects.

Readout errors instead affect the qubit measurement process: these may be modeled as bit flip channels, which stochastically produce erroneous assignments, while the state of a qubit is being probed.

Finally, coherent errors may also arise in the implementation of single- and two-qubit logic gates primarily due to imperfect device calibration and manipulation. These typically result in systematic infidelities of the individual operations.

Two observations are in order. On the one hand, it would seem cautious to expect that standard ML techniques will not be able by themselves to compensate for hardware noise, unless specifically designed for this purpose:^{74–76} as a result, a minimal well posed target is to show the trainability of an MLP up to an overall model error—with respect to noiseless exact values—matching as close as possible the characteristic inaccuracy induced by noise on the training points. On the other hand, one should also keep in mind that fast technological advancements, possibly in combination with error mitigation techniques,^{62,77–85} will progressively reduce the impact of hardware noise. It is therefore interesting to investigate the possible improvements that ML generated potentials could enjoy in the future, showing that their quality could closely follow the increased accuracy of the available datasets.

V. NEURAL-NETWORK FORCE FIELDS

In this paper, we adopt the high-dimensional neural-network potential (HDNNP) architecture of Behler and Parinello⁵¹ for the machine learning potential (MLP). For the general motivation and the description of this ML model, we refer to a review by Behler.²⁴ Here, we provide a detailed discussion of some non-trivial aspects of the architecture, which are also important to reproduce our results.

We use the following procedure for the training of an MLP. (i) Prepare a training dataset and a validation dataset. This should be done with VQE as explained in Sec. III. (ii) Fix the neural-network architecture (neural-network geometry, learning parameters, and symmetry functions) (iii) Train an MLP using the training dataset. (iv) Evaluate the MLP on the validation dataset. (v) Repeat steps (ii)–(iv) for different sets of hyperparameters. (vi) Choose the MLP with the lowest prediction error on the validation dataset.

The prediction error is measured in terms of the *root-mean-square error* (RMSE), which is defined for the energy (E) as

$$\text{RMSE}(E) = \sqrt{\frac{1}{N_s} \sum_{i=1}^{N_s} (E_{\text{MLP}}^i - E_{\text{Ref}}^i)^2} \quad (8)$$

and for the forces (F) as

$$\text{RMSE}(F) = \sqrt{\frac{1}{N_s} \sum_{i=1}^{N_s} \frac{1}{3N_a^i} \sum_{j=1}^{N_a^i} (\mathbf{F}_{\text{MLP},j}^i - \mathbf{F}_{\text{Ref},j}^i)^2}, \quad (9)$$

where the explicit dependence of N_a^i on the sample index i comes from the fact that, in general, the dataset contains structures with different numbers of atoms. Note that the dataset labels are normalized as explained in Appendix A.

Other crucial ingredients for this procedure are the *symmetry functions*. These are many-body functions that capture, in a

compact fashion, the structural information in the local environment of an atom. The symmetry function values are the real inputs for the NN, instead of the raw Cartesian coordinates of the atoms. The main motivation behind this choice is that translational and rotational invariance can be easily implemented.²⁴

In this work, we adopt the so-called G_2 and G_3 symmetry function classes. The first is a family of *radial* symmetry functions made of two-body terms, while the second also contains three-body terms, which are needed to encode the tridimensional structure of an atomic configuration. We provide in [Appendix B](#) the explicit functional form of these functions, as well as other details needed to reproduce our settings.

It is important to note that one would like to avoid redundancies in the symmetry function set. In this work, we first define a set of candidate symmetry functions and then select a small subset that still enables us to capture the structural information of a given dataset. To this end, we adopt the automatic selection of symmetry functions as proposed by Imbalzano *et al.*,⁸⁶ which is detailed in [Appendix C](#).

However, in the case of the bulk systems of [Sec. VI A](#), which have already been studied in [Refs. 27 and 87](#), we adopt the symmetry functions already used in the respective publications.

VI. RESULTS AND DISCUSSION

We now present the results of the proposed HDNNP approach trained with electronic structure calculations performed with a quantum algorithm and affected by typical noise sources compatible with near-term quantum computers. We proceed systematically by analyzing the impact of each different noise source on the quality of the predictions for a series of model systems. For the statistical error analysis, we start with the study of the effect of a Gaussian distributed noise model on the energies of forces evaluated for liquid and solid water. We then proceed to the investigation of a smaller system, namely, a single water molecule, which can be implemented on today's quantum devices and for which a resource assessment is possible. Finally, we validate our approach for the case of the H_2-H_2 cluster, where the sampling of intermolecular distances and orientations is required. The analysis of the impact of the optimization errors on the quality of the HDNNP predictions is investigated for the same water molecule system introduced above using different wavefunction ansätze. Finally, the effect of hardware noise is investigated on the simpler molecular system, namely, H_2 , for which we can efficiently perform the required sampling of the intramolecular distance both in simulations and in hardware experiments. Details on the simulation parameters and system setups will be introduced in order of appearance in the following sections [VI A–VI C](#).

A. Resilience against statistical noise

1. A bulk system example: Liquid and solid water

The first of our assessments concerns the trainability of an MLP in the presence of the statistical noise alone (see [Sec. IV A](#)). This study can be performed already in a prototypical bulk system, which is the end goal of the whole technique. Indeed, the statistical noise in the labels of the training dataset can be easily and rigorously *emulated* by adding a Gaussian distributed random variable with zero mean. For each structure in the training dataset, the reference energy and forces are modified according to

$$E \rightarrow E + \mathcal{N}(0, \Delta_E), \quad (10)$$

$$F_i^\mu \rightarrow F_i^\mu + \mathcal{N}(0, \Delta_F), \quad (11)$$

where E is the energy of the structure and F_i^μ is the force corresponding to atom i and component $\mu = \{x, y, z\}$. Δ_E and Δ_F correspond to the variance of the statistical noise that is introduced for the energy and the forces, respectively.

In this study, we consider a bulk water system. The dataset is taken from [Ref. 88](#) and contains 7241 configurations of ice and liquid water. The energies and forces were calculated with DFT using the Revised Perdew-Burke-Ernzerhof (RPBE) functional⁸⁹ with D3 corrections.⁹⁰ The mean energy in the dataset is -694.47 eV/atom with a standard deviation of 0.11 eV/atom. The standard deviation of the forces is 1.225 eV/Å. Here, we follow the reasonable assumption that the potential energy surface obtained with a DFT model is in qualitative agreement with the exact one and that the remaining difference does not play any role in this particular assessment concerning the learnability of an MLP from a noisy dataset.

We then use the noisy training datasets to fit the MLPs and a noiseless validation dataset to assess their accuracy. The amount of noise in the energies and the forces is varied independently. The values considered for the energy noise are $\{1000, 100, 10, 1, 0.1\}$ meV/atom. For the force noise, the values $\{10, 1, 0.1, 0.01, 0.001\}$ eV/Å are used. As a reference, the training with no noise in the energy and/or the force labels is also considered. The symmetry functions are taken from [Ref. 53](#). The resulting prediction RMSE values of the trained MLPs are shown in [Fig. 1](#).

First of all, we observe that the prediction accuracy for the training on the dataset without noise (cell in the top right corner in the plots of [Fig. 1](#)) is consistent with [Ref. 91](#) (0.7 meV/atom for the energy and 0.036 eV/Å for the forces). Most importantly, we note that there exists a limiting value of Δ_E and Δ_F below which the prediction accuracy is as low as in the noiseless case. The important consequence is that one is not forced to reduce the statistical error bars in the dataset to *zero*, enabling, in principle, a practical implementation of the method. Additionally, we would like to highlight that the MLPs do a remarkable job at reducing the statistical error even in the region where the RMSE is not saturated. We can, for example, look at the two leftmost columns in the panel on the left in [Fig. 1](#). The MLP is able to reduce the error from 1000 to 60–70 meV/atom and from 100 to 15–20 meV/atom. However, as already observed above, this ability to reduce the statistical fluctuations is saturated at the point where the error is solely due to the learning ability of the MLP itself.

2. A single water molecule

The goal of this section is to introduce a smaller system, a single water molecule, for which a quantum resource assessment is feasible. We will translate the error threshold Δ_E into a quantum measurement resource estimate. The single molecule configurations are extracted from the bulk water dataset used in [Sec. VI A 1](#).

We first repeat the above assessment using simulated noise. Concerning the training, we select 20 symmetry functions for hydrogen and 15 symmetry functions for oxygen with the CUR feature selection method⁸⁶ (see [Appendix C](#)). In this case, we also consider the possibility to train the MLP *without* the use of the forces. In this case, it is also interesting to assess the dependence of the RMSE on

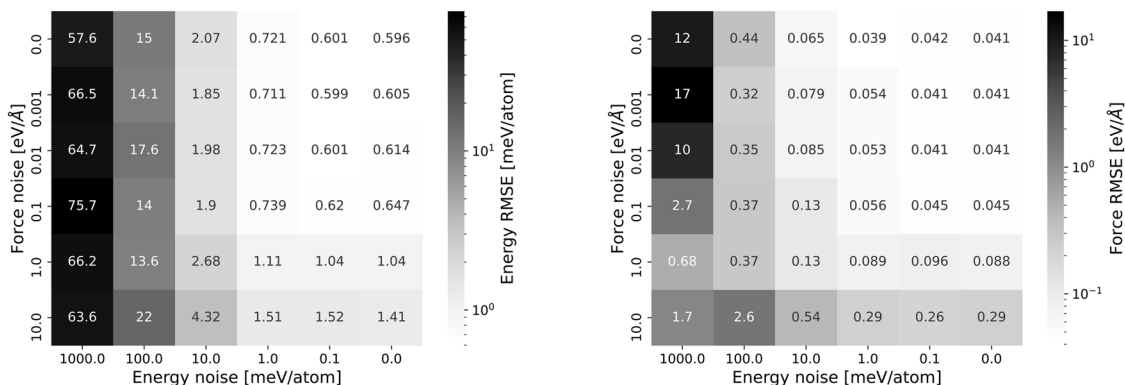


FIG. 1. Bulk water system: RMSE of the MLPs trained using noisy labels. **Left:** Energy RMSE as a function of the noise level (see the text) affecting the energy labels (x axis) and the force labels (y axis) in the training dataset. The standard noiseless case corresponds to the top-right entry. **Right:** Same assessment but targeting the MLP force RMSE.

the dataset size. The results of both, the training with and without using the forces, are shown in Fig. 2. We qualitatively observe the same behavior as for the bulk water.

In view of the non-trivial computational cost of an electronic structure calculation on a quantum computer, we aim to reduce the number of configurations in the training and validation dataset as much as possible using the CUR decomposition.⁸⁶ In Fig. 2 (right panel), we observe that we can reduce this number down to 100 configurations in the training set without a noticeable increase in the RMSE.

In terms of the dependence on the energy noise, the behavior of the RMSE in the training with and without using the forces is qualitatively the same. However, there is a small advantage if the forces are used in the training. The calculation of the forces with VQE was already proposed by Sokolov *et al.*⁶⁸ However, due to technical limitations and the fact that the training without the forces is possible, we focus on the energy noise threshold in the following discussion.

The next step is to assess the number of shots M (see Sec. IV A) required to achieve the desired accuracy. The noise level threshold where the RMSE exits the region of low errors lies between 10 and 1 meV/atom. In the following, we will calculate a lower and an upper bound on the number of shots to reach this threshold. The following evaluation will demonstrate the estimation of the lower bound (noise level 10 meV/atom); the upper bound can be estimated analogously by replacing the noise level with 1 meV/atom.

To estimate the number of total measurements M to reach a certain accuracy \mathcal{E} in the energy estimation of an N -qubit Hamiltonian, we consider the probability $p(\delta < \mathcal{E})$ that the deviation of the energy estimate to the ground state energy E_0 is smaller than the desired accuracy. Following Ref. 43, this probability is given by

$$p(\delta < \mathcal{E}) = \text{Erf}\left(\frac{\mathcal{E}\sqrt{S/2\sigma^2[H]}}{\delta}\right), \quad (12)$$

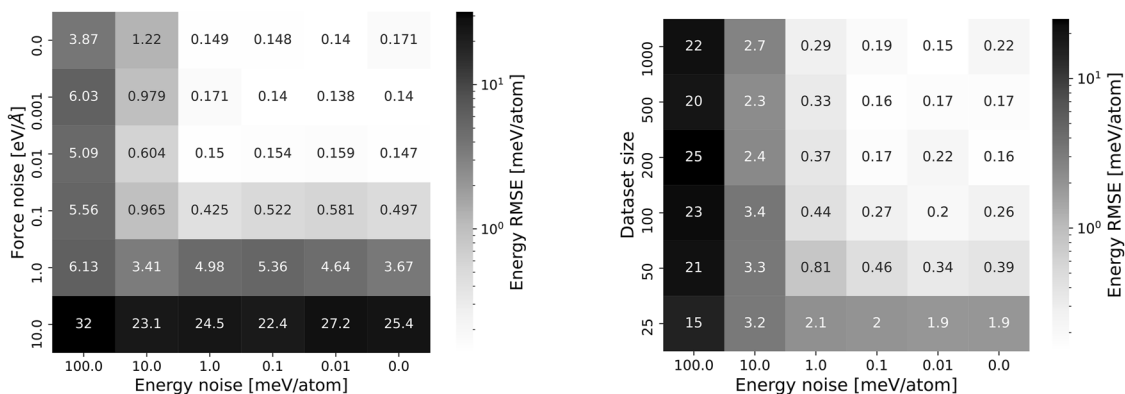


FIG. 2. Single water molecule: Energy RMSE of the MLPs trained using noisy labels and different training dataset sizes. **Left:** Energy RMSE as a function of the noise level (see the text) affecting the energy labels (x axis) and the force labels (y axis) in the training dataset. Both the energy and force labels are used for the training. The training dataset contains 100 single water molecule structures. **Right:** Energy RMSE as a function of the noise level (see the text) affecting the energy labels (x axis) and the number of structures in the training dataset (y axis). Only the energy labels are used for the training.

where $S = M/K$ is the number of measurements per Pauli operator \hat{P}_k and $\sigma^2[H]$ is the measurement variance of the Hamiltonian. The estimate for the total number of measurements is then given by the number of measurements that is required to reach $p(\delta < \mathcal{E}) \approx 1$.

A loose upper bound of the resource estimation p_{\max} can be obtained by determining the variance in the equation above with

$$\sigma_{\max}^2[H] = \left(\sum_k |c_k| \right)^2, \quad (13)$$

where c_k are the coefficients of the qubit Hamiltonian in Eq. (2).⁵⁷ However, a more realistic estimate should be performed by directly emulating the quantum measurements process,

$$\sigma_{\text{qc}}^2[H] = \sum_{k=1}^K |c_k|^2 \sigma^2[\hat{P}_k], \quad (14)$$

where $\sigma^2[\hat{P}_k]$ is the variance of the samples obtained from the measurement of the Pauli string \hat{P}_k .

The water molecule consists of three atoms, and therefore, the desired accuracy to be inserted in the previous formula is $\mathcal{E} = 30$ meV, which is comparable with chemical accuracy (1.6 mHa \approx 43.5 meV).

We then define the second quantized Hamiltonian using the molecular orbitals obtained from the minimal STO-3G atomic basis set. The fermion-to-qubit mapping is then achieved using the parity mapping.⁹² This results in a Hamiltonian encoded on 12 qubits. We further reduce this requirement down to nine qubits by exploiting the mapping-specific two-qubit reduction, the planar structure of the molecule (this feature holds even in the presence of distortions), and the freezing of the core orbitals in the oxygen atom. For the configurations in the training dataset, the resulting nine qubit Hamiltonians have a number of Pauli string operators around $K = 1030$. The randomly chosen water molecule, used in the following evaluation, is encoded with $K = 1027$ Pauli string operators.

The probability $p(\delta < \mathcal{E})$ that the deviation of the ground state energy estimate of the H_2O molecule to the exact ground state energy is less than $\mathcal{E} = 30$ meV is shown in Fig. 3.

In Fig. 3, we observe that a probability of $p(\delta < \mathcal{E}) \approx 1$ is reached for a total number of about 10^{10} measurements. The same estimation can be done to estimate an upper bound (using noise level 1 meV/atom). The total number of measurements to reach the desired noise level threshold, therefore, lies between about 10^{10} and 10^{12} . However, advances in quantum measurement protocols are expected to improve this estimate by some orders of magnitude.^{14,43,63,64,67,93}

3. $\text{H}_2\text{-H}_2$ cluster

This model system features two hydrogen molecules, with intramolecular distances sampled from a Gaussian distribution having mean $\mu = 1.42$ bohrs and standard deviation $\sigma = 0.03$ bohr. The intermolecular distances are, instead, sampled from a skewed distribution with two Gaussian tails of different widths. This corresponds to distances between about 4.5 and 10 bohrs and a mean value of 6.0 bohrs. Their respective molecular orientations are also sampled randomly. This system is particularly challenging since it is either unbounded or weakly bounded depending on the level of theory.⁹⁴

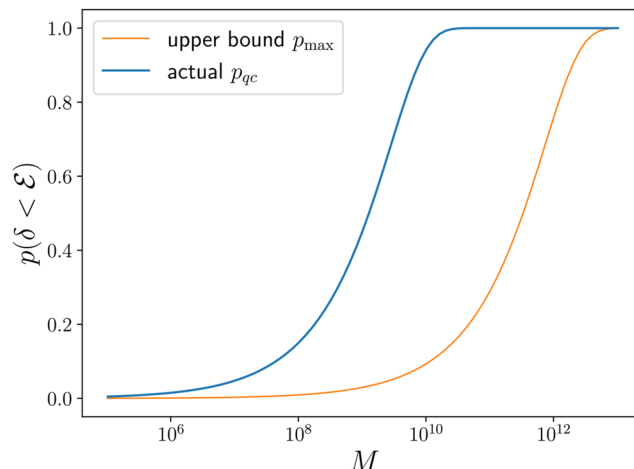


FIG. 3. Probability of obtaining an energy estimate with a statistical error δ smaller than \mathcal{E} as a function of the number of measurements. Here, \mathcal{E} is 30 meV for the single water molecule model (see the text), and the probability reaches $p \approx 1$ when the number of shots is about 10^{10} using the standard Pauli measurement technique (blue line). The upper bound as defined in Eq. (13) would exceed 10^{12} (orange line).

We perform the same kind of assessment and investigate the RMSE on the energy and the forces as a function of the strength of the artificially generated statistical noise. We use a training dataset of 1000 configurations, and we compute the label using DFT, applying the PBE functional with D3 corrections. The mean energy of the dataset is -15.8066 eV/atom with a standard deviation of 2.75 meV/atom. The standard deviation in the force labels is 0.318 eV/Å.

Figure 4 qualitatively shows the same results as for the bulk water and the single water molecule cases. However, this time the energy noise threshold to be met in order to get an RMSE on the same level as the noiseless case lies between 0.1 and

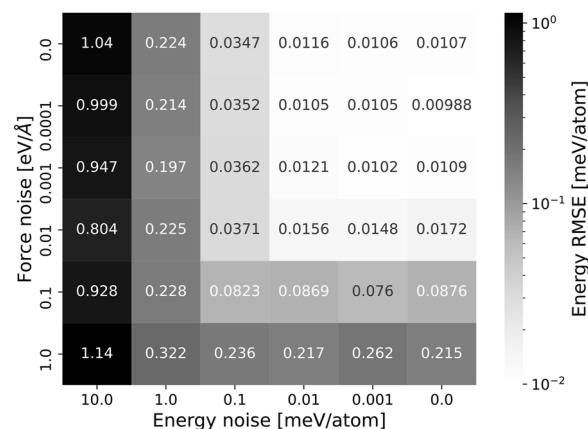


FIG. 4. $\text{H}_2\text{-H}_2$ cluster: Energy RMSE as a function of the noise level (see the text) affecting the energy labels (x axis) and the force labels (y axis) in the training dataset.

0.01 meV/atom. This target is more demanding compared to the single water molecule case as the energy scale of the bounded cluster is also much smaller (about 1.9 meV/atom). For this reason, the number of shots required to compute the energy within the threshold is between about 10^{12} and 10^{13} .

B. Resilience against optimization errors

In this section, we discuss the kind of error we can expect when using variational approaches for the calculation of energy expectation values. More specifically, we consider the case when the electronic structure calculations are not fully converged. In particular, we test two types of variational circuits: the unitary couple cluster (UCC) ansatz and the heuristic ansatz.

Clearly, the MLP is not supposed to improve upon the level of theory at which the dataset has been computed. Therefore, here, we focus on the impact of unconverged variational optimization in the training. For instance, an ansatz featuring many variational parameters can be more difficult to optimize compared to others, thus producing a dataset with scattered labels.

We compare the performance of UCCSD⁹⁵ with a heuristic ansatz, the so-called RY-CNOT circuit, with linear connectivity and depth 24, meaning that the circuit features 24 repeating subunits of RY single qubit gates and a cascade of CNOT (or CX) gates, which represent an entangling block. This depth was necessary to obtain results comparable with the UCCSD ansatz. These results refer to the nine qubit single water molecule model introduced above. The UCCSD ansatz features 58 variational parameters with a total CNOT gate count of 4056, while the heuristic ansatz contains 225 parameters but a less complex circuit, made of 192 CNOT gates.

In Fig. 5, we observe that the MLP trained on datasets coming from the two different variational circuits gives good results on the respective validation dataset, reaching an RMSE of 0.212 and 21.2 meV/atom for the UCCSD and the heuristic ansatz, respectively. As expected, the UCCSD ansatz outperforms the heuristic ansatz since the circuit optimization proceeds smoothly in that case.

It is important to stress that the energies in the validation dataset, reported in the top panels of Fig. 5 (i.e., on the x axis), are also computed with VQE, meaning that they are also affected by the same optimization errors. As it will become clear in the following, this explains, for the most part, the deviations in the top-right panel of Fig. 5.

In the panels on the bottom in Fig. 5, we plot the VQE labels (left) and the MLP predictions (right) against the exact energies (obtained via exact diagonalization). We observe that the MLP fit achieves a significant reduction of the energy variance (i.e., less scattered points). Indeed, the RMSE of the validation dataset compared to the exact energy benchmark is 96.0 ± 63.9 meV/atom, while the RMSE of the MLP on the same benchmark is 95.5 ± 32.1 meV/atom. Therefore, not only we were able to train successfully an MLP from noisy training and validation datasets, but the resulting MLP features a smoother energy landscape compared to the direct VQE calculations. This property is obviously required in MD applications as an artificially corrugated potential implies unphysical, noisy, forces.

C. Resilience against hardware noise

In this section, we assess the third type of errors, which is due to the uncorrected hardware errors typical of state-of-the-art noisy

quantum computers. Similar to the variational noise discussed in Sec. VI A, we do not expect the MLP to improve upon the energies calculated under the effect of hardware noise. Therefore, the focus in this section is on the effect of hardware noise on the learned MLP.

Our assessment includes both noisy simulation and real hardware experiments. Simulations are useful to investigate different levels of gate errors, also beyond the current values. Real hardware experiments are important as they include all possible sources of errors, beyond the ones considered in simulations.

1. Noisy hardware simulations

We simulate the actual hardware noise using a custom Qiskit⁹⁶ noise model whose baseline parameters are derived from the calibration data of current IBM Quantum backends. The custom noisy backend consists of identical qubits and identical gates, meaning that each type of gate behaves identically on all qubits (\mathbb{I} , RZ, SX, X) and all pairs of qubits (CX). The parameters of the custom backend are listed in Table I.

We specifically focus on two types of hardware noise: gate error and readout error. To make the full analysis suited for hardware calculations, we will limit our investigation to a simpler model, namely, the hydrogen molecule, H_2 .

a. Gate error. We model the gate error using coherence limited fidelity for individual gates, i.e., assuming that the reported gate error is due solely to thermal relaxation and dephasing effects, parameterized with the thermal relaxation time T_1 and the decoherence time T_2 , respectively. While this simplified scenario does not entirely reflect the actual experimental conditions—where other effects (e.g., coherent control and calibration errors) and noise channels (including correlated multi-qubit noise) may be present—it is, nevertheless, sufficient to capture the dominant behavior of current noisy processors without complicating the analysis. The baseline value for both T_1 and T_2 is $100 \mu\text{s}$ (see Table I). However, in simulations, it is also possible to assess different scenarios that would correspond to future expected technical improvements in device fabrication. To this end, we systematically increase T_1 and T_2 to investigate the effect of a future gate error reduction on the calculation of the energies. More specifically, we extend T_1 to a maximum value of 2 ms, which is a realistic prediction for the next years according to recent hardware developments.

For this analysis, we randomly create 20 training datasets and one validation dataset for the hydrogen molecule, each with 20 configurations characterized by different bond lengths. Each molecular configuration is randomly rotated in space and features an intramolecular bond distance in the range [0.6, 4.2] bohrs. The H_2 wavefunction is encoded in the STO-3G basis set using the parity mapping and the mapping specific two-qubit reduction, which results in a Hamiltonian on two qubits. The VQE calculations feature a simple variational ansatz tailored to the system, which contains only one variational parameter and one CNOT gate. For each dataset, we calculate the energy of the configurations at different levels of the gate errors and train an MLP on each set of labels. As a reference, we also train an MLP on each training dataset with noiseless energies. In Fig. 6, we report the energy RMSE at different gate errors for the configurations in the validation dataset (blue solid line), the average energy RMSE of the MLP predictions (orange dots), and the average energy RMSE of the reference (noiseless) MLP

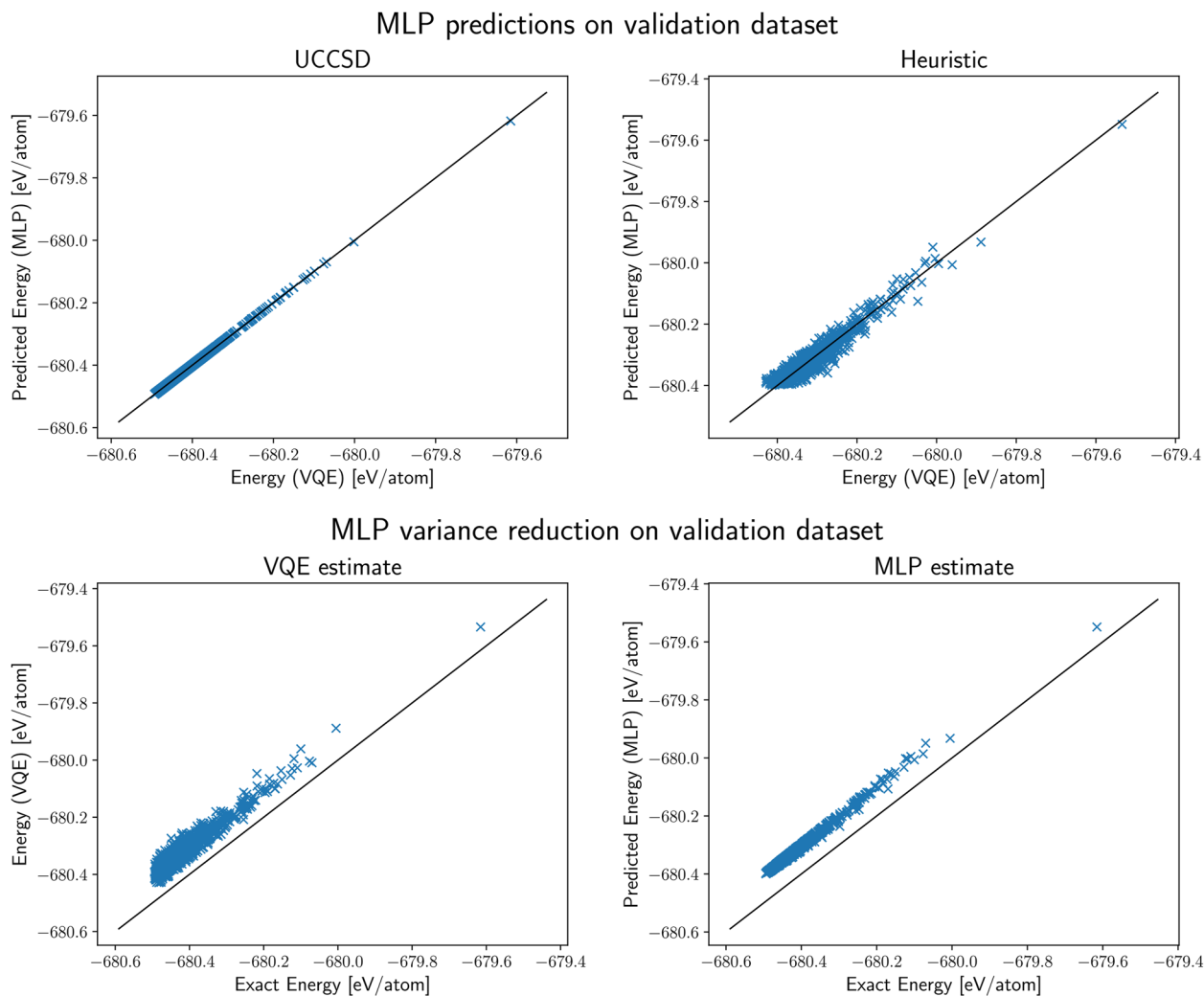


FIG. 5. Top: MLP prediction on the validation dataset for a single water molecule model, where the energy labels have been computed using VQE with the UCCSD ansatz (left) and the heuristic ansatz (right). The VQE error is also present in the validation dataset labels. In the bottom panels, we plot both data series as a function of the exact energy instead. **Bottom left:** VQE energy labels for the validation dataset plotted against their exact values. The positive offset shows the residual variational error of the ansatz, while the fluctuations around it are due to the optimization noise, namely, the energy of some configurations is optimized better compared to others. **Bottom right:** MLP energy predictions for the validation dataset plotted against their exact values. While the MLP (correctly) cannot improve the average variational error of the ansatz, it strongly reduces the fluctuations. The data of the bottom panels refer to the heuristic ansatz only.

predictions (green dashed line). All RMSE values are given with respect to energies obtained with noiseless VQE calculations.

We first note that the MLP predictions closely follow the blue line (for noisy VQE), and we therefore conclude that the MLPs faithfully learn the *noisy* potential energy landscape. As expected, reducing the gate error leads to more accurate MLP in the absolute sense. The crossing point, at which the error due to the gate noise gets smaller than the model error of the MLP, is at about $T_1 \approx 1.75$ ms. Above this value, the MLP error saturates to the model error.

Some comments are in order: on a positive take, these results show that, in principle, even a finite *gate noise* can produce MLPs,

which are as accurate as the best MLP trained on noiseless data. On the other hand, the hydrogen molecule is one of the simplest systems we can study, and the circuit used is very shallow. Larger molecules will require much deeper circuits, and therefore, we expect that the effect of the gate errors will increase significantly. Therefore, this assessment represents a *best-case scenario* concerning MLP training on quantum data in the non-fault-tolerant setting.

b. Readout error. The second type of error we simulate is the readout error. The baseline parameters for the readout errors are listed in Table I. The readout error is best probed by emulating the measurement process. To highlight readout inaccuracies, we

TABLE I. Baseline parameters for custom noise backend. The values are either taken directly from a specific qubit (frequency and anharmonicity) or inspired by an average of different IBM Quantum devices (all remaining values).

Parameter	Baseline value
Thermal relaxation time T1	100 μ s
Dephasing time T2	100 μ s
Qubit frequency	4.77 GHz
Qubit anharmonicity	-0.334 GHz
Readout error ($ 0\rangle$ instead of $ 1\rangle$)	4%
Readout error ($ 1\rangle$ instead of $ 0\rangle$)	2%
1-qubit gate error ^a	0.03%
1-qubit gate time ^a	35.6 ns
CX gate error	1%
CX gate time	430 ns

^aThe RZ gate is applied virtually, and therefore, the gate error and time are both 0.

suppress the statistical fluctuations (see Sec. IV A) using 10^5 shots per circuit to measure the energy. The number of measurements required to reach chemical accuracy for a hydrogen molecule in the 6-31g basis is 10^8 , as mentioned in Sec. IV A. Instead, here, we use the minimal STO-3G basis to describe the hydrogen molecule, and we expect that 10^5 measurements are sufficient for an appropriate suppression of the statistical fluctuations.

In this section, we simulate the readout error at the baseline level and at a level where the error is reduced by a factor of 100 to create a training and validation dataset of the hydrogen molecule,

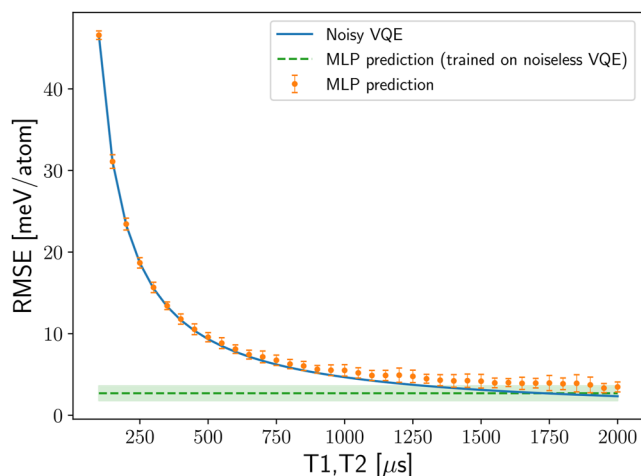


FIG. 6. Average energy RMSE at different levels of gate errors. The gate errors are characterized by the thermal relaxation time T1 and the dephasing time T2, which are set to the same value and varied simultaneously. The blue solid line shows the energy RMSE of the validation configurations, where the energies are obtained at the corresponding level of gate errors and compared to the respective noiseless energies. Each orange point is an average of 20 MLPs that were trained on different training datasets. The error bars show one standard deviation of the energy predictions. The green dashed line serves as a reference and shows the average energy RMSE of the MLP predictions where no gate errors were present in the energy calculations of the reference datasets.

each with 20 configurations. For demonstration purposes, we also train and evaluate an MLP on the resulting datasets. Their performance is reported in Fig. 7. In Fig. 7, the black solid line shows the exact energies of the hydrogen molecule dissociation path, the dashed lines show the predictions of the trained MLPs, and the dots show the energies of the configurations in the validation datasets. The MLPs achieve an energy RMSE of 19.7 and 13.5 meV/atom on the validation dataset, having the baseline and the reduced readout error, respectively. The average shift (mean deviation) between the predicted and exact dissociation curve is 526 and 182 meV/atom, respectively. The non-parallelity error, defined as the difference between the maximum deviation and the minimum deviation from the exact values, of the predicted dissociation curves is 710 and 264 meV/atom, respectively. As expected, reducing the readout error leads to more accurate energy estimations and, therefore, to more accurate MLPs.

At this point, we would like to reaffirm our choice for the number of shots to suppress statistical fluctuations. In Fig. 7, the readout error clearly dominates the statistical fluctuations of the data points, which confirms that 10^5 measurements were sufficient.

2. Hardware experiments

Finally, we also run experiments on IBM Quantum superconducting processors, where all actual error sources are present. We run the hardware calculations for training and validation datasets of the hydrogen molecule, each with 20 configurations. For each configuration, we run 4, 5 and 10 VQE runs on the IBM Quantum devices *ibmq_toronto*, *ibmq_bogota*, and *ibmq_manila*, respectively. All these quantum processors feature a quantum volume of 32. The final energy label is obtained by averaging over the different experiment realizations, after excluding clear unconverged runs. Data

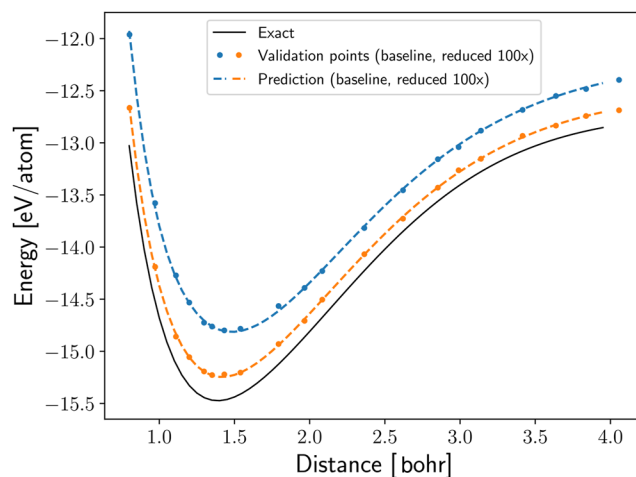


FIG. 7. Predicted hydrogen molecule dissociation path at different levels of readout error assumed in the calculation of the reference energies. The dashed lines show the predictions by the MLPs, the dots show the energies of the configurations in the validation datasets, and the black line shows the dissociation path obtained by exact diagonalization. The baseline readout error values used for the data in blue are listed in Table I. For the data in orange, the readout error is reduced by a factor of 100.

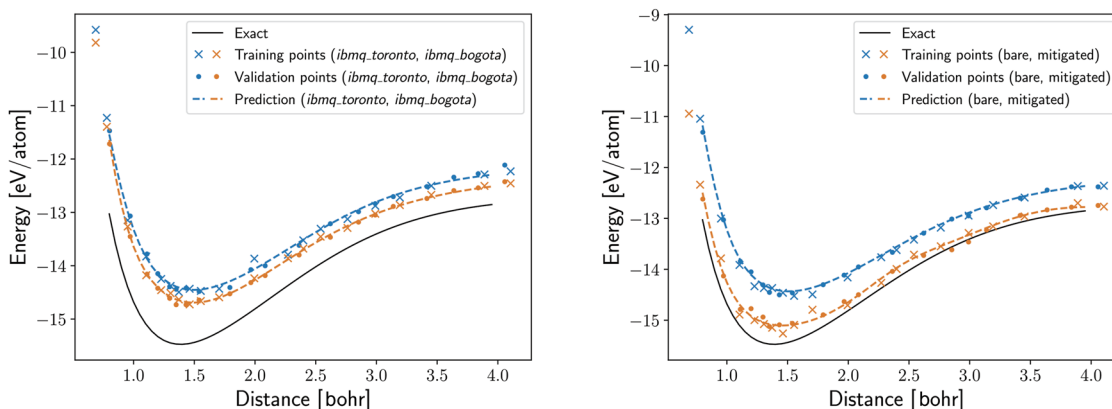


FIG. 8. Left: Prediction of the hydrogen molecule dissociation path by an MLP that was trained and evaluated on datasets obtained with the IBM Quantum devices *ibmq_toronto* (blue) and *ibmq_bogota* (orange). The energies in the training and validation dataset are a filtered average over 4 (5) VQE runs for *ibmq_toronto* (*ibmq_bogota*). **Right:** Prediction of the hydrogen molecule dissociation path by an MLP that was trained and evaluated on datasets obtain with the IBM Quantum device *ibmq_manila* using no readout error mitigation (blue) and using readout error mitigation (orange). The energies in the training and validation dataset are a filtered average over 10 VQE runs.

measured on different devices contribute to separate datasets. All energy expectation values are computed with 8192 measurements (or shots).

The results are summarized in Fig. 8.

The plot on the left reports the curves obtained with *ibmq_toronto* (blue) and *ibmq_bogota* (orange), while the plot on the right shows the results for *ibmq_manila* (blue). In all cases, the predictions of the MLPs closely follow the data points of the training and validation dataset (cross and point markers, respectively). The difference between the curves is due to the different properties of each device, with *ibmq_bogota* outperforming *ibmq_toronto*. We report the error of the MLPs, the average shift of the predicted dissociation curve, and the non-parallelity error in Table II.

With *ibmq_manila*, we also apply measurement error mitigation techniques; to this end, we apply the full calibration matrix method⁸⁰ on 10 additional VQE runs of each selected configuration, where the calibration matrix is refreshed every 30 min. To estimate the final energy of each configuration, we apply the same procedure as described above; as usual, we neglect unconverged VQE runs and average over the remaining energy measurements. The energy estimates using the MLP trained on the obtained datasets are shown in

Fig. 8 in the panel on the right. We observe that the mitigated energies are much closer to the exact energies, in agreement with the error simulation of Sec. VI C 1 b.

We stress once more that the goal of training an MLP from quantum data is not hardware error mitigation *per se*, but rather to obtain a smooth and re-usable interpolation of the noisy data. The computational gain compared to the straightforward molecular dynamics approach of Ref. 68 even when performed on the same single molecule is straightforward. The standard method requires a new VQE calculation for each iteration, while the training of an MLP in this specific case only requires order $\mathcal{O}(10)$ single point VQE runs. Moreover, in Ref. 68, a costly Lanczos error mitigation scheme was sometimes needed, making every single-point VQE calculation $\mathcal{O}(10^2)$ times more expensive compared to the present work, where the stability of the dynamics is ensured by the smoothness of the MLP surface. Finally, this cost needs to be multiplied by the total number of time steps of the MD. To summarize, for a 100 fs simulation of the H_2 molecule, assuming a time step of 0.2 fs, the total cost of a stable quantum-powered MLP simulation is now reduced by a factor of 10^5 compared to the straightforward approach.

TABLE II. Characteristics of the MLPs trained on hardware experiments from different devices. It also includes the RMSE of the trained MLP evaluated on the respective validation dataset, the average shift, defined as the mean deviation between the predicted and the exact dissociation path, and the non-parallelity error, defined as the difference between the maximal and minimal deviation between the predicted and the exact dissociation path.

Device	Energy RMSE (meV/atom)	Average shift (meV/atom)	Non-parallelity error (meV/atom)
<i>ibmq_toronto</i>	64.3	797	941
<i>ibmq_bogota</i>	38.6	571	1048
<i>ibmq_manila</i>	46.6	780	1366
<i>ibmq_manila</i> (mitig.)	72.5	214	456

VII. CONCLUSIONS

We propose the usage of classical machine learning potentials (MLPs) trained on quantum electronic structure data to enable large-scale materials simulations. The motivation behind this is quite simple: while quantum computing algorithms can outperform their classical counterpart for electronic structure problems, they still feature a polynomial runtime (possibly with a large prefactor) that can still prevent applications to bulk materials.

MLPs have been successfully introduced in materials simulations powered by classical approximate electronic structure solvers, enabling truly large-scale and equilibrated simulations.^{27,29} Here, we assess the trainability of an MLP using quantum data obtained from a variational calculation. In particular, we study the impact of three types of noise that are characteristic to the quantum algorithm: the statistical noise, the optimization error, and the hardware errors.

These errors impact the training and validation energy labels as

$$E_{\text{label}} = E_{\text{true}} + \Delta + \eta, \quad (15)$$

where Δ is a systematic error and η is a fluctuation around this offset. While the MLP is not intended to compensate for any systematic error Δ , it can greatly mitigate the random fluctuations affecting the labels. These may arise from the statistical error in the evaluation of the energy and force estimator, the VQE optimization error that may affect some dataset points more than others, and any non-systematic component of the hardware noise.

Here, we use an MLP based on the state-of-the-art neural-network potentials and show that there exists a threshold value for the noise strength such that we can achieve a training as good as in the noiseless case. The resulting MLP features a smooth energy surface that would allow for stable molecular dynamics simulations or structural optimizations.

We substantiate our research through the simulations of the separate sources of noises. We finally generate training and validation datasets using actual quantum hardware and obtain the first MLP trained with electronic structure calculations using a real quantum computer.

While in our assessment we consider a neural-network type of MLP, next research directions include the possibility of using kernel-based models, which tend to have a better performance when only few training data are available.⁹⁷

ACKNOWLEDGMENTS

I.T. acknowledges the financial support from the Swiss National Science Foundation (SNF) through Grant No. 200021-179312. IBM, the IBM logo, and ibm.com are trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. The current list of IBM trademarks is available at <https://www.ibm.com/legal/copytrade>.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Julian Schuhmacher: Data curation (equal); Investigation (equal); Software (equal); Validation (equal); Visualization (equal); Writing – original draft (equal); Writing – review & editing (equal). **Guglielmo Mazzola:** Conceptualization (equal); Investigation (equal); Methodology (equal); Supervision (equal); Writing – original draft (equal); Writing – review & editing (equal). **Francesco Tacchino:** Conceptualization (equal); Investigation (equal); Methodology (equal); Supervision (equal); Writing – original draft (equal); Writing – review & editing (equal). **Olga Dmitriyeva:** Data curation (equal); Investigation (equal); Software (equal). **Tai Bui:** Data curation (equal); Investigation (equal); Software (equal). **Shanshan Huang:** Data curation (equal); Investigation (equal); Software (equal). **Ivano Tavernelli:** Conceptualization (equal); Funding acquisition (equal); Investigation (equal); Methodology (equal); Supervision (equal); Writing – original draft (equal); Writing – review & editing (equal).

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon reasonable request.

APPENDIX A: DATASET NORMALIZATION

Normalizing the labels in the training dataset is a common practice to improve the training of a neural network.⁹¹ In the case of a machine learning potential (MLP), the normalization defines internal units, which are independent of a physical unit system. Fortunately, the normalization process is integrated into the training procedure of n2p2.⁵³ Given a dataset of atomic structures with energy (E) and force (F), the normalization transformation is parameterized by three parameters, the mean energy per atom $\langle E \rangle$, a conversion factor for the energies $c_{\text{energy}} = 1/\sigma_E$, and a conversion factor for distances $c_{\text{length}} = \sigma_F/\sigma_E$, where σ_E and σ_F are the standard deviation of the energy and the forces, respectively. Applying the transformation

$$E_i^* = \frac{E_{\text{Ref}}^i - N_a^i \langle e \rangle}{\sigma_e}, \quad (A1)$$

$$F^* = \frac{c_{\text{energy}}}{c_{\text{length}}} F_{\text{Ref}} \quad (A2)$$

to each configuration in the dataset ensures that the transformed labels (E^* , F^*) have zero mean and unit standard deviation,⁹¹ i.e., $\langle E^* \rangle = 0$, $\sigma_E^* = 1$, and $\sigma_F^* = 1$ (the forces should already have zero mean).

The normalization is successful as long as we input both the energies *and* the forces. However, using the VQE algorithm, we only calculate the energy of the atomic configurations and set all the forces to zero. Concerning the training, this is no issue as the training on only the energy labels is supported by n2p2. However, during the normalization process, the conversion factor for distances is set to $c_{\text{length}} = 0$ ($\sigma_F = 0$). This would set all distances to 0 and lead to problems in the subsequent calculation of the symmetry function values. Therefore, if the forces are not available for the normalization process, we manually set $c_{\text{length}} = 1$.

APPENDIX B: SYMMETRY FUNCTIONS

Two types of symmetry functions are used in this article. The first is the radial symmetry function class. For an atom labeled with index i , the radial symmetry function is defined as

$$G_2^i = \sum_j e^{-\eta(R_{ij}-R_s)^2} \cdot f_c(R_{ij}), \quad (\text{B1})$$

where η determines the width and R_s determines the position of the Gaussian.

The cutoff function $f_c(R_{ij})$ ensures that the value and the derivative of the symmetry function go to zero if the distance R_{ij} between the central atom and a neighboring atom is bigger than the cutoff radius R_c .

The cutoff function that we used for the MLPs trained in this paper is

$$f_c(R_{ij}) = \begin{cases} \tanh^3[1 - R_{ij}/R_c] & \text{for } R_{ij} \leq R_c, \\ 0 & \text{for } R_{ij} > R_c. \end{cases} \quad (\text{B2})$$

A list of other cutoff functions can be found in Ref. 98.

The second type of symmetry functions we consider are angular symmetry functions, which are a sum of three-body terms. It is defined as

$$G_3^i = 2^{1-\zeta} \sum_j \sum_{k \neq j} (1 + \lambda \cdot \cos \theta_{ijk})^\zeta e^{-\eta(R_{ij}^2 + R_{ik}^2 + R_{jk}^2)} \times f_c(R_{ij})f_c(R_{ik})f_c(R_{jk}), \quad (\text{B3})$$

where $\theta_{ijk} = \arccos\left(\frac{\mathbf{R}_{ij} \cdot \mathbf{R}_{ik}}{R_{ij} \cdot R_{ik}}\right)$ is the angle between the three atoms i , j , and k . The parameters determining the shape of the function are η , λ , and ζ . The parameter η , again, determines the width of the Gaussian part of the function. The parameter λ can only take the values 1 and -1 , which shifts the maximum of the cosine part either to $\theta_{ijk} = 0^\circ$ or $\theta_{ijk} = 180^\circ$, respectively. The parameter ζ determines the angular resolution.

1. Normalization

Similarly to the labels, the input to the NN is also normalized. This balances the impact of the different symmetry functions on the first hidden layer in the NN.

The normalization transformation is

$$G_i^{\text{scaled}} = \frac{G_i - \langle G_i \rangle}{G_{i,\text{max}} - G_{i,\text{min}}}, \quad (\text{B4})$$

which centers the symmetry functions G_i with their mean $\langle G_i \rangle$ and rescales them to the interval $[-1, 1]$.⁹¹

2. Forces

The force component $F_{i,k}$ of atom i is calculated from the total energy by taking the derivative with respect to the component k of the position \vec{R}_i of the atom,

$$F_{i,k} = -\frac{\partial E_{\text{NNP}}}{\partial R_{i,k}} = -\sum_{j=1}^{N_{\text{atom}}} \frac{\partial E_j}{\partial R_{i,k}}. \quad (\text{B5})$$

This expression can be evaluated by applying the chain rule,

$$F_{i,k} = -\sum_{j=1}^{N_{\text{atom}}} \sum_{l=1}^{N_{\text{sym},j}} \frac{\partial E_j}{\partial G_{jl}} \frac{\partial G_{jl}}{\partial R_{i,k}}, \quad (\text{B6})$$

where $N_{\text{sym},j}$ is the number of symmetry functions for atom j and G_{jl} is the l th symmetry function of atom j . The first partial derivative is given by the functional form of the NN. The second partial derivative is given by the functional form of the symmetry functions and can be calculated analytically. For the two symmetry function types given above, the derivatives can be found in the supplementary material of Ref. 98.

APPENDIX C: AUTOMATIC SELECTION OF SYMMETRY FUNCTIONS

This section provides a short review of the method to automatically select symmetry functions proposed in Ref. 86 and adopted in this work. The algorithm is based on a feature selection method, called the CUR decomposition, which creates a low-rank approximation of the initial feature matrix \mathbf{X} in terms of its columns and rows. The first step of the procedure is the construction of a pool of N candidate symmetry functions $\{\Phi_j\}$. Given a dataset of M configurations $\{A_i\}$, the feature matrix is defined as $X_{ij} = \Phi_j(A_i)$. In the second step, we then apply the feature selection to the columns (rows) of the feature matrix to select a small subset of N' symmetry functions (M' configurations), which capture the important structural information of the considered system. Below the two steps are reviewed in more detail.

The creation of candidate symmetry functions is done by generating values for the parameters that determine the shape of the symmetry functions. For the radial symmetry functions G_2 [Eq. (B1)], two parameter sets are created. In the first set, the Gaussians are centered at the reference atom ($R_s = 0$) and have widths chosen according to

$$\eta_m = \left(\frac{n^{m/n}}{r_c}\right)^2, \quad (\text{C1})$$

where n is the number of desired parameters in this parameters set and $m = 0, 1, \dots, n$. The second set of parameters is created in the following way:

$$R_{s,m} = \frac{r_c}{n^{m/n}}, \quad (\text{C2})$$

$$\eta_{s,m} = \frac{1}{(R_{s,n-m} - R_{s,n-m-1})^2}, \quad (\text{C3})$$

which creates a set of Gaussians that are narrow close to the reference atom and wider as the distance increases.

For the angular symmetry functions G_3 [Eq. (B3)], only one set of parameters is created. The values for η are chosen according to Eq. (C1); λ takes the values $\{-1, 1\}$, and for ζ , a few values on a logarithmic scale are chosen, e.g., $\{1, 4, 16\}$.

The method to select the most important features of a feature matrix \mathbf{X} has the following form:

$$\mathbf{X} \approx \tilde{\mathbf{X}} = \mathbf{C} \mathbf{U} \mathbf{R}, \quad (\text{C4})$$

where \mathbf{C} and \mathbf{R} are matrices that consist of a subset of columns and rows of the original feature matrix \mathbf{X} . We execute the following steps for the selection of the subset of columns (\mathbf{C}).

- Calculate the singular value decomposition (SVD) of \mathbf{X} .
- Calculate an importance score for each column c ,

$$\pi_c = \sum_{j=1}^k \left(v_c^{(j)} \right)^2, \quad (\text{C5})$$

where $v_c^{(j)}$ is the c th coordinate of the j th right singular vector and k is the number of singular vectors that are considered for the score. A value of $k = 1$ is proposed for an efficient selection.

- Pick column l with the highest importance score.
- Orthogonalize the remaining columns in the feature matrix \mathbf{X} with respect to the l th column X_l ,

$$X_j \leftarrow X_j - X_l (X_l \cdot X_j) / |X_l|^2. \quad (\text{C6})$$

- Repeat the steps above on the orthogonalized matrix until the desired number of columns is reached or the error of the approximation [Eq. (C7)] is below a desired threshold.

The extracted columns form the matrix \mathbf{C} . Similarly, the matrix \mathbf{R} can be constructed using the algorithm above to select a subset of columns from \mathbf{X}^T (rows from \mathbf{X}). The matrix \mathbf{U} then is defined as $\mathbf{U} = \mathbf{C}^+ \mathbf{X} \mathbf{R}^+$ ($+$ indicates the pseudoinverse). The accuracy of the approximation is

$$\epsilon = \|\mathbf{X} - \mathbf{C} \mathbf{U} \mathbf{R}\|_F / \|\mathbf{X}\|_F. \quad (\text{C7})$$

APPENDIX D: CREATING REFERENCE DATASETS

For the bulk water, we use a dataset that is already established in the literature.^{87,88,91} The dataset for the single water molecule is derived from the bulk water dataset by extracting H_2O atom groups. The initial training dataset was then created by randomly selecting 1000 configurations from the extracted H_2O configurations. We could further reduce the number of configurations by applying the CUR decomposition⁸⁶ (see Appendix C) to select for a subset of the configurations. We found that 100 is a convenient dataset size. The validation dataset is created in the same way by choosing a different (distinct) set of extracted H_2O configurations.

We created our own reference datasets for the H_2 - H_2 cluster and the hydrogen molecule, H_2 . For the creation of a reference dataset, it is recommended to use the procedure reviewed in Ref. 24. However, the considered systems are very simple, and we expect a sufficient coverage of the configuration space already from a random sampling. In both cases, the H_2 - H_2 cluster and the hydrogen molecule, we created an initial dataset with 1000 configurations. We also tried to reduce the number of configurations in the dataset with the CUR decomposition. For the H_2 - H_2 cluster, the training accuracy got gradually worse when reducing the number of configurations, so we kept all the 1000 configurations in the training dataset. In the case of the hydrogen molecule, we found that 20 is a convenient dataset size. For both systems, the validation datasets are also

created by randomly sampling from the configuration space of the respective system.

REFERENCES

- ¹R. P. Feynman, "Simulating physics with computers," *Int. J. Theor. Phys.* **21**, 467–488 (1982).
- ²A. Aspuru-Guzik, A. D. Dutoi, P. J. Love, and M. Head-Gordon, "Simulated quantum computation of molecular energies," *Science* **309**, 1704–1707 (2005).
- ³N. Moll, P. Barkoutsos, L. S. Bishop, J. M. Chow, A. Cross, D. J. Egger, S. Filipp, A. Fuhrer, J. M. Gambetta, M. Ganzhorn, A. Kandala, A. Mezzacapo, P. Müller, W. Riess, G. Salis, J. Smolin, I. Tavernelli, and K. Temme, "Quantum optimization using variational algorithms on near-term quantum devices," *Quantum Sci. Technol.* **3**, 030503 (2018).
- ⁴Y. Cao, J. Romero, J. P. Olson, M. Degroote, P. D. Johnson, M. Kieferová, I. D. Kivlichan, T. Menke, B. Peropadre, N. P. D. Sawaya *et al.*, "Quantum chemistry in the age of quantum computing," *Chem. Rev.* **119**, 10856–10915 (2019).
- ⁵P. Krantz, M. Kjaergaard, F. Yan, T. P. Orlando, S. Gustavsson, and W. D. Oliver, "A quantum engineer's guide to superconducting qubits," *Appl. Phys. Rev.* **6**, 021318 (2019).
- ⁶C. D. Bruzewicz, J. Chiaverini, R. McConnell, and J. M. Sage, "Trapped-ion quantum computing: Progress and challenges," *Appl. Phys. Rev.* **6**, 021314 (2019).
- ⁷A. Chatterjee, P. Stevenson, S. De Franceschi, A. Morello, N. P. de Leon, and F. Kuemmeth, "Semiconductor qubits in practice," *Nat. Rev. Phys.* **3**, 157–177 (2021).
- ⁸M. Saffman, T. G. Walker, and K. Mølmer, "Quantum information with Rydberg atoms," *Rev. Mod. Phys.* **82**, 2313–2363 (2010).
- ⁹A. Montanaro, "Quantum algorithms: An overview," *npj Quantum Inf.* **2**, 15023 (2016).
- ¹⁰B. Bauer, S. Bravyi, M. Motta, and G. K.-L. Chan, "Quantum algorithms for quantum chemistry and quantum materials science," *Chem. Rev.* **120**, 12685–12717 (2020).
- ¹¹M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, and P. J. Coles, "Variational quantum algorithms," *Nat. Rev. Phys.* **3**, 625–644 (2021).
- ¹²K. Bharti, A. Cervera-Lierta, T. H. Kyaw, T. Haug, S. Alperin-Lea, A. Anand, M. Degroote, H. Heimonen, J. S. Kottmann, T. Menke, "Noisy intermediate-scale quantum (NISQ) algorithms," *Rev. Mod. Phys.* **94**, 015004 (2022).
- ¹³A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O'Brien, "A variational eigenvalue solver on a photonic quantum processor," *Nat. Commun.* **5**, 4213 (2014).
- ¹⁴A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J. M. Chow, and J. M. Gambetta, "Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets," *Nature* **549**, 242–246 (2017).
- ¹⁵S. McArdle, S. Endo, A. Aspuru-Guzik, S. C. Benjamin, and X. Yuan, "Quantum computational chemistry," *Rev. Mod. Phys.* **92**, 015003 (2020).
- ¹⁶K. Burke, "Perspective on density functional theory," *J. Chem. Phys.* **136**, 150901 (2012).
- ¹⁷W. M. C. Foulkes, L. Mitás, R. J. Needs, and G. Rajagopal, "Quantum Monte Carlo simulations of solids," *Rev. Mod. Phys.* **73**, 33–83 (2001).
- ¹⁸K. Nakano, C. Attaccalite, M. Barborini, L. Capriotti, M. Casula, E. Coccia, M. Dagrada, C. Genovese, Y. Luo, G. Mazzola *et al.*, "TurboRVB: A many-body toolkit for *ab initio* electronic simulations by quantum Monte Carlo," *J. Chem. Phys.* **152**, 204121 (2020).
- ¹⁹C. Gidney and A. G. Fowler, "Efficient magic state factories with a catalyzed $|CCZ\rangle$ to $|2T\rangle$ transformation," *Quantum* **3**, 135 (2019).
- ²⁰M. Reiher, N. Wiebe, K. M. Svore, D. Wecker, and M. Troyer, "Elucidating reaction mechanisms on quantum computers," *Proc. Natl. Acad. Sci. U. S. A.* **114**, 7555–7560 (2017).
- ²¹M. Rupp, "Machine learning for quantum mechanics in a nutshell," *Int. J. Quantum Chem.* **115**, 1058–1073 (2015).
- ²²A. P. Bartók and G. Csányi, "Gaussian approximation potentials: A brief tutorial introduction," *Int. J. Quantum Chem.* **115**, 1051–1057 (2015).

- ²³J. Behler, "Perspective: Machine learning potentials for atomistic simulations," *J. Chem. Phys.* **145**, 170901 (2016).
- ²⁴J. Behler, "Constructing high-dimensional neural network potentials: A tutorial review," *Int. J. Quantum Chem.* **115**, 1032–1050 (2015).
- ²⁵F. Noé, A. Tkatchenko, K.-R. Müller, and C. Clementi, "Machine learning for molecular simulation," *Annu. Rev. Phys. Chem.* **71**, 361–390 (2020).
- ²⁶A. P. Bartók, S. De, C. Poelking, N. Bernstein, J. R. Kermode, G. Csányi, and M. Ceriotti, "Machine learning unifies the modeling of materials and molecules," *Sci. Adv.* **3**, e1701816 (2017).
- ²⁷B. Cheng, G. Mazzola, C. J. Pickard, and M. Ceriotti, "Evidence for supercritical behaviour of high-pressure liquid hydrogen," *Nature* **585**, 217–220 (2020).
- ²⁸T. E. Gartner, L. Zhang, P. M. Piaggi, R. Car, A. Z. Panagiotopoulos, and P. G. Debenedetti, "Signatures of a liquid–liquid transition in an *ab initio* deep neural network model for water," *Proc. Natl. Acad. Sci. U. S. A.* **117**, 26040–26046 (2020).
- ²⁹V. L. Deringer, N. Bernstein, G. Csányi, C. Ben Mahmoud, M. Ceriotti, M. Wilson, D. A. Drabold, and S. R. Elliott, "Origins of structural and electronic transitions in disordered silicon," *Nature* **589**, 59–64 (2021).
- ³⁰D. Wang, O. Higgott, and S. Brierley, "Accelerated variational quantum eigensolver," *Phys. Rev. Lett.* **122**, 140504 (2019).
- ³¹G. Wang, D. E. Koh, P. D. Johnson, and Y. Cao, "Minimizing estimation runtime on noisy quantum computers," *PRX Quantum* **2**, 010346 (2021).
- ³²I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT Press, 2016).
- ³³G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science* **313**, 504–507 (2006).
- ³⁴Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature* **521**, 436–444 (2015).
- ³⁵G. Carleo, I. Cirac, K. Cranmer, L. Daudet, M. Schuld, N. Tishby, L. Vogt-Maranto, and L. Zdeborová, "Machine learning and the physical sciences," *Rev. Mod. Phys.* **91**, 045002 (2019).
- ³⁶G. Carleo and M. Troyer, "Solving the quantum many-body problem with artificial neural networks," *Science* **355**, 602–606 (2017).
- ³⁷K. Choo, A. Mezzacapo, and G. Carleo, "Fermionic neural-network states for *ab-initio* electronic structure," *Nat. Commun.* **11**, 2368 (2020).
- ³⁸D. Pfau, J. S. Spencer, A. G. Matthews, and W. M. C. Foulkes, "*Ab initio* solution of the many-electron Schrödinger equation with deep neural networks," *Phys. Rev. Res.* **2**, 033429 (2020).
- ³⁹J. Hermann, Z. Schätzle, and F. Noé, "Deep-neural-network solution of the electronic Schrödinger equation," *Nat. Chem.* **12**, 891–897 (2020).
- ⁴⁰G. Torlai, G. Mazzola, J. Carrasquilla, M. Troyer, R. Melko, and G. Carleo, "Neural-network quantum state tomography," *Nat. Phys.* **14**, 447 (2018).
- ⁴¹B. S. Rem, N. Käming, M. Tarnowski, L. Asteria, N. Fläschner, C. Becker, K. Senegstock, and C. Weitenberg, "Identifying quantum phase transitions using artificial neural networks on experimental data," *Nat. Phys.* **15**, 917–920 (2019).
- ⁴²J. F. Rodríguez-Nieva and M. S. Scheurer, "Identifying topological order through unsupervised machine learning," *Nat. Phys.* **15**, 790–795 (2019).
- ⁴³G. Torlai, G. Mazzola, G. Carleo, and A. Mezzacapo, "Precise measurement of quantum observables with neural-network estimators," *Phys. Rev. Res.* **2**, 022060 (2020).
- ⁴⁴G. Pilińska, C. Wang, X. Jiang, S. Rajasekaran, and R. Ramprasad, "Accelerating materials property predictions using machine learning," *Sci. Rep.* **3**, 2810 (2013).
- ⁴⁵M. Rupp, A. Tkatchenko, K.-R. Müller, and O. A. von Lilienfeld, "Fast and accurate modeling of molecular atomization energies with machine learning," *Phys. Rev. Lett.* **108**, 058301 (2012).
- ⁴⁶R. Ramakrishnan, P. O. Dral, M. Rupp, and O. A. von Lilienfeld, "Big data meets quantum chemistry approximations: The Δ -machine learning approach," *J. Chem. Theory Comput.* **11**, 2087–2096 (2015).
- ⁴⁷F. A. Faber, A. Lindmaa, O. A. von Lilienfeld, and R. Armiento, "Machine learning energies of 2 million elpasolite (ABC₂D₆) crystals," *Phys. Rev. Lett.* **117**, 135502 (2016).
- ⁴⁸J. Schmidt, J. Shi, P. Borlido, L. Chen, S. Botti, and M. A. L. Marques, "Predicting the thermodynamic stability of solids combining density functional theory and machine learning," *Chem. Mater.* **29**, 5090–5103 (2017).
- ⁴⁹B. Cheng, E. A. Engel, J. Behler, C. Dellago, and M. Ceriotti, "Ab initio thermodynamics of liquid and solid water," *Proc. Natl. Acad. Sci. U. S. A.* **116**, 1110–1115 (2019).
- ⁵⁰T. B. Blank, S. D. Brown, A. W. Calhoun, and D. J. Doren, "Neural network models of potential energy surfaces," *J. Chem. Phys.* **103**, 4129–4137 (1995).
- ⁵¹J. Behler and M. Parrinello, "Generalized neural-network representation of high-dimensional potential-energy surfaces," *Phys. Rev. Lett.* **98**, 146401 (2007).
- ⁵²L. Bonati and M. Parrinello, "Silicon liquid structure and crystal nucleation from *ab initio* deep metadynamics," *Phys. Rev. Lett.* **121**, 265701 (2018).
- ⁵³A. Singraber, mpbircher, S. Reeve, D. W. Swenson, J. Lauret, and philippedavid, <https://github.com/CompPhysVienna/n2p2> : Version 2.1.1, 2021.
- ⁵⁴P. K. Barkoutsos, J. F. Gonthier, I. Sokolov, N. Moll, G. Salis, A. Fuhrer, M. Ganzhorn, D. J. Egger, M. Troyer, A. Mezzacapo, S. Filipp, and I. Tavernelli, "Quantum algorithms for electronic structure calculations: Particle-hole Hamiltonian and optimized wave-function expansions," *Phys. Rev. A* **98**, 022322 (2018).
- ⁵⁵M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition* (Cambridge University Press, 2010).
- ⁵⁶D. S. Abrams and S. Lloyd, "Quantum algorithm providing exponential speed increase for finding eigenvalues and eigenvectors," *Phys. Rev. Lett.* **83**, 5162 (1999).
- ⁵⁷D. Wecker, M. B. Hastings, and M. Troyer, "Progress towards practical quantum variational algorithms," *Phys. Rev. A* **92**, 042303 (2015).
- ⁵⁸H.-Y. Huang, R. Kueng, and J. Preskill, "Predicting many properties of a quantum system from very few measurements," *Nat. Phys.* **16**, 1050–1057 (2020).
- ⁵⁹C. Hadfield, S. Bravyi, R. Raymond, and A. Mezzacapo, "Measurements of quantum Hamiltonians with locally-biased classical shadows," *Commun. Math. Phys.* **391**, 951–967 (2022).
- ⁶⁰A. Jena, S. Genin, and M. Mosca, "Pauli partitioning with respect to gate sets," [arXiv:1907.07859](https://arxiv.org/abs/1907.07859) (2019).
- ⁶¹T.-C. Yen, V. Verteletskyi, and A. F. Izmaylov, "Measuring all compatible operators in one series of single-qubit measurements using unitary transformations," *J. Chem. Theory Comput.* **16**, 2400–2409 (2020).
- ⁶²W. J. Huggins, J. R. McClean, N. C. Rubin, Z. Jiang, N. Wiebe, K. B. Whaley, and R. Babbush, "Efficient and noise resilient measurements for quantum chemistry on near-term quantum computers," *npj Quantum Inf.* **7**, 23 (2021).
- ⁶³P. Gokhale, O. Angiuli, Y. Ding, K. Gui, T. Tomesh, M. Suchara, M. Martonosi, and F. T. Chong, "Minimizing state preparations in variational quantum eigensolver by partitioning into commuting families," [arXiv:1907.13623](https://arxiv.org/abs/1907.13623) (2019).
- ⁶⁴O. Crawford, B. van Straaten, D. Wang, T. Parks, E. Campbell, and S. Brierley, "Efficient quantum measurement of Pauli operators in the presence of finite sampling error," *Quantum* **5**, 385 (2021).
- ⁶⁵A. Zhao, A. Tranter, W. M. Kirby, S. F. Ung, A. Miyake, and P. Love, "Measurement reduction in variational quantum algorithms," *Phys. Rev. A* **101**, 062322 (2020).
- ⁶⁶I. Hamamura and T. Imamichi, "Efficient evaluation of quantum observables using entangled measurements," *npj Quantum Inf.* **6**, 56 (2020).
- ⁶⁷G. García-Pérez, M. A. Rossi, B. Sokolov, F. Tacchino, P. K. Barkoutsos, G. Mazzola, I. Tavernelli, and S. Maniscalco, "Learning to measure: Adaptive informationally complete generalized measurements for quantum algorithms," *PRX Quantum* **2**, 040342 (2021).
- ⁶⁸I. O. Sokolov, P. K. Barkoutsos, L. Moeller, P. Suchsland, G. Mazzola, and I. Tavernelli, "Microcanonical and finite-temperature *ab initio* molecular dynamics simulations on quantum computers," *Phys. Rev. Res.* **3**, 013125 (2021).
- ⁶⁹A. Tirelli, G. Tenti, K. Nakano, and S. Sorella, "High pressure hydrogen by machine learning and quantum Monte Carlo," *Phys. Rev. B* **106**, L041105 (2022).
- ⁷⁰H. R. Grimsley, S. E. Economou, E. Barnes, and N. J. Mayhall, "An adaptive variational algorithm for exact molecular simulations on a quantum computer," *Nat. Commun.* **10**, 3007 (2019).
- ⁷¹J. Lee, W. J. Huggins, M. Head-Gordon, and K. B. Whaley, "Generalized unitary coupled cluster wave functions for quantum computation," *J. Chem. Theory Comput.* **15**, 311–324 (2019).
- ⁷²B. Cooper and P. J. Knowles, "Benchmark studies of variational, unitary and extended coupled cluster methods," *J. Chem. Phys.* **133**, 234102 (2010).
- ⁷³S. Wang, E. Fontana, M. Cerezo, K. Sharma, A. Sone, L. Cincio, and P. J. Coles, "Noise-induced barren plateaus in variational quantum algorithms," *Nat. Commun.* **12**, 6961 (2021).

- ⁷⁴C. Kim, K. D. Park, and J.-K. Rhee, "Quantum error mitigation with artificial neural network," *IEEE Access* **8**, 188853–188860 (2020).
- ⁷⁵E. R. Bannett, F. Hopfmueller, B. Kulchytskyy, J. Carrasquilla, and P. Ronagh, "Neural error mitigation of near-term quantum simulations," *Nat. Mach. Intell.* **4**, 618–624 (2022).
- ⁷⁶L. Cincio, K. Rudinger, M. Sarovar, and P. J. Coles, "Machine learning of noise-resilient quantum circuits," *PRX Quantum* **2**, 010324 (2021).
- ⁷⁷Y. Li and S. C. Benjamin, "Efficient variational quantum simulator incorporating active error minimization," *Phys. Rev. X* **7**, 021050 (2017).
- ⁷⁸A. Kandala, K. Temme, A. D. Córcoles, A. Mezzacapo, J. M. Chow, and J. M. Gambetta, "Error mitigation extends the computational reach of a noisy quantum processor," *Nature* **567**, 491–495 (2019).
- ⁷⁹S. Endo, S. C. Benjamin, and Y. Li, "Practical quantum error mitigation for near-future applications," *Phys. Rev. X* **8**, 031027 (2018).
- ⁸⁰S. Bravyi, S. Sheldon, A. Kandala, D. C. McKay, and J. M. Gambetta, "Mitigating measurement errors in multiqubit experiments," *Phys. Rev. A* **103**, 042605 (2021).
- ⁸¹P. Suchsland, F. Tacchino, M. H. Fischer, T. Neupert, P. K. Barkoutsos, and I. Tavernelli, "Algorithmic error mitigation scheme for current quantum processors," *Quantum* **5**, 492 (2021).
- ⁸²B. Koczor, "Exponential error suppression for near-term quantum devices," *Phys. Rev. X* **11**, 031057 (2021).
- ⁸³J. R. McClean, Z. Jiang, N. C. Rubin, R. Babbush, and H. Neven, "Decoding quantum errors with subspace expansions," *Nat. Commun.* **11**, 636 (2020).
- ⁸⁴T. E. O'Brien, S. Polla, N. C. Rubin, W. J. Huggins, S. McArdle, S. Boixo, J. R. McClean, and R. Babbush, "Error mitigation via verified phase estimation," *PRX Quantum* **2**, 020317 (2021).
- ⁸⁵Z. Chen *et al.*, "Exponential suppression of bit or phase errors with cyclic error correction," *Nature* **595**, 383–387 (2021).
- ⁸⁶G. Imbalzano, A. Anelli, D. Giofré, S. Klees, J. Behler, and M. Ceriotti, "Automatic selection of atomic fingerprints and reference configurations for machine-learning potentials," *J. Chem. Phys.* **148**, 241730 (2018).
- ⁸⁷T. Morawietz, A. Singraber, C. Dellago, and J. Behler, "How van der Waals interactions determine the unique properties of water," *Proc. Natl. Acad. Sci. U. S. A.* **113**, 8368–8373 (2016).
- ⁸⁸T. Morawietz and J. Behler (2019), "HDNNP training data set for H₂O," Zenodo. <https://doi.org/10.5281/zenodo.2634098>
- ⁸⁹B. Hammer, L. B. Hansen, and J. K. Nørskov, "Improved adsorption energetics within density-functional theory using revised Perdew-Burke-Ernzerhof functionals," *Phys. Rev. B* **59**, 7413–7421 (1999).
- ⁹⁰S. Grimme, J. Antony, S. Ehrlich, and H. Krieg, "A consistent and accurate *ab initio* parametrization of density functional dispersion correction (DFT-D) for the 94 elements H-Pu," *J. Chem. Phys.* **132**, 154104 (2010).
- ⁹¹A. Singraber, T. Morawietz, J. Behler, and C. Dellago, "Parallel multistream training of high-dimensional neural network potentials," *J. Chem. Theory Comput.* **15**, 3075–3092 (2019).
- ⁹²S. Bravyi, J. M. Gambetta, A. Mezzacapo, and K. Temme, "Tapering off qubits to simulate fermionic Hamiltonians," [arXiv:1701.08213](https://arxiv.org/abs/1701.08213) [quant-ph] (2017).
- ⁹³A. F. Izmaylov, T.-C. Yen, R. A. Lang, and V. Verteletskiy, "Unitary partitioning approach to the measurement problem in the variational quantum eigensolver method," *J. Chem. Theory Comput.* **16**, 190–195 (2019).
- ⁹⁴R. J. Hinde, "A six-dimensional H₂–H₂ potential energy surface for bound state spectroscopy," *J. Chem. Phys.* **128**, 154308 (2008).
- ⁹⁵Z. Cai, "Resource estimation for quantum variational simulations of the Hubbard model," *Phys. Rev. Appl.* **14**, 014059 (2020).
- ⁹⁶H. Abraham (2019), "Qiskit: An open-source framework for quantum computing," Zenodo. <https://doi.org/10.5281/zenodo.2562111>
- ⁹⁷O. T. Unke, S. Chmiela, H. E. Sauceda, M. Gastegger, I. Poltavsky, K. T. Schütt, A. Tkatchenko, and K.-R. Müller, "Machine learning force fields," *Chem. Rev.* **121**, 10142–10186 (2021).
- ⁹⁸A. Singraber, J. Behler, and C. Dellago, "Library-based LAMMPS implementation of high-dimensional neural network potentials," *J. Chem. Theory Comput.* **15**, 1827–1840 (2019).