

## Density functionals and Kohn-Sham potentials with minimal wavefunction preparations on a quantum computer

Thomas E. Baker<sup>1</sup> and David Poulin<sup>1,2,3</sup>

<sup>1</sup>*Institut quantique & Département de physique, Université de Sherbrooke, Sherbrooke, Québec J1K 2R1 Canada*

<sup>2</sup>*Quantum Architecture and Computation Group, Microsoft Research, Redmond, Washington 98052, USA*

<sup>3</sup>*Canadian Institute for Advanced Research, Toronto, Ontario, M5G 1Z8 Canada*



(Received 12 August 2020; accepted 12 October 2020; published 16 November 2020)

One of the potential applications of a quantum computer is solving quantum chemical systems. It is known that one of the fastest ways to obtain somewhat accurate solutions classically is to use approximations of density functional theory. We demonstrate a general method for obtaining the exact functional as a machine learned model from a sufficiently powerful quantum computer. Only existing assumptions for the current feasibility of solutions on the quantum computer are used. Several known algorithms including quantum phase estimation, quantum amplitude estimation, and quantum gradient methods are used to train a machine learned model. One advantage of this combination of algorithms is that the quantum wavefunction does not need to be completely re-prepared at each step, lowering a sizable prefactor. Using the assumptions for solutions of the ground-state algorithms on a quantum computer, we demonstrate that finding the Kohn-Sham potential is not necessarily more difficult than the ground-state density. Once constructed, a classical user can use the resulting machine learned functional to solve for the ground state of a system self-consistently, provided the machine learned approximation is accurate enough for the input system. It is also demonstrated how the classical user can access commonly used time- and temperature-dependent approximations from the ground-state model. Minor modifications to the algorithm can learn other types of functional theories including exact time and temperature dependence. Several other algorithms—including quantum machine learning—are demonstrated to be impractical in the general case for this problem.

DOI: [10.1103/PhysRevResearch.2.043238](https://doi.org/10.1103/PhysRevResearch.2.043238)

### I. INTRODUCTION

Quantum computing has been proposed as an alternative to classical computing [1], and there are some problems which can be solved faster than known classical algorithms [2–7]. One of the most sought after and potentially far reaching applications on a quantum computer is the solution of quantum chemistry problems [8–13]. Obtaining exact solutions from a quantum computer efficiently could revolutionize modern applications including the creation of new medicines, fertilizers, batteries, superconductors, and more [14–20].

To do this, one important quantity to determine is the ground-state energy. The energy is a highly useful quantity for determining properties such as the equilibrium geometry of a molecule. Yet, the energy is not descriptive enough to fully characterize all desired properties of a system. For example, the band structure can be a useful tool to characterize a material, but this requires measurements at several  $k$ -points [21]. So, many measurements of the wavefunction would be required for some simple quantities.

Measurement on the quantum computer is expensive because the wavefunction must often be reprepared before a second measurement is performed. It has already been shown on a quantum computer that obtaining the wavefunction can be extremely costly [22–25], taking months or years even for moderately sized systems [24,25], and that the wavefunction cannot be copied [26]. The wavefunction is therefore a valuable commodity and measurements should be minimized [27].

One option to encode many solutions into one measurement is to use a machine learned (ML) model [28]. In general, ML models can interpolate remarkably well between input data to give access to many systems, including those not already solved. In principle, the ML model can be constructed directly on the quantum computer or from classical data generated by the quantum computer.

Using ML models would also allow for users of the quantum computer to export solutions to classical users. The results could then be quickly retrieved classically from the model and are generally accurate over a training manifold on which the model was constructed in a desire to generate the best machine learned model possible.

Finding the full wavefunction would require exponentially many measurements, so this can be difficult to implement on a quantum computer. But the same information can be expressed in a more compact form. So, we can look at alternative formulations of quantum physics for the most descriptive model.

Published by the American Physical Society under the terms of the [Creative Commons Attribution 4.0 International](https://creativecommons.org/licenses/by/4.0/) license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

The route pursued here is with density functional theory (DFT) [29]. Hohenberg and Kohn established that the one-body density,  $n(\mathbf{r})$ , is one-to-one with the external potential  $v(\mathbf{r})$  up to a constant shift of the potential. In essence, the density can replace the wavefunction, but it has fewer variables.

In order to use DFT, we must find some other means of obtaining the energy, since the Hamiltonian is not used in DFT. Instead, the universal functional,  $F[n]$ , must be found. It was proven that the universal functional exists and is common to all problems of the same electron-electron interaction [29,30].

The quantities required for the classical user to find self-consistent solutions are the exact functional (determining the energy) and the functional derivative [31]. So, in addition to finding  $F[n]$ , we also must find some other quantity such as the density,  $n(\mathbf{r})$ , or the Kohn-Sham (KS) potential,  $v_s(\mathbf{r})$  [30]. With these components, we can fully characterize a quantum ground state and solve for other measurable quantities.

It has already been established that the density functional can be successfully modeled with ML methods on the classical computer [31–45]. Exact quantities at several different external potentials must be found for the ML models to be trained. The number of training points needed to construct accurate models are not prohibitively large. From the ML functionals, self-consistent solutions can be obtained [38,46–48]. Numerically accurate ML functionals satisfy all exact conditions of  $F[n]$  [33,49] and escape the common errors of approximated density functionals [50,51].

To apply the classical ML-DFT methods on a quantum computer, some additional constraints must be minded. Previous attempts to obtain functionals from the quantum computer have relied on many measurements of the wavefunction for each system of interest [13,52–56]. In our view, a worthy goal is to avoid both excessive measurements and re-preparations of the wavefunction especially in the case of time-dependent quantities [53,57].

This work proposes a feasible algorithm that finds the ML model for  $F[n]$  on the quantum computer if a ground-state wavefunction is available. The algorithm leaves the wavefunction largely undisturbed so it can be used as the starting state for another system, greatly reducing the prefactor required to solve other systems. This is accomplished by using a state-preserving quantum counting [58,59] algorithm to extract descriptive quantities such as the density. Much of the algorithm is kept entirely on the quantum computer to motivate future improvements for speed, but the counting algorithm does allow for information to be output classically.

This algorithm is an alternative to running one very long computation and just measuring one energy, in that each step of the wavefunction preparation is proposed to solve another system. Thus, no step from the ground-state solver is wasted when using the algorithm here.

We also demonstrate that the Kohn-Sham potential can be solved using a similar strategy as the wavefunction. A gradient evaluated on a cost function for the KS system allows for the determination of the exact KS potential. This strategy can be faster than obtaining the density. Further, we demonstrate how access to the functional can be used to find approximate time- and temperature-dependent behavior in a system from the ground-state functional, and modifications can be added to obtain exact results.

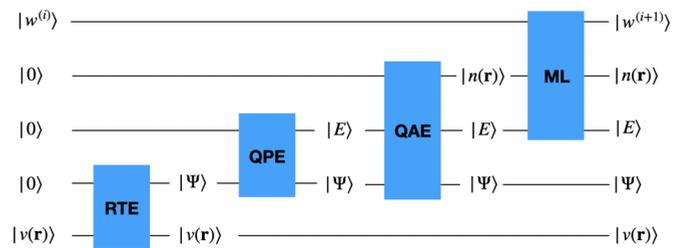


FIG. 1. One step of the RWMP algorithm for the density. The next iteration uses the output wavefunction as the starting state for the next system to reduce the prefactor. A similar procedure can be used to find the KS potential, replacing  $n(\mathbf{r})$  with  $v_s(\mathbf{r})$  (with a QGA as an oracle query for the QAE). The ML model may need inputs from other registers.

One temptation would be to use quantum machine learning, but long-known bounds on the efficiency of these methods preclude their use here [60]. This agrees with recent demonstrations that some known quantum machine learning algorithms are not universally advantageous [61–64]. We also discuss general limitations on known algorithms such as quantum machine learning and why these algorithms are expected to be inefficient here.

Section II presents the algorithm. Section III will discuss several uses of the resulting functional and considerations in choosing quantities to solve for. Section IV will discuss known limitations and justify why the algorithm is constructed as presented. Necessary background information on quantum chemistry, DFT, ML, and quantum computing algorithms is given in the Appendixes.

## II. ALGORITHM FOR THE FUNCTIONAL

In order to establish an algorithm for the quantum computer that gives results in quantum chemistry, knowledge of both fields must be understood. To avoid a lengthy summary in the main text, we have included relevant background knowledge in the Appendixes in case they are needed. Nearly all of the computational steps (e.g., machine learning the functional) have already been demonstrated by us in the references and the algorithms performed accurately. This section will contain all the elements of the algorithm and assumes only a background of algorithms in quantum computing.

We provide Fig. 1 to illustrate the steps necessary for one iteration of the algorithm, which we will refer to as a recycled wavefunction for minimal prefactor (RWMP) method. Although many quantities could be produced from this algorithm, we will focus on the components useful for the density functional. The inputs are the external potential for some system ( $|v(\mathbf{r})\rangle$ ) and initial guess weights for the ML model ( $|w^{(i)}\rangle$ ) represented as classical variables throughout. The following steps are required to obtain the solution for a given system and then update the parameters of the ML model.

(1) We prepare a ground-state wavefunction  $|\Psi\rangle$  for a given external potential,  $|v(\mathbf{r})\rangle$ , and number of electrons,  $N_e$ .

This can be done by real-time evolution (RTE; see Appendix D 4). In Fig. 1, this is denoted by a box for RTE. The

subroutine that obtains the wavefunction here does not have to be RTE. If another, more advanced solver is developed and used, then this can be substituted with no change to the rest of the RWMP algorithm.

Note that the methods to obtain quantities from the result of classical computations are not available since the quantum wavefunction has coefficients that are stored in superposition (a linear combination of  $|0\rangle$  and  $|1\rangle$ ). This means that the coefficients of the wavefunction cannot be found except with many measurements. We must first find the energy before obtaining other relevant quantities.

2. We obtain the ground-state energy  $|E\rangle$  from the ground-state wavefunction  $|\Psi\rangle$  given in the previous step.

Access to the ground-state energy is provided by quantum phase estimation (QPE, Appendix D 2) or with some other method like qubitization [65]. On Fig. 1, this step is denoted by QPE.

The next task is to determine some quantities of interest without requiring a full measurement of the wavefunction. The counting algorithm used allows for the wavefunction to be used again on the next iteration.

3. Given the energy  $|E\rangle$  and wavefunction  $|\Psi\rangle$ , we generate some quantity that is denoted here as  $|n(\mathbf{r})\rangle$  from a quantum amplitude estimation (QAE; Appendix D 5), a name which we use interchangeably with quantum counting throughout.

The symbol used,  $n(\mathbf{r})$ , is for the density (Appendix B), but we can substitute this quantity for others. For example, one could also determine the KS potential  $|v_s(\mathbf{r})\rangle$ . This step is denoted as QAE in Fig. 1. Note that this step may involve an oracle query such as a quantum gradient algorithm (QGA; Appendix D 3) or have other subroutines. This is a point we will expand on in the next section, Sec. II A, when discussing how to obtain either  $n(\mathbf{r})$  or  $v_s(\mathbf{r})$ . For now, we focus on what to do once the quantity is obtained.

The output wavefunction is slightly modified by the QAE but remains nearly the same state with a small amount of error. The procedure to return the wavefunction to its original state does not completely reprepare the wavefunction and instead has an iterative set of steps to repair the wavefunction as explained in Ref. [58] (see also Appendix D 5).

4. The output of the QAE can be used to update the ML model's parameters  $|w^{(i)}\rangle$  to the next iteration,  $|w^{(i+1)}\rangle$ . More than one ML model can be trained here (e.g., a ML model for  $|E\rangle$  and for  $|n(\mathbf{r})\rangle$ ).

The typical steps in a stochastic gradient descent (SGD, Appendix C) of forward and backward propagation can be used. For backward propagation, the output of a QGA can be used to update the ML parameters. This step is marked as ML on Fig. 1. This ML operation may need to be controlled on  $|v(\mathbf{r})\rangle$  depending on which quantity is being trained. Note that the QAE output could also be stored classically and then machine learned not on the quantum computer, skipping this step but eliminating the opportunity for an improvement in the ML box from quantum advantages.

5. Another external potential is provided ( $|v + \Delta v\rangle$  (not shown in Fig. 1) and the wavefunction is reused as the starting state for the next RTE.

Here,  $\Delta v$  is chosen by the user and simply added to the coefficients of  $v(\mathbf{r})$  to give the next potential. Many passes

through a set of potentials must occur to obtain an accurate ML model.

Recall that the resource estimate given in Ref. [24], several months may be required for a small molecule with RTE. The RWMP algorithm allows for a sequence of intermediate systems to be visited, effectively making use of that time to obtain data. So, if the system starts in a configuration where the initial wavefunction is accurate, then this can become the first data point for the ML model. Each subsequent time step could be another potential data point for the model.

The second advantage to this strategy is that the sequence of potentials in the RWMP algorithm can allow for the ordering of the next potential to be close by. This allows for the best starting state for the next system to be used and reduce the total amount of RTE steps that must be run over all systems. In summary, the RWMP algorithm here could make use of the preparation time for a hard to solve system by finding data from the intermediate steps and reduce the prefactor in the solution.

The RWMP algorithm repeats until all systems are visited. The final step is to measure the parameters of the ML model,  $|w\rangle$ . The model can then be used classically.

In what follows, we discuss which quantities should be obtained by the QAE for the best description of the ground state. Then in Sec. III, we discuss aspects of this strategy that were not absolutely necessary in defining the algorithm and how a classical user can use the result. Finally, we expand on many points that were not crucial for the basic understanding of the RWMP algorithm and explain limits that ultimately lead to this algorithm (and why other subroutines were not used) in Sec. IV.

## A. Quantities of interest

We move on to discussing which quantities are best to determine from the wavefunction via the QAE. There are two main options: the coefficients of the density matrix and the KS potential. When choosing the quantities of interest, both the functional and the functional derivative must be determined with the ML model in order to find solutions on the classical computer. There are several types of functionals that can be trained for this, some of which are presented here.

### 1. Density functionals

In the RWMP algorithm, one option is to find the density,  $n(\mathbf{r})$ , since this is proven to be a suitable replacement for the wavefunction from the Hohenberg-Kohn theorem in Ref. [29]. The  $N^2$  elements of the density matrix are expectation values of the operator  $\hat{c}_i^\dagger \hat{c}_j$  (not just diagonal elements; see Appendix A 1) where  $\hat{c}$  is a fermionic operator defined in Appendix A 1. Spin indices are ignored for simplicity for now. Since the expectation value is on the interval  $[-1, 1]$ , we can use the operator  $(\hat{c}_i^\dagger \hat{c}_j + 1)/2$  (defined on  $[0, 1]$ ) and afterward shift the result back to the original interval. Using this shifted operator is a necessary component to using the QAE because the expectation value can now be related to a probability. The number of rounds required for the QAE relates to the inverse of the probability of failure requested [58].

There are several options for the ML model. The ML step can train directly from  $n(\mathbf{r})$  to  $E$  or we can take as an input

$v(\mathbf{r})$  to the model and train both  $n[v](\mathbf{r})$  and  $E[v]$ . The first option gives a pure density functional (Appendix B). The second option gives a potential functional, which is a dual functional to DFT (Appendix B 2) [66]. Both of these theories can be solved self-consistently. One can also train the bifunctional  $E[n, v]$  [34].

## 2. Kohn-Sham potentials

The other main quantity, which has more value, is the KS potential,  $v_s(\mathbf{r})$ . The defining feature of the KS system is a noninteracting system that has the same density as a given interacting system (see more discussion and ways to realize this potential in Appendix B 1). The potential defining this noninteracting system is defined as  $v_s(\mathbf{r})$  which is described by  $N$  parameters.

Two potentials can have the same density if the systems do not have the same electron-electron interactions, so as not to violate the Hohenberg-Kohn theorem [29]. The KS potential is highly valuable since it can be applied in several other instances. This includes finding time and temperature-dependent calculations from  $v_s(\mathbf{r})$  or the KS band structure [57,67,68].

A central question is whether  $v_s(\mathbf{r})$  exists for a given interacting system. This is known as the problem of  $v$ -representability. Since it is proven that the KS potential always exists on a lattice,  $v_s(\mathbf{r})$  always exists here [69–72].

The KS potential must be converged to, just as we had to evolve an initial state in RTE to a final state. The difference in finding the KS potential is that a gradient is applied instead of a time evolution operator for the wavefunction.

The KS potential,  $v_s$ , will satisfy the minimization [73–75]

$$\min_{v_s} (\langle \Psi[v] | \hat{T} + \hat{V}_s | \Psi[v] \rangle - \langle \Phi[v_s] | \hat{T} + \hat{V}_s | \Phi[v_s] \rangle) \quad (1)$$

where  $\Psi$  is the interacting wavefunction and  $\Phi$  is the KS wavefunction (see Appendix B 1 d for more information). There are other methods to obtain the KS potential [76–79], but the method used here is straightforward on a quantum computer given a close enough starting guess or small enough molecule (i.e., those solvable by RTE). Many other methods to find the Kohn-Sham potential would either require numerous measurements of the wavefunction or large overhead in terms of qubits for operations that are simple on the classical computer but costly on the quantum computer including addition, division, etc. [80].

Equation (1) is used as the output of the oracle query in QAE for the QGA [81,82]. Note that the QGA is particularly useful for finding the functional derivatives here, notably taking the variation of all possible  $v_s(\mathbf{r})$  in one oracle query [81]. The resulting gradient is applied on the coefficients of the KS potential and the process is repeated sufficient times until the true KS potential is obtained. An initial guess for the parameters could be taken from existing semilocal approximations such as local density approximations [83,84] or using a classical method [72–79,85]. The other classical methods where a gradient is used to evolve the potential may be useful, but the density does not need to be constructed to use Eq. (1).

In order to construct Eq. (1) on a quantum computer, the eigenvalues of free Hamiltonians, such as the KS Hamiltonian ( $\hat{T}_s + \hat{V}_s$ ) can be mapped to the interval  $[0,1]$  for the QAE. The

operator must also be scaled by a constant, but we also note that shifting the potential by a constant,  $v_s(\mathbf{r}) \rightarrow v_s(\mathbf{r}) + \mathcal{C}$ , is allowed to ensure all eigenvalues are positive without changing the eigenvectors [29]. Further, identifying some upper bound on the expectation value,  $\varepsilon_{\max}$ , the scaled operator could appear as  $\langle \Psi | \hat{T}_s + \hat{V}_s + \mathcal{C} | \Psi \rangle / \varepsilon_{\max}$ .

The expectation value  $\langle \Phi | \hat{T}_s + \hat{V}_s | \Phi \rangle$  can be evaluated in one of two ways. First, it can be computed by diagonalizing  $\hat{T}_s + \hat{V}_s$  (determining  $h_k$ ) and taking the sum  $\langle \tilde{\Phi} | \sum_k h_k \hat{c}_k^\dagger \hat{c}_k | \tilde{\Phi} \rangle$  in the diagonalized basis with  $\tilde{\Phi}$  is now used. This procedure uses QAE to find the expectation value analogously to finding the coefficients of the density. Alternatively, one could apply a QPE (or qubitization) directly for the KS system, noting that the gate count is drastically less for the noninteracting system.

The KS potential as encoded in the ML model can be expressed as either  $v_s[v](\mathbf{r})$  or as  $v_s[n](\mathbf{r})$ , where both are well defined [86].

## B. Example for the Kohn-Sham potential

To illustrate further some of the more abstract quantities in the RWMP algorithm in Sec. II, we provide an expanded example of the RWMP algorithm of how to obtain the ML-KS potential.

(1) An initial potential  $v(\mathbf{r})$  is chosen.

This can be done by assuming the external potential is a set of nuclei with a Coulombic interaction,  $v(\mathbf{r}) = \sum_a (-Z/|\mathbf{r} - \mathbf{r}_a|)$ , where  $a$  indexes the positions of the nuclei,  $\mathbf{r}_a$  with atomic number  $Z$ .

(2) A basis set is chosen,  $\varphi_k(\mathbf{r})$ .

The model can then be discretized in this basis and the resulting model has fermionic operators (see Appendix A 1).

(3) RTE is run and the ground state is obtained.

(4) QPE is used to find the ground-state energy,  $E$ .

(5)  $\Psi$  and  $E$  are used in the QAE to find the first term in Eq. (1). The energy of the noninteracting system is also obtained for the second term in Eq. (1) using some initial guess for  $v_s(\mathbf{r})$ . This constructs the oracle in the QGA.

(6) The QGA result is added to the coefficients of the KS potential and the last step is repeated until the  $v_s(\mathbf{r})$  a certain number of times to find the minimum.

(7) The values  $E$  and  $v_s(\mathbf{r})$  are input into a ML model. The gradient of the parameters in the model are updated with a QGA by repeatedly computing the gradient and adding it to the current coefficients.

We can also use the QGA to output  $E$  and  $v_s(\mathbf{r})$  to the classical user and learn the final set of potentials classically.

(8) A new set of atomic coordinates are provided,  $\mathbf{r}'_a$ , the difference  $\Delta v$  between the previous and this potential is computed, and the result is added to the old potential.

We now have the next potential and can start again at step 3 until all systems are visited.

Note that this computation can be restarted at any time (at the significant cost of re-preparing the wavefunction). Note that it has been assumed that all systems are run for the same basis set, although this condition could be relaxed in principle.

One advantage of using this method is that the Kohn-Sham potential is characterized by  $N$  coefficients  $\kappa_k$ ,  $v_s(\mathbf{r}) = \sum_{k=1}^N \kappa_k \varphi_k(\mathbf{r})$ . This is a factor  $N$  less than the density

required. The evolution of the Kohn-Sham potential from an initial guess potential by gradients is very similar to how the wavefunction is evolved with RTE.

### III. ADDITIONAL CONSIDERATIONS AND USE OF THE FUNCTIONAL

In the previous section, an RWMP algorithm for finding the density functional and Kohn-Sham potential were detailed. It was also noted that other variations of the density functional could be found using the same technology.

Several points that could extend the RWMP algorithm—and other details about what was introduced—are discussed here. These include resource costs, a comparison between finding the density and the KS potential, the universality of the functional, how the functional can be used, applications to other types of functional theories, and opportunities for near-term studies. Some points relating to a quantum advantage that are discussed here are continued in Sec. IV A.

#### A. Scaling and resources required

The algorithm will require a fully scalable quantum computer, probably with self-correcting memory. Near-term examples are available for density functionals, and a full discussion of the known features of finding the density functional with this method is given here.

##### 1. Algorithmic scaling

The scaling of the RWMP algorithm for the density functional in terms of the number of basis sets is  $O(N^2)$  for one system asymptotically in the number of basis functions due to time evolution. The actual complexity in practice is between  $O(N^2)$  and  $O(N^4)$  for intermediate system size (see Appendixes A 2 and D 4). The other subroutines scale at most as  $O(N^2)$  (see the Appendixes).

##### 2. Prefactor for convergence of the wavefunction

Even though the scaling in terms of the number of basis functions is polynomial, the actual cost is significant due to a prefactor [24].

The prefactor of the algorithmic scaling is problem dependent and will be the dominant cost to obtaining a ground-state wavefunction. The prefactor will depend on the time used to prepare  $\Psi$ , the number of steps that the QAE algorithm must be run, and the required number of systems to be visited. Note that the number of steps that must be run to obtain the correct time evolution is dependent on how close the starting state is to the system's solution, the number of electrons, how strongly correlated the electrons are, and many other variables.

Note that if the first system in the RWMP algorithm is exactly equal to the initial wavefunction (e.g., the well-separated limit for a neutral molecule where each separated piece is a hydrogen atom with one electron and where Hartree-Fock is exact for one-electron systems), then the time to make the first solution is zero and each subsequent motion of the atoms closer together will be another data point that may not require too many time steps to obtain.

#### 3. Convergence of the machine learned model

In order to aid the convergence of the ML model, it would be useful to have several quantum computers running at once. This would mean that more than one system can be used to construct a mini-batch update for the cost function of the gradient descent, making the resulting update to the ML model more accurate. One also can save the output quantities as future training points at the cost of extra registers. Once an accurate ML model is generated, the problem does not need to be solved again.

#### 4. Resources required

The RWMP algorithm presented in this work requires  $4 + \zeta$  qubit registers for  $\zeta$  quantities of interest that are not the energy [e.g., learning  $E$  and  $v_s(\mathbf{r})$  implies  $\zeta = 1$ ]. There will also be overhead for the QAE and other parts of the RWMP algorithm. One can also divide the parameters in the ML model into separate registers for each additional quantity of interest since the model spaces between them should be sparse.

The number of qubits required is clearly large and dependent on the specific steps of the RWMP algorithm used. This means the RWMP algorithm is best suited for a fully working quantum computer that is expected in the future. There are many ways to reduce the steps necessary. For example, in one way, by outputting the QAE to a classical user. However, we want to maintain the flexibility that a quantum advantage could be realized here as discussed next in Sec. III B.

#### B. Structure of the neural network

In a neural network, the form of the nonlinear function,  $\mathcal{S}$  (see Appendix C), is often chosen so that it is easily differentiable. On the quantum computer, determining the gradients are accomplished in one oracle query (see Appendix D 3), so the traditionally used functions on a classical computer (e.g., sigmoids) can be swapped out for another function that may lead to better performance or faster convergence. It is not clear if this will produce any detectable advantage.

The number of coefficients required to obtain an accurate ML model can be very high, but some guidelines can reduce this number for a given quantity [87]. For the problem at hand, note that the training is done in the same basis as the problem is expressed. In this case, the connectivity of the neural network can be constrained on physical arguments. Known bounds on the structure of local correlations as proved by Hastings [88] apply to densities that was originally shown by Kohn *et al.* [89,90]. This means that perturbations on the density decay exponentially with distance for gapped systems and as a power law for gapless systems [91]. Then, the neural network models for the density do not need to account for arbitrarily long ranged connections and can remain local. In essence, the connectivity of edges on the graph for the neural network would be similar to the structure of a multi-scale entanglement renormalization ansatz [92] (scaling as  $N \log N$ ) but in three dimensions. Note that this estimate is a minimum, and it may still be advantageous to add more hidden layers or connectivity. Note that we will expect that gradient-free

methods (Appendix C 1) will not necessarily be more useful in training the ML model.

Not all quantities will have a local structure. The  $v_s(\mathbf{r})$  is fully nonlocal due to the Hartree potential (see Appendix B 1 a) and can require all-to-all connectivity between subsequent hidden layers in the neural network.

Note that it is not required to train the ML model on the quantum computer. One can determine the quantities of interest for each system and output the results with the QAE from the quantum computer to the classical computer [58]. But we give the option here to train the neural network on the quantum computer in case a quantum advantage can be realized. Using the RWMP algorithm as presented in Sec. II would still avoid excessive measurement; meanwhile, outputting to classical variables will reduce the amount of overhead needed to store ML parameters.

### C. Comment on the universal versus exact functional

ML functionals are very accurate over the training manifold on which they were constructed [47]. We have chosen the appropriate adjective (exact or universal) describing the functional very carefully in each use here. The procedure we describe here is not truly universal since the accuracy of the ML functional is limited to the training manifold of potentials that we have explored for the ML model. For example, trying to solve a model trained for solutions with  $N_e$  electrons with a new potential in the training manifold with  $N_e + 1$  electrons will try to project the solution back onto the  $N_e$  solutions.

The resulting ML functional can be numerically exact, however, for a given problem. This functional will be called the exact functional, implying that it is accurate for some systems but not all possible systems.

To complete the discussion, one can have a universal but not exact functional. For example, if one estimated the energy to be a fixed value for all systems, this would be universal but not very useful.

### D. Comparing the search for the density and Kohn-Sham potential

To actually compare the true cost of the  $O(N^2)$  operations to find the density with the  $O(TN)$  coefficients for the KS potential, one would need to know the number of times  $\mathcal{T}$  a gradient must be applied. This depends on the system studied and starting KS potential. It is therefore not clear as a general guideline if directly finding the coefficients of the density matrix is always better than starting from a KS potential that is close and applying the gradients. The key difference in the two cases is that the KS potential relies on a suitable starting state and the density does not. The assumption for the KS potential—that a good starting state is required—is very similar to the restriction on RTE itself to find the ground state, so all assumptions are consistent.

The fact that the KS potential is competitive with finding the density here is more a comment on the overhead required for the density (which is far greater than on the classical computer) than the KS potential being any easier to find. There is one advantage to using the minimization for  $v_s(\mathbf{r})$  here in that the QGA is much more efficient than the classical computer.

In addition to comparing the scaling, the nature of the KS problem requires a large basis set to obtain the proper KS potential and avoid any Gibbs oscillations that would appear in a truncated basis set [78]. So, the  $N$  required for an accurate density or energy might be smaller than for an accurate KS potential in practice.

Regardless, any effort to find  $v_s(\mathbf{r})$  is worth it since  $v_s(\mathbf{r})$  is far more descriptive and gives access to useful quantities.

### E. Use of the functional

The algorithmic cost in Sec. III A is paid only once. When the model is given to a classical user, a separate cost must be paid to solve it for the ground state. Given the ML model, the classical user can solve the ML functional self-consistently by determining the Euler-Lagrange equation to minimize the functional (see Appendix B 1 f). This requires that a projection back into the training manifold must occur to ensure the functional derivative is accurate [34,46]. This projection is estimated to scale as  $O(N^2)$  but can be machine learned separately to speed up computation [33,34,34,46].

Finding the pure density functional has better scaling for the classical user, scaling like the number of basis functions,  $N$ . Note that even though we can learn  $F[n]$ , the external potentials (as well as particle numbers and polarizations) over which the ML model is accurate must also be given to the classical user so it is understood what the training manifold is.

The KS system scales as the cube of the number of basis functions formally,  $N^3$ , since it is a noninteracting problem. Note that when solving the KS system with the exact functional, convergence is proven [72,93].

If a bifunctional,  $E[n, v]$ , is used, then a self-consistent solution is not necessary [34].

Once the model is trained sufficiently accurately, one can always retrain for another type of functional (e.g., from the KS potential, the density can be obtained and a pure density functional can be trained).

Finding the density functional—and using it to compute quantities—is preferable to generating a library of system properties and machine learning those properties. From the DFT model, the other quantities can be constructed.

As an example application, if trained over enough external potentials, this method could efficiently evaluate molecular dynamics problems giving accurate results on laptops instead of supercomputers [94].

### F. Other types of functional theories

There are many other types of functional theories that are related to DFT that can be obtained with these methods. The most easily extended method is density matrix functional theory (DMFT) [95]. Since our method of obtaining the density was to actually obtain the coefficients of the density matrix (see Appendix A), the density matrix functional is learned.

Extensions of DFT can also be solved with this method including the motion of nuclei,[96] time dependence (TD-DFT) [57,97–99], thermal properties [100], superconducting functionals [101,102], quantum electrodynamics-DFT [103–105], ensembles [106,107], and others

[66,108–112]. In each case, the solution on the quantum computer is modified in some way. For example, a superconducting functional theory can be found if the pairing potential [113] is also computed and learned.

Two of the methods in the above list (time- and temperature-dependent methods) deserve extended discussion in the subsequent subsections since both exact and approximate functionals can be found from both. Note that the approximate methods only require the ground-state functional in both cases.

### 1. Time evolution

Given an accurate enough ML approximation for the KS potential, one can time evolve the system [44]. If the system is adiabatically time evolved with a weak enough perturbation, this would be available immediately from just the KS system at zero time [see Eq. (B25) in Appendix B 3]. The adiabatic approximation is often sufficient for many physical processes.

For time evolution beyond the adiabatic limit, an exchange-correlation kernel denoted as  $f_{xc}$  (see Appendix B 3) would need to be found. Full evaluation of  $f_{xc}$  could be accomplished with the QGA (Appendix D 3 a); however, one can estimate the kernel via back propagation in the neural network (see Appendix C). An accurate functional derivative will be required in this case [46]. One can also time evolve on the quantum computer and use expectation values at various times for other quantities [56,65,114,115].

### 2. Finite temperature calculations

For temperature-dependent processes, one approximation that is available from the ground-state KS potential is the Fermi-weighted KS technique [116]. In this method, occupied and unoccupied orbitals found at zero-temperature can be weighted with a Fermi-Dirac distribution to obtain a finite temperature density.

For exact computations at finite temperature, Appendix B 4 discusses the temperature-dependent KS potential (which is very similar to the ground-state case) and can be found if a finite temperature wavefunction is provided [114,117].

Other theories may require computation of other quantities, including those not relevant to quantum chemistry. Yet, they can follow the same strategy as the RWMP algorithm (i.e., reuse of the wavefunction and QAE) to find the relevant quantities.

### G. Nondensity functional quantities

Note that the RWMP algorithm is not specific to the density, the KS potential, or even a density functional. Many quantities of interest can be obtained. To see how the continued fraction representation of the Green's function can be obtained, see Ref. [59].

### H. Reduced examples for testing

Throughout, it is implied that many qubits will be required, but we view this as similar to requiring sufficient memory on the classical computer. However, there are test systems that are used to provide insight into DFT and may be useful for proofs of principle [118–126]. A simple model that may be

within reach of existing quantum computers is the two-site Hubbard model, which has been used to study simplified DFT [127–135].

Each individual algorithm has been tested in cases of relevant interest for other problems, many of which by the authors. Citations to many of the tests of these algorithms are included near their description.

## IV. LIMITATIONS OF KNOWN ALGORITHMS

This section discusses limitations on the types of algorithms that could have been used to machine learn the functional and how future improvements in algorithms must continue to improve the performance on quantum computers. We hope that a clear and complete discussion of the current hurdles for quantum computation motivate future algorithms and provide a consistent account.

### A. Feasibility of obtaining the starting state

In the previous section, we omitted a discussion of how to prepare the initial state in superposition and how it is converged. The time it takes to obtain a wavefunction is known to be inefficient with current techniques. For some algorithms, the ideal input would have been a superposition of solutions that would include all combinations of particle numbers and spin polarizations for all external potentials.

In this section, we consider the complications for even preparing a suitable state in superposition and known limits. We will discuss the RTE algorithm in the context of its scaling, why the large prefactor can prohibit its use on some systems, comparing the implementation of RTE on the classical and quantum computer, and limitations to constructing a superposition of all solutions with any method. This last point will address the feasibility of finding the original starting state for the QML.

Some other algorithms that were not used as subroutines in the RWMP method are also discussed.

### 1. General considerations for real time evolution

One of the primary motivations for making the RWMP algorithm for the density functional is to recycle the ground-state wavefunction, reducing the cost of RTE. It is true that RTE scales only polynomially (Appendix A 2) with the number of orbitals. In comparison, the full configuration interaction (FCI) gives the exact results but scales as  $O(N_e!)$  [136]. So, RTE has a scaling advantage over FCI; however, the prefactor matters.

The exact same RTE algorithm can be run on a classical computer. The reason that quantum chemistry computations are not run with RTE is that the prefactor equal to the number of time steps is large [24,137] This is especially problematic for large systems and those where a near-degeneracy must be avoided, necessitating a small step size. But this is exactly where quantum computers are hoped to be applied.

Note also that the Trotterization of the time evolution operator is suited for planar molecules since interactions are comparatively localized, which is the same reason that matrix product states prefer these geometries [91,138,139] further limiting the usefulness of RTE.

Further comment becomes far more complicated because these alternative, smaller wavefunction ansatzs on the classical computer may display all the same features of the true ground state and accurate energy. These solvers typically have systematic errors that are studied, can solve systems faster, and have led to solutions of large systems [140] and new discoveries [141]. So, it is not clear if a quantum computation can beat all classical representations in terms of efficiency. Note that on the classical computer, another solver can be used (e.g., tensor networks [91,138], quantum Monte Carlo [142], random phase approximation [143,144], or many, many other methods [145]). The number of choices here is vast and storied, so we leave more discussion for others [145,146].

In summary, the time complexity of solving the quantum computation should be expected to be larger than classical solvers. Quantum computers can represent a reduction in the amount of space required to store a wavefunction, but it is not clear if this will always beat every classical representation. Improved methods of finding the ground state would be highly valuable.

## 2. Choice of basis sets for algorithms

It is true that the quantum computer provides a different representation of the wavefunction. In some representations for classical algorithms, the memory will grow considerably with system size. For example, a coupled cluster calculation can have many coefficients as in the number of operations may become exponentially large [147]. This is due to the need to store more coefficients for a given basis. The quantum wavefunction on the quantum computer may have some advantage here since coefficients are stored in superposition on each qubit. However, there are many wavefunction ansatzes to consider in comparison because some methods can find accurate answers with considerably less coefficients, and the time needed for the quantum computer's wavefunction can be lengthy.

Some recent efforts on the quantum computer have sought to impose specific conditions to reduce the amount of operations required. The strategies that are pursued are to use different basis sets and generate criteria for removing terms from the Hamiltonian. For example, plane waves lead to a sparse Hamiltonian (see also Appendix A 2) [148]. Such systematic methods for this should be expected to be as difficult as (or more so than) solving the wavefunction outright.

The proper comparison between RTE on a quantum computer with a plane-wave basis set and RTE on a classical computer is that that the classical computer can handle any basis. So, a comparison with a plane-wave basis set on a quantum computer should be compared to a RTE on the classical computer with some other basis set (e.g., Gaussians). The difference in the number of orbitals required to accurately simulate matter for plane waves can be much higher than other basis sets due to cusps in the wavefunction that require large numbers of plane waves to resolve [145,149–151]. So, restricting the basis to plane waves at best matches the classical equivalent in terms of operations required (see also Appendix A 2).

Wavelets have appeared in some references recently suggesting that these functions can provide a systematic way

to compress a Hamiltonian, but in the 30-year existence of these functions (for more information, see the references in Ref. [152]), they have been demonstrated to scale poorly to large system sizes due to the curse of dimensionality [153]. These methods typically give access to three or fewer electrons when not approximating the Fock operator. To use these functions, a compression ratio of nearly 100% would be required. Using these functions beyond one dimension faces significant hurdles in the general, interacting case.

In conclusion, restrictions to a particular basis set or procedures to remove terms in a Hamiltonian is not a cure-all for computation in general. The task of identifying which terms of the Hamiltonian without first solving the problem is typically complicated, and the resulting answer is not necessarily exact anymore. Approximated calculations are essentially the strategy of classical methods which leave out some effect or terms, and it is not clear how to do this systematically in general, nor if any simple strategy can be expected. A single method or change in basis is not a panacea to making the solution on a quantum computer less complex.

## 3. Limits on solutions in superposition

With regard to applying any generic algorithm to find a superposition of solutions, we can place a limit on finding the initial state required for algorithms that need this superposition of solutions.

*Theorem IV.1.* A quantum computer cannot efficiently generate a superposition of solutions for all potentials necessary for  $F[n]$  unless BQP=QMA-complete.

*Proof.* It is also known that at least one of the systems contained in the universal functional is in the computational class QMA-complete to solve [154]. This is not efficient on the quantum computer to compute objects in this complexity class. So, no algorithm should be able to obtain all solutions efficiently. ■

Thus, some elements should be expected to be unconverged in a superposition unless the algorithm is run for an impractically long time. Note that algorithms requiring a superposition generally require more than one solution, so the superposition is not run just once.

## 4. Alternative algorithms to real time evolution

We do not rule out that some improvement may allow RTE (or some other method) to receive a quantum advantage. If a superior algorithm is developed (e.g., the tools exist to make imaginary time evolution [155], perturbation theory [156], preparation of projected entangled pair states [157], or a very expensive version of the density matrix renormalization group [138,158,159] at the present time), it is likely that it will rely on converging from some initial state. Also, any algorithm like exact diagonalization for general systems is not expected to be efficient since this problem is not contained in the BQP complexity class [154,160,161]. This creates more motivation to focus on algorithms that converge.

We do note that progress through the decades on quantum chemistry has been difficult to find a unifying principle that would help in algorithm design [146], but perhaps quantum computing may motivate a new way to look at the problem

for cases of interest since the prospects of finding a general algorithm are prohibitive.

### B. Other methods and quantum machine learning

In regards to other methods to train the ML model, we had investigated using alternative subroutines using a superposition of solutions. Algorithms that we considered included Grover's algorithm [4,7], quantum walks [162,163], and others. Each of these requires a solution of a superposition of systems, some means of identifying the correct solutions through a phase kickback, undoing the superposition of systems, and then repeating the process until the error is low enough to ensure the correct solution is determined to some high probability.

The limitations described in Theorem IV.1 are one hurdle, but in our view, these strategies of uninformed search were too lengthy even on the quantum computer. Since the improvement in the number of steps required is only by a square root factor, searching the exponentially sized database causes the algorithm to run for far longer than other classical algorithms for electronic structure.

The variational quantum eigensolver (VQE) [164] might be adapted to finding, for example, the Kohn-Sham potential with a method from Refs. [72–79], but it is not clear if errors can be kept small in a reasonable amount of time, and we wish to keep focus on finding the exact KS potential. The VQE would also require the exact density from another method, but it could in principle be adapted. The main question is how well this suggestion would perform on a quantum computer based on current hardware.

#### 1. Quantum machine learning

Quantum machine learning (QML) algorithms were also not suitable since known bounds on the number of oracle queries imply that there is only a polynomial speedup for QML algorithms. See Ref. [60] and experimental proof on a simple case in Ref. [165]. A confirming statement is also found in Ref. [166].

Lacking an exponential speedup, it is likely too expensive for the quantum computer to run in a reasonable amount of time here.

Existing statements in the literature on the hardness of determining the universal functional can lead to limits on the types of QML algorithms we expect can exist. We formalize the relevant limits in some statements here [167].

*Theorem IV.2.* No QML algorithm can discover the universal functional in polynomial time unless QMA-complete reduces to the complexity class BQP.

*Proof.* The functional is proven to be QMA-complete to learn [161]. If an algorithm determined the functional in polynomial time, then BQP=QMA-complete. ■

Note that this says nothing about whether QML can do slightly better than the classical algorithm, but typically a step in a QML algorithm is to reprepare the wavefunction and this is one of the main issues we want to avoid here. Because the functional is known to be QMA-complete to learn, finding the exact and universal functional with the RWMP algorithm would require an exponential amount of time to visit all systems, as expected [154,160,161,168]. We continue on to make a connection with some statements about learnability.

Theorem IV.2 also implies limits on how learnable the universal functional is by any method. In Ref. [60], under the probably approximately correct (PAC) [169] model of ML, only a polynomial reduction in oracle queries (training points) is possible with QML.

*Lemma IV.3.* The assumed limitations on the number of oracle queries (quantum and classical learning differ by only polynomial factors) required to discover  $F[n]$  with QML are the same as those under the PAC model [60].

This is a consequence of the hardness of finding the functional. If this were not true, we could find a QML algorithm that could discover  $F[n]$  in exponentially fewer steps, which is a violation of both Theorem IV.2 and Ref. [60]. So, there can be no exponential speedup for QML under the PAC model here.

While there may be cases that lie outside of the PAC model [170–174], we have no evidence that the functional is not subject to the PAC assumptions. There is also good evidence to suggest that simple systems—at least—obey the limits of the PAC model [165].

In summary, the difficulty of finding the universal functional on a quantum computer places limits on the ability for many-body solvers on the quantum computer and QML as well. Note the generality of the statements here for all QML algorithms. Recent works have shown that some existing QML algorithms are not as efficient as once thought [61,62], but the arguments here apply to QML in general for this problem.

These general arguments do not prohibit a quantum advantage if another algorithm can be found to solve systems in a more specific case or restricted class of systems. Recent progress on finding classical algorithms that are superior to quantum algorithms illustrate the need for caution when proposing an efficient QML algorithm, however [64]. Still, QML does not seem to be a feasible way forward for the problem of interest here.

### C. Summary

The main take-away from these statements is that an exponentially more efficient algorithm for the most general case is ruled out when discussing the solution on a quantum computer. Reducing the prefactor, therefore, becomes highly beneficial. This does not preclude algorithms on more restricted systems, but there is no hint of how exactly to construct such an algorithm or that this is any easier than a straightforward solution.

The RWMP method avoids repeated measurement, reduces the prefactor to solve each system iteratively and allows for more systems to be solved with RTE or some other method.

## V. CONCLUSION

It has been demonstrated that a combination of algorithms applied to a wavefunction on the quantum computer can yield the Kohn-Sham potential, energy, and density matrix coefficients without completely re-preparing the ground-state wavefunction each time. The determined quantities can be used to train a machine learned model using gradient-based methods either on the quantum computer or classically. The ground-state wavefunction was used as the starting point for

the next system, reducing the prefactor and avoiding an expensive computation of the ground state at each step. This efficiency was also used for the Kohn-Sham potential with a minimization condition.

Once a model is created, a classical user can extract the relevant quantities from the machine learned model and use it for ground-state, time-, and temperature-dependent calculations. Finding the Kohn-Sham potential is especially useful here since it gives access to many properties of the ground state; in addition, there was some indication that the Kohn-Sham potential might scale better in some cases as opposed to finding the density. Known limitations on the complexity of finding the universal functional and quantum machine learning have constrained the choice of subroutines in the algorithm here. A better method to solve for the ground state on the quantum computer must be a focus of future research to make quantum chemistry studies feasible, but this algorithm will allow for solutions to be exported to many users.

## ACKNOWLEDGMENTS

T.E.B. acknowledges funding provided by the postdoctoral fellowship from Institut quantique and Institut Transdisciplinaire d'Information Quantique (INTRIQ). This research was undertaken thanks in part to funding from the Canada First Research Excellence Fund (CFREF). We thank useful discussions at the 2018 New Trends in Quantum Error Correction workshop at Université de Sherbrooke. The authors are thankful for useful discussions with Frank Verstraete, Guillaume Duclos-Cianci, Anirban Narayan Chowdhury, Jonathan A. Gross, Colin Trout, Li Li (李力), Yehua Liu, Vamsee Voora, Shane Parker, Raphael Ribeiro, Agustin Di Paolo, Anirudh Krishna, Maxime Tremblay, Jessica Lemieux, Benjamin Bourassa, Stuart Clark, Rex Godby, and Nikitas Gidopoulos.

## APPENDIX A: QUANTUM CHEMISTRY

In this section, we review some background information on quantum chemistry.

### 1. Many-body problems

The problem of interest is to solve the many-body problem expressed by the Hamiltonian [175]

$$\mathcal{H} = \sum_{ij\sigma} \left[ t_{ij} \hat{c}_{i\sigma}^\dagger \hat{c}_{j\sigma} + \sum_{kl\sigma'} (V_{ijkl} \hat{c}_{i\sigma}^\dagger \hat{c}_{j\sigma'}^\dagger \hat{c}_{l\sigma'} \hat{c}_{k\sigma}) \right] \quad (\text{A1})$$

with fermionic operators  $\hat{c}$  on discretized lattice sites (or basis functions) indexed by  $i, j, k, \ell \in \{1, \dots, N\}$  (for  $N$  basis functions) with spin  $\sigma$ . Note the order of indices [175,176]. The one-electron integral is

$$t_{ij} = \int \varphi_i^*(\mathbf{r}) \left[ -\frac{1}{2} \nabla^2 + v(\mathbf{r}) \right] \varphi_j(\mathbf{r}) d\mathbf{r}, \quad (\text{A2})$$

which is the kinetic plus external potential terms. The two-electron integral is

$$V_{ijkl} = \frac{1}{2} \iint \varphi_i^*(\mathbf{r}) \varphi_j^*(\mathbf{r}') v_{ee}(\mathbf{r} - \mathbf{r}') \varphi_k(\mathbf{r}) \varphi_\ell(\mathbf{r}') d\mathbf{r} d\mathbf{r}', \quad (\text{A3})$$

where  $v_{ee}(\mathbf{r} - \mathbf{r}') = 1/|\mathbf{r} - \mathbf{r}'|$  for the case of a Coulomb interaction and that this expression assumes the orbitals for both spin-up and spin-down electrons are the same. Note that a Hubbard model is an approximation with only the most diagonally dominant terms of the Coulomb operator,  $V_{ijkl} = U \delta_{ij} \delta_{jk} \delta_{kl}$  for Hubbard interaction  $U$  [177]. We have restricted our consideration to the Born-Oppenheimer approximation [178], even though the discussion can be generalized to the motion of nuclei [96].

Solving the entire many-body problem is known to be difficult if not impossible. However, approximate methods can yield results that are accurate to what is known as chemical accuracy (1 mHa) or a stricter limit applies in some cases [145].

### 2. Basis sets

We can note that Eq. (A1) has been written in the second quantized form since we expect to need a basis to truncate the problem to a more manageable size. One may, for example, choose Gaussian orbitals [179] so that Eqs. (A2) and (A3) can be evaluated analytically and chemical accuracy can be obtained with only a few functions. Other basis functions can also be chosen [145].

It is known that Eq. (A3), when represented in a local basis, reduces to [145]

$$\lim_{\text{loc.}} V_{ijkl} = \frac{1}{2} \iint |\gamma_{ik}(\mathbf{r}, \mathbf{r}')|^2 v_{ee}(\mathbf{r} - \mathbf{r}') d\mathbf{r} d\mathbf{r}' \sim O(N^2) \quad (\text{A4})$$

for a density matrix  $\gamma$  where the limit is taken for well separated, local orbitals at large distances. This reduces the computational complexity from  $O(N^4)$  to  $O(N^2)$  in the asymptotic limit, although the true scaling lies somewhere in-between depending on the details of the system [24]. This argument applies only to orbitals that drop off sufficiently quickly with distance from the origin.

#### a. The curse of dimensionality and other limitations in pointlike basis sets

We note that the reduction in Eq. (A4) happens immediately when using purely local basis sets, with no spatial extent, is used. In that case, we would reduce to a sum over only the diagonal elements of the two-electron integral,  $V_{ijkl}$ , if the orbitals were pointlike. However, using only these localized orbitals (e.g., grid points, plane-waves, wavelets, etc.) comes at a steep price.

In particular, note that wavelets are very expensive for large-scale quantum chemistry problems. It has been known for some time now that a curse of dimensionality shows that the number of functions in one dimension scales as  $N_{\text{1D}}^d$  for  $d$  dimensions with a number of functions in one dimension,  $N_{\text{1D}}$  [153]. Due to the large number of basis functions, wavelet based functions have only been able to solve two- and three-electron systems maximum in the general case [180], although these functions can be efficient for larger

noninteracting or single Slater-determinant theories or other cases of very particular interest [181]. Wavelets are simply not expected to be efficient for real three-dimensional systems of any meaningful size based on preexisting works unless the problem is converted to a noninteracting theory or a special geometry is chosen. For more information, see the references in Ref. [152].

For plane-wave functions, many thousands of functions are required to resolve the electron-electron cusp (e.g., the behavior of the wavefunction at the nucleus in a hydrogen  $1s$  orbital) [145,149–151]. We will not consider pointlike basis functions further here to concentrate on the general case, although plane waves can be useful for periodic systems. With respect to the density matrix (which is a highly important quantity in Sec. B), the full double sum will be taken [see Eq. (B3)] and not just diagonal elements.

In summary, even though the scaling of the two-electron operator is better for pointlike basis sets, many more basis functions will be required to obtain accurate results except in special cases. So, choosing a pointlike basis function will not represent a general strategy for all types of quantum chemical problems that we may wish to solve.

## APPENDIX B: DENSITY FUNCTIONAL THEORY

The foundations of density functional theory (DFT), including the Kohn-Sham system and other variants, are introduced here.

A compact representation of the quantum ground state is the one-body density. In DFT, the ground-state wavefunction is replaced with the density. It was proven in Ref. [29] that the one-body density, defined as

$$n(\mathbf{r}) = \int \dots \int |\Psi(\mathbf{r}, \mathbf{r}_2, \dots, \mathbf{r}_{N_e})|^2 d\mathbf{r}_2 \dots d\mathbf{r}_{N_e} \quad (\text{B1})$$

is sufficient to characterize the ground state. Note that in order to obtain this quantity on the lattice, the one-body reduced density matrix must be obtained for  $N_e$  electrons,

$$\hat{\rho}(\mathbf{r}, \mathbf{r}') = \int \dots \int \Psi^*(\mathbf{r}, \mathbf{r}_2, \dots, \mathbf{r}_{N_e}) \times \Psi(\mathbf{r}', \mathbf{r}_2, \dots, \mathbf{r}_{N_e}) d\mathbf{r}_2 \dots d\mathbf{r}_{N_e} \quad (\text{B2})$$

and is related to the density in the limit where  $\mathbf{r} \rightarrow \mathbf{r}'$

$$n(\mathbf{r}) = \frac{1}{2} \sum_{ij} \varphi_i^*(\mathbf{r}) \rho_{ij} \varphi_j(\mathbf{r}), \quad (\text{B3})$$

where  $\rho_{ij} = \langle \Psi | \hat{c}_i^\dagger \hat{c}_j | \Psi \rangle$ . A spin index has been suppressed, signifying a spin degenerate ground state. However, extensions to ground states without spin degeneracy are also available [51].

Having replaced the wavefunction for the more compact density, the Hamiltonian must be substituted for another mathematical object that acts on the density. In general, an object that maps a function to a scalar value is known as a functional [182]. In DFT, a functional maps the one-body density to a scalar energy value.

In order to find the ground-state energy, we can use a minimization over all densities [183]

$$E = \min_n \left( F[n] + \int n(\mathbf{r}) v(\mathbf{r}) d\mathbf{r} \right), \quad (\text{B4})$$

although it is impractical to search for the ground-state density with this formulation. The second term in Eq. (B4) is the external potential functional (often denoted as  $V[n]$ ) and has a known form. Contrastingly, the universal functional,  $F[n]$ , is defined as the search over all wavefunctions  $\Psi$  constrained to give the density [50,51],

$$F[n] = \min_{\Psi \rightarrow n} \langle \Psi[n] | \hat{T} + \hat{V}_{ee} | \Psi[n] \rangle \quad (\text{B5})$$

and is common to all systems since it does not depend on the external potential. Clearly, the minimization is not an efficient way to find the functional, but it is useful as a mathematical tool.

Because  $F[n]$  is unknown explicitly (its existence is proven by contradiction), it requires approximation to use [30]. Some limiting cases are known, such as one- or two-electron cases, the uniform gas via a fitting procedure, and one dimension [184–187]. Many exact properties of the functional are known from rigorous mathematical statements [188,189] or limited test cases [93,125]. One common way to design new functionals is to build in exact conditions [190–192].

Note that to solve a problem with  $F[n]$ , the functional derivative can be used and is defined as [50,51]

$$\int \frac{\delta F[g(\mathbf{x})]}{\delta g(\mathbf{x})} \Upsilon(\mathbf{x}) d\mathbf{x} \equiv \lim_{\eta \rightarrow 0} \left\{ \frac{F[g(\mathbf{x}) + \eta \Upsilon(\mathbf{x})] - F[g(\mathbf{x})]}{\eta} \right\} \quad (\text{B6})$$

$$= \left. \left\{ \frac{d}{d\eta} F[g(\mathbf{x}) + \eta \Upsilon(\mathbf{x})] \right\} \right|_{\eta=0}, \quad (\text{B7})$$

where  $\Upsilon$  is an arbitrary test function,  $g$  is the function we wish to evaluate  $F$  around, and  $\eta$  is a small parameter. The first functional derivative is best known from classical physics, where it is used to minimize the Lagrangian via Euler-Lagrange minimization [193]

In order to find the minimal density, a functional derivative can be taken. This is synonymous with the Euler-Lagrange equations in this case [194]

$$\frac{\delta F[n]}{\delta n(\mathbf{r})} + v(\mathbf{r}) = \mu, \quad (\text{B8})$$

where a constant chemical potential  $\mu$  was added as a Lagrange multiplier for the total particle number. This equation is then used to solve for orbital-free DFT.

### 1. Kohn-Sham density functional theory

One useful alternative formulation of  $F[n]$  is KS-DFT. This reformulation of DFT proposes an external potential whose solution resulting one-body density is equivalent to obtaining the one-body density of the fully interacting system. The original goal of DFT was to propose a purely wavefunction-free method to characterize the quantum ground state, but it is difficult to find suitable approximations that are accurate enough.

It can be noted that approximating  $F[n]$  is a large approximation on the total energy. In this alternative formulation, one introduces an easy to solve, noninteracting, auxiliary system to make the required approximation a smaller fraction of the overall energy. Obtaining the KS potential gives insight to many more physical quantities than just the density, and the orbitals of the noninteracting system can be used in a variety of other contexts.

#### a. Finding the Kohn-Sham potential

To formalize the KS system, what is known as the adiabatic connection can be used to transform from the original problem to the final noninteracting problem [195]. Equation (A1) can be rewritten as [196]

$$\mathcal{H}^\lambda = \hat{T} + \lambda \hat{V}_{\text{ee}} + \hat{V}^{(\lambda)}, \quad (\text{B9})$$

where an express dependence on the coupling constant  $\lambda$  has been introduced. The tuning parameter  $\lambda$  can vary between the KS system ( $\lambda = 0$ , where  $\hat{V}^{(\lambda=0)} = \hat{V}_s$ ) and the original system ( $\lambda = 1$ ). Note that the external potential operator ( $\hat{V} = v(\mathbf{r})$ ) has received an implicit coupling constant dependence, but there is no simple analytic form for  $\hat{V}^{(\lambda)}$ . The constraint given in this problem is that the density must be the same for any  $\lambda$ ,

$$n(\mathbf{r}) \equiv n^{(\lambda=1)}(\mathbf{r}) \stackrel{!}{=} n^{(\lambda=0)}(\mathbf{r}), \quad (\text{B10})$$

which is difficult to construct in practice. Note that the other limit of  $\lambda \rightarrow \infty$  can also be used to base a functional theory [197].

#### b. Components of the functional in the Kohn-Sham system

The form of the universal functional for the KS case is

$$F[n] = T_s[n] + U[n] + E_{\text{xc}}[n]. \quad (\text{B11})$$

This form is known from perturbative expansions of the many-body system [30,175]. Note that the subscripted “s” on the kinetic energy is to signify that  $T_s$  is evaluated over noninteracting wavefunctions,  $\phi$ , but has the same form as the same kinetic energy operator in Eq. (A2). This term shows that the KS scheme is not a pure density functional but one that relies the auxiliary noninteracting orbitals. The cost to solve the noninteracting system is larger than pure-DFT, but still smaller than many other approximations.

In addition to the kinetic energy, another known energy in Eq. (B11) is the Hartree energy,

$$U[n] = \frac{1}{2} \iint \frac{n(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r} d\mathbf{r}', \quad (\text{B12})$$

which is fully nonlocal.

The unknown term in Eq. (B11) is the exchange-correlation energy,  $E_{\text{xc}}$ , which is not known as a density functional and requires approximation in practice. If the exact  $E_{\text{xc}}$  is used, then the theory is exact. The usefulness of defining the KS system is that the approximation to the total energy is small for many systems of practical interest.

#### c. Kohn-Sham potential by functional derivatives

The KS potential is explicitly

$$\begin{aligned} v_s(\mathbf{r}) &= \frac{\delta}{\delta n} \left\{ U[n] + E_{\text{xc}}[n] + \int n(\mathbf{r})v(\mathbf{r}) d\mathbf{r} \right\} \\ &= v_{\text{H}}[n](\mathbf{r}) + v_{\text{xc}}[n](\mathbf{r}) + v(\mathbf{r}), \end{aligned} \quad (\text{B13})$$

where a functional derivative is taken over the relevant energy terms, for example,

$$v_{\text{H}}[n](\mathbf{r}) = \frac{\delta U[n]}{\delta n} = \int \frac{n(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r}' \quad (\text{B14})$$

for the Hartree potential, and the form of  $v_{\text{xc}}[n](\mathbf{r})$  is not known explicitly. In summary, by regrouping the nonkinetic energy terms in the Hamiltonian,  $U[n] + V[n] + E_{\text{xc}}[n]$ , the resulting system will appear as noninteracting. The electron-electron term is contained in the resulting potential of the noninteracting system.

#### d. Variational principle for the Kohn-Sham potential

The Kohn-Sham potential also satisfies the minimization of the quantity [73]

$$T_\Psi[v_s] = \langle \Psi[v] | \hat{T} + \hat{V}_s | \Psi[v] \rangle - \langle \Phi[v_s] | \hat{T} + \hat{V}_s | \Phi[v_s] \rangle, \quad (\text{B15})$$

where we follow Refs. [73–75] closely. Note that  $\Psi(\mathbf{r}, \mathbf{r}_2, \dots, \mathbf{r}_{N_e})$  is not an eigenstate of  $\hat{T} + \hat{V}_s$  but that  $\Phi(\mathbf{r})$  is. So,

$$\langle \Psi[v] | \hat{T} + \hat{V}_s | \Psi[v] \rangle > \langle \Phi[v_s] | \hat{T} + \hat{V}_s | \Phi[v_s] \rangle \quad (\text{B16})$$

by the variational principle [198]. The functional derivative of Eq. (B15) with respect to  $v_s(\mathbf{r})$  is [73]

$$\frac{\delta T_\Psi[v_s]}{\delta v_s} = n_\Psi(\mathbf{r}) - n_\Phi(\mathbf{r}) \quad (\text{B17})$$

and equals the difference in the densities of the two systems, one computed from  $\Psi$  ( $n_\Psi$ ) and the other density from  $\Phi$  ( $n_\Phi$ ) [73]. When this difference is zero, the condition for the Kohn-Sham potential is found given in Eq. (B10).

#### e. v-representability

The KS scheme is exactly defined provided that  $v$ -representability is satisfied. In common practice, this is not a concern since it was proven on a grid that the system must be  $v$ -representable since the kinetic energy is regularized [70,71]. So, we always expect  $v$ -representability here.

#### f. Minimization of the Kohn-Sham functional

Note that the Euler-Lagrange minimization of the functional yields the KS equations

$$\left[ -\frac{\nabla^2}{2} + v_s(\mathbf{r}) \right] \phi_j(\mathbf{r}) = \epsilon_j \phi_j(\mathbf{r}) \quad (\text{B18})$$

for some KS energy eigenvalues  $\epsilon_j$  and KS orbitals  $\phi_j(\mathbf{r})$ . The density is then the sum over occupied orbitals equivalent to

$$n(\mathbf{r}) = \sum_{j \in \text{occ.}} |\phi_j(\mathbf{r})|^2, \quad (\text{B19})$$

which can be found from Eq. (B3) by noting that the excitations are orthogonal. One recovers Eq. (B3) with an additional index for the excitations when  $\phi$  is decomposed into a chosen basis.

**g. Relationship between the energies of the Kohn-Sham and the fully interacting system**

The adiabatic connection from Sec. B 1 a does not conserve energy. The relation between the ground-state energy of the interacting system,  $E$ , and the energy of the KS system (the sum of eigenvalues of the noninteracting system,  $\sum_{j \in \text{occ.}} \epsilon_j$ ) is [50]

$$E = \sum_{j \in \text{occ.}} \epsilon_j - U[n] + E_{xc}[n] - \int n(\mathbf{r})v_{xc}(\mathbf{r}) d\mathbf{r} \quad (\text{B20})$$

for Hartree energy  $U$ , exchange correlation energy  $E_{xc}$ , and exchange-correlation potential  $v_{xc}$ . Note that Eq. (B20) shows it is not sufficient to have only the KS potential to find  $E$ , although perturbation theory on the density can be used [199].

**2. Potential functional theory**

When examining Eq. (B4), it is natural to ask if a dual theory can be formulated based on  $v(\mathbf{r})$  instead of  $n(\mathbf{r})$  since both are one-body quantities. This question stems from noticing that functional derivatives of  $n(\mathbf{r})$  yield equations that can be solved for the density, resulting in the Euler-Lagrange minimization for the density functional from Eq. (B8).

To the question: Can we instead take a functional derivative with respect to  $v(\mathbf{r})$  instead? The answer is yes. It was proven in Ref. [66] that the dependence on the functional in terms of the external potential was sufficient to describe the ground state. In this theory, the density must be determined from  $v(\mathbf{r})$  directly as  $n(\mathbf{r}) \rightarrow n[v](\mathbf{r})$ . The resulting energy becomes

$$E = \min_{n[v]} \left( F[v] + \int n[v](\mathbf{r})v(\mathbf{r}) d\mathbf{r} \right), \quad (\text{B21})$$

where

$$F[v] = \min_{\Psi \rightarrow n[v]} \langle \Psi[v] | \hat{T} + \hat{V}_{ee} | \Psi[v] \rangle, \quad (\text{B22})$$

which is similar to Eq. (B5).

**a. Why the functional derivative is also necessary**

Very importantly, one cannot determine the entire character of the ground state [i.e., find  $n(\mathbf{r})$  or equivalent] with only  $E[v]$ . To see this, note that the Euler-Lagrange equation for potential functional theory is [112]

$$\frac{\delta F[v]}{\delta v} + \int \frac{\delta n[v](\mathbf{r})}{\delta v(\mathbf{r}')} v(\mathbf{r}') d\mathbf{r}' = 0, \quad (\text{B23})$$

where a derivative of the density with respect to the external potential in the second term of the left-hand side must be determined to solve this equation and find the density. In summary, one can formulate potential functionals,  $E[v]$ , provided that  $n[v](\mathbf{r})$  is known [86,111]. In other words, the functional derivative is also necessary to perform self-consistent calculations if only the energy is known for a given potential.

**3. Time-dependent density functional theory**

In a time-dependent DFT (TD-DFT), one may simply propagate the KS potential according to Schrödinger's equation for time evolution [57,194]

$$i \frac{\partial}{\partial t} \phi_j(\mathbf{r}, t) = \left\{ -\frac{\nabla^2}{2} + v_s[n, \Phi_0](\mathbf{r}, t) \right\} \phi_j(\mathbf{r}, t) \quad (\text{B24})$$

with an initial starting state  $\Phi_0$ . A formal justification for the existence of TD-DFT is available [97,98].

Computing response functions is also necessary if a perturbation to  $v_s(\mathbf{r})$  is applied. Knowing just the KS orbitals is sufficient to determine the response function for the KS system [57],

$$\chi_s(\mathbf{r}, \mathbf{r}', \omega) = \lim_{\eta \rightarrow 0^+} \sum_{k,j=1}^{\infty} (\xi_k - \xi_j) \frac{\phi_k^*(\mathbf{r})\phi_j(\mathbf{r})\phi_j^*(\mathbf{r}')\phi_k(\mathbf{r}')}{\omega - (\epsilon_j - \epsilon_k) + i\eta} \quad (\text{B25})$$

with occupation numbers  $\xi_j$ , eigenvalues  $\epsilon_j$ , frequency  $\omega$ , and small parameter  $\eta$ . Hence, knowing all eigenvalues of the  $v_s(\mathbf{r})$  at  $t = 0$  gives the KS response function. One can also relate  $\chi_s$  to the interacting response function  $\chi$  via a kernel ( $n_{g.s.}$  is the ground-state density)

$$f_{xc}[n](\mathbf{r}, t, \mathbf{r}', t') = \left. \frac{\delta v_{xc}[n](\mathbf{r}, t)}{\delta n(\mathbf{r}', t')} \right|_{n=n_{g.s.}} \quad (\text{B26})$$

and the relation

$$f_{xc}(\mathbf{r}, \mathbf{r}', \omega) = \chi_s^{-1}(\mathbf{r}, \mathbf{r}', \omega) - \chi^{-1}(\mathbf{r}, \mathbf{r}', \omega) - v_{ec}(\mathbf{r} - \mathbf{r}'), \quad (\text{B27})$$

which is similar to a Dyson's equation. Many cases of interest obtain sufficiently accurate answers with only the adiabatic approximation, however. TD-DFT can be used to find excited states [99].

**4. Density functional theory at finite temperature**

In order to incorporate finite temperature effects into the density functional, an entropy term can be added following the original treatment by Mermin [100], one can write the grand canonical free energy as

$$\hat{\Omega} = \mathcal{H} - \tau \hat{S} - \mu \hat{N} \quad (\text{B28})$$

for temperature  $\tau$ , chemical potential  $\mu$ , number operator  $\hat{N}$ , and entropy operator

$$\hat{S} = -k_B \ln \hat{\Gamma}, \quad (\text{B29})$$

where [200]

$$\hat{\Gamma} = \sum_{N_e, i} p_{N_e, i} |\psi_{N_e, i}\rangle \langle \psi_{N_e, i}| \quad (\text{B30})$$

with  $\sum_{\{N_e, i\}} p_{N_e, i} = 1, 0 \leq p_{N_e, i} \leq 1$ , and states  $\psi_{N_e, i}$  indexing excitations over a particular number of particles  $N_e$ .

The minimum of  $\hat{\Omega}$  is (and adding descriptive indices)

$$\hat{\Omega}_{v, \mu}^{\tau} = \min_n \left\{ F^{\tau}[n] + \int n(\mathbf{r})(v(\mathbf{r}) - \mu) d\mathbf{r} \right\} \quad (\text{B31})$$

and

$$F^{\tau}[n] \equiv \min_{\hat{\Gamma} \rightarrow n} \{ T[\hat{\Gamma}] + V_{ec}[\hat{\Gamma}] - \tau S[\hat{\Gamma}] \}. \quad (\text{B32})$$

In summary, if a system is solved at a given temperature, one can solve for the KS potential analogously to the ground state with an extra term  $-\tau\hat{S}$  (and a term for the particle number) representing the entropy in the functional. Note that the ground-state density is replaced by the  $\hat{\Gamma}$  object which is akin to a density matrix and that extra weights must be solved. In order to find this, several states  $\psi_{N_e,i}$  must be used.

### 5. Comment on density functional approximations

There are many functionals that can be used to approximate  $E_{xc}$ . Each performs with its own set of systematic deficiencies.

The most pertinent approximations for this paper are the ML functionals [31,33–38,41–45]. The general strategy of fitting a functional may be unpalatable [83], but the generic strategy of fitting exact data is not unique to ML functionals. The simplest approximation to the functional—known as the local density approximation—is a fit of highly accurate quantum Monte Carlo data [201,202]. Further, coefficients present in hybrid functionals are also fit to existing data [203], among other examples. The ML functionals simply represent a more robust approximation that can interpolate well provided the system solved is close to the training manifold. This strategy would capture the exact conditions of the exact functional [33,49–51,204].

### APPENDIX C: TRAINING A MACHINE LEARNING MODEL WITH STOCHASTIC GRADIENT DESCENT

Machine learning methods rely on the minimization of some cost function. Here we describe the most basic version of this, the stochastic gradient descent (SGD). If a function must be minimized, some procedure for the minimization is necessary. We can define a cost function,  $g$ , that could take the form

$$g = \sum_{\iota, \mathbf{x}} [n_{\hat{\mathbf{w}}}(\mathbf{x}_{(\iota)}) - n_{(\iota)}]^2 \quad (C1)$$

for some observable  $n$  with known values of  $n_{(\iota)}$  (called a training set) indexed by  $\iota$  with some coefficients for the weight  $\hat{\mathbf{w}}$  and bias  $\mathbf{b}$  in the form

$$\mathbf{x}^{(i+1)} = \mathcal{S}(\hat{\mathbf{w}}^{(i)} \cdot \mathbf{x}^{(i)} + \mathbf{b}^{(i)}) \quad (C2)$$

for a level  $i$  of the neural network with a nonlinear function  $\mathcal{S}$  with a given input  $\mathbf{x}_{(\iota)}$ . The final level of the neural network will be the final quantity of interest,  $n_{\hat{\mathbf{w}}}(\mathbf{x}_{(\iota)})$ .

In order to minimize  $g$  and therefore construct the best approximation to the known values, a gradient descent can be performed. The basic idea is to ensure that any evolution of the coefficients  $\hat{\mathbf{w}}$  occur along the steepest negative gradient in the system. In order to ensure that the gradient is negative, we can start from a consequence of Gauss’s law which states that a gradient of a scalar (here,  $g$ ) along the direction of changing  $\hat{\mathbf{w}}$  (denoted  $\delta\hat{\mathbf{w}}$ ) is equivalent to the Laplacian of  $g$ ,  $\Delta g = \nabla_{\hat{\mathbf{w}}} g \cdot \delta\hat{\mathbf{w}}$  where  $\nabla_{\hat{\mathbf{w}}} g = (\partial_{w_1} g, \partial_{w_2} g, \dots)$ .

If the form of the update from iteration over a time  $\delta t$  (which can also be expressed in the discrete case as earlier) is chosen as

$$\hat{\mathbf{w}}_{t+\delta t} = \hat{\mathbf{w}}_t - \eta \nabla_{\hat{\mathbf{w}}} g \quad (C3)$$

for some small parameter  $\eta$  (called a learning rate), then applying  $\nabla_{\hat{\mathbf{w}}} g$  to each side with a dot product gives  $\Delta g = -\eta |\nabla_{\hat{\mathbf{w}}} g|^2 < 0$ . In other words, the Laplacian of  $g$  is guaranteed to be negative for a small enough  $\eta$  such that  $g$  is linear on a small neighborhood. The argument was shown for the weights of the neural network, but the same argument applies to the biases.

Evaluating Eq. (C1) for all  $\iota$  provided can be very costly since the resulting gradient is very noisy. To speedup the gradient descent, a randomly sampled subset of the provided training data, called a mini-batch. This can be one or more selected points. Since the points are randomly selected at each step, the gradient descent is taken stochastically.

Even though the cost function evaluated over the mini-batch must be negative, the entire cost function evaluated over all training points does not need to decrease. However, on average, the observable’s value is lowered over several SGD steps. The condition is  $\langle \partial_t g \rangle < 0$  for a mini-batch in the general case but would be  $\partial_t g < 0$  if all points are used.

There are two steps required when training a ML model with SGD: forward and backward propagation. In forward propagation, Eq. (C2) is applied straightforwardly from the input to the output layers of the neural network. The backward propagation requires that a derivative of Eq. (C2) from the output to input layers. Repeating this constructs the cost function and then applies the gradient to all weights and biases in the network until the model is more converged.

More advanced algorithms can also be used to converge gradient descents as well [205].

### 1. Gradient-free methods

This section has been focused on training ML models with gradients. Another class of method, one that uses random walks [206–215], is also available. However, these methods of training that involve a random walk typically only perform well on small numbers of parameters. In fact, these methods perform better than gradients in some cases for these small systems. If the problem is too large, then a gradient-based method is generally better [211]. There may also be opportunities to combine the two [216,217], although this may require that the wavefunction is re-solved to implement on the quantum computer. We were not able to find any evidence that random walks can reliably compete with gradient-based training methods; however, if one could use such an algorithm, a quadratic speedup is available on the quantum computer for training with the random walk [162].

We also note that kernel based methods can be used to train the neural network but that they are not as “choice-free” since a functional form of the kernel must be selected. The minimization of the coefficients with a kernel does not require gradients [48].

### APPENDIX D: QUANTUM ALGORITHMS

On a quantum computer, both the method of manipulating and storing information is different from a classical computer. On a classical computer, electrons are moved around and represent different information based on where they are placed.

In a quantum computer, the quantity that we manipulate is the spin of an electron or some other quantity that is allowed to exist in a superposition of states [1]. We will refer to qubits in this work but note that one can extend the ideas to qudits where more than two states are possible. It is not possible to determine all coefficients of a wavefunction in a superposition without an exponential number of measurements to find them (i.e., measuring the spin for the both  $|0\rangle$  and  $|1\rangle$  states for each qubit individually). To manipulate the state of a quantum wavefunction, we can apply operators, specifically operators that are unitary. Often, the operators are applied to a specified qubit and also another auxiliary qubit to keep required operations unitary.

Sequences of these unitary operators can be cast as a tensor network diagram. Each line of the diagram represents a single qubit or a group of qubits called a register. Each block is a unitary operation that manipulates the state of one or more qubits. Note that the allowed operations on the quantum computer maintain the number of lines and do not involve truncations of the space at any point. This is qualitatively different from other uses of tensor network diagrams used for “tensor network methods” which can refer to a class of algorithms that solve for the ground state of a quantum system on classical computer [91]. So, there is a subtle distinction between the tensor network methods and the tensor network diagrams we draw here even though the general properties of the diagrams is the same in both. The main difference here is that no form of truncation or a connection with a renormalization group is taken explicitly.

One example of a useful gate that will appear in several places is the Hadamard gate,

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \tag{D1}$$

which is written in the  $\{|0\rangle, |1\rangle\}$  qubit basis states. Applying this gate on a  $|0\rangle$  state will give an equal superposition over the  $|0\rangle$  and  $|1\rangle$  states,

$$H|0\rangle = (|0\rangle + |1\rangle)/\sqrt{2}, \tag{D2}$$

which can be applied identically to more qubits as denoted by the  $\otimes$  operator.

We present some common algorithms in the context of solving quantum chemistry problems [218]. The quantum gradient algorithm (QGA) is used to find derivatives of a function, generically. Meanwhile, the quantum phase estimation (QPE) determines the phase of a given state. An important sub-algorithm in the QGA and QPE is the quantum Fourier transform (QFT) which is detailed first [1]. Real-time evolution (RTE) in addition to quantum amplitude estimation (QAE) are also discussed.

Note that everywhere we use the symbol  $N$  for the number of qubits in this section. Many times in the literature, the number of qubits may also implicitly mean  $N$  registers with  $r$  qubits each for  $r$  digits of precision. If this is the case, the same concepts would apply, regardless.

### 1. Quantum Fourier transform

The QFT begins with a set of input data recorded on an initial set of registers, which we denote as  $|y\rangle$ . The end result

of the QFT is to change the data  $|y\rangle$  into  $|x\rangle$  according to the discrete Fourier transform [219]

$$|x\rangle = \frac{1}{2^{N/2}} \sum_{y=0}^{2^N-1} \exp(i2\pi xy/2^N) |y\rangle \tag{D3}$$

for  $N$  qubits with a normalization factor  $2^{-N/2}$  coming from the prefactor in Eq. (D1). The key first step is to understand how Eq. (D3) can be reexpressed in binary instead of integers integer  $y$ . A number  $y$  can be reexpressed in base 2 with the digits (assuming value either 0 or 1, corresponding to the states of a qubit)  $y_0 y_1 \dots y_{N-1}$  where  $N$  is the maximum number of (qu)bits we allow for the binary number. In full,  $y$  relates to the binary digits as

$$y = 2^{N-1}y_0 + 2^{N-2}y_1 + \dots + 2y_{N-2} + y_{N-1}. \tag{D4}$$

A similar form can be written for using a qudit where a different basis is used (e.g., trinary).

To express the binary representation of  $|y\rangle$  as qubits, let us express the state as

$$|y\rangle = |y_0 \dots y_{N-1}\rangle = |y_0\rangle \otimes \dots \otimes |y_{N-1}\rangle \equiv \bigotimes_{\ell=0}^{N-1} |y_\ell\rangle \tag{D5}$$

with each of the subindices representing another digit of  $y$ 's binary representation ( $y_\ell \in \{0, 1\}$ ).

We can substitute  $y$  for its binary representation as using the previously defined expressions in Eq. (D4) to find [1]

$$|x\rangle = \frac{1}{2^{N/2}} \bigotimes_{\ell=0}^{N-1} \sum_{y_\ell=0}^1 \exp(i\pi x y_\ell / 2^\ell) |y_\ell\rangle, \tag{D6}$$

where the sums in Eq. (D3) are rewritten for the binary representation as

$$\sum_{y=0}^{2^N-1} \equiv \bigotimes_{\ell=0}^{N-1} \sum_{y_\ell=0}^1 = \sum_{y_0=0}^1 \sum_{y_1=0}^1 \dots \sum_{y_{N-1}=0}^1 \tag{D7}$$

for simplicity. Noticing that each term is factorizable, this becomes

$$|x\rangle = \frac{1}{2^{N/2}} \bigotimes_{\ell=0}^{N-1} \left\{ |0\rangle + \exp(i\pi x / 2^\ell) |1\rangle \right\}, \tag{D8}$$

where for a given  $\ell$ , the division by a power of 2 in the argument of the exponential will reveal one digit of the binary representation of the integer  $x$  [1].

The operators necessary to perform a QFT on a quantum computer can be identified in Eq. (D8). Note that applying the Hadamard gate on a qubit written in an arbitrary binary form is

$$H|x_k\rangle = [|0\rangle + \exp(i\pi x_k) |1\rangle] / \sqrt{2} \tag{D9}$$

as can be verified by noting that  $x_k$  is either 0 or 1. So, a Hadamard will rotate each bit of  $x$  into a basis that is only different from an individual term in Eq. (D8) by a phase on the  $|1\rangle$  state. If we apply the phase rotation gate of the form

$$R_\ell = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/2^\ell} \end{pmatrix} \tag{D10}$$

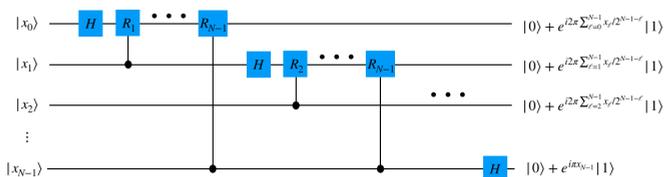


FIG. 2. Circuit diagram of the quantum Fourier transform. Note that the rotation operator is about the  $z$  axis, and the Hadamard is about the  $x$  axis. So, they cannot commute nor can they be placed in a different order.

after the Hadamard, then this will constitute the most basic operation in the QFT. That is, applying  $H$  and then  $R_\ell$  a certain number of times will generate terms required in Eq. (D8). More than one  $R_\ell$  gate may need to be applied depending on which bit of  $x$  we are acting on. The structure of gates is shown in Fig. 2. The next qubit has all but the first register’s rotation matrix applied. We then continue to the next qubit, applying one less set of gates, and continue until we apply only the Hadamard on the last qubit. If a swap operation is applied, the qubits will appear in order, and this completes the QFT.

Note that the overall cost of this algorithm scales as  $O(N^2)$  but that a cheaper  $O(N \log N)$  is also available [220]. Note also that since we cannot efficiently obtain all coefficients from the superposition, the algorithm does not provide a useful speedup over the classical algorithm which scales exponentially, not polynomially. However, the time to measure all elements is exponentially long, so this quantum advantage is not truly advantageous. Still, the QFT can be useful as a tool in other subroutines.

### 2. Quantum phase estimation

Given an input state,  $|\psi\rangle$ , we want to determine the associated eigenvalue of the form  $\exp(i2\pi\varphi)$  for some real  $\varphi$  denoting a phase. For this algorithm, we must first have the initial state,  $|\psi\rangle$ , and a number of auxiliary qubits at least equal to the number of digits that the phase must be accurate to.

The strategy will be to generate, from an initial wavefunction  $\psi$ , the binary digits of the phase of  $\varphi$ —as defined in Appendix D 1—and then perform an inverse QFT to obtain the phase on an auxiliary register.

Let a gate  $\mathcal{U}$  [=  $R_1$  from Eq. (D10)] be one such that when applied to  $\psi$ , and controlled on one of the auxiliary qubits, it produces a phase that is the  $j$ th value of the binary representation,

$$\mathcal{U}^{2^j} H|0\rangle_j |\psi\rangle = \frac{1}{\sqrt{2}} (|0\rangle + e^{i2\pi 2^{-j}\varphi} |1\rangle)_j |\psi\rangle, \quad (\text{D11})$$

where  $|0\rangle_j$  is the  $j$ th qubit and  $t$  is the total number of qubits that this operator will be applied to.

The gate  $\mathcal{U}$  is applied as in Eq. (D11) to a register of auxiliary qubits and to  $\psi$  as in Fig. 3. The resulting state is then

$$\frac{1}{2^{t/2}} \bigotimes_{j=1}^t (|0\rangle + e^{i2\pi 2^{-j}\varphi} |1\rangle) = \frac{1}{2^{t/2}} \sum_{y=0}^{2^t-1} e^{i2\pi\varphi y} |y\rangle, \quad (\text{D12})$$

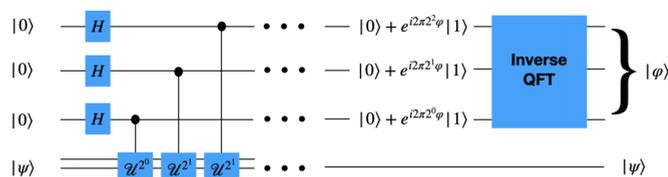


FIG. 3. Circuit diagram of the quantum phase estimation algorithm. The input is an auxiliary register and the initial wavefunction. The output is the phase corresponding to the eigenvalue and the original wavefunction. Essentially, one obtains the associated energy for an input wavefunction.

where the same conversion to and from a binary representation in Appendix D 1 is used. Once an inverse Fourier transform is applied on the last step, we obtain  $|\varphi\rangle$  and have then represented the phase on a set of qubits.

In order to construct the unitary operator  $\mathcal{U}$ , we simply must obtain the representation of the exponentiated Hamiltonian as  $\exp(-i\mathcal{H}t)$  [11]. The energy of the wavefunction ( $\varphi$  related to  $E$ ) is then related to the time applied and may involve other predetermined constants [11]

In order to apply the exponentiated Hamiltonian, one option is to use the Trotter-Suzuki decomposition [221] to decompose the exponential into a product of exponentials with fewer terms [24].

By expanding the number of qubits used in the auxiliary register, we can increase the accuracy of the final result. We will defer detailed analysis of this point to Ref. [1], but it is worth noting that some error in this algorithm can be reduced with more resources. The total error can be reduced arbitrarily to  $1 - \eta$  for some small number  $\eta$ .

Note that improvements can be applied to the algorithm to generate more methods of QPE [222]. A recent improvement known as qubitization can bring down the gate count for the determining the phase. We will defer to the discussion in Ref. [65].

### 3. Quantum gradient algorithm

Given an oracle for a function  $f$ , its gradients can be computed in one query of the oracle instead of  $m + 1$  classically for  $m$  grid points [81].

We start with three multiqubit registers. One is used for the computation of  $f$ . The other contains an equal superposition over all states (representing the infinitesimal directions that  $f$  can be shifted for an eventual gradient). The last receives a QFT (on initial register set to 1 while the others are set to zero). This final register will be used for a phase kickback. The purpose of the phase kickback is to modify the phases of the QFT. When the inverse QFT is applied, we obtain the gradient similar to how the phase was obtained for the QPE.

The steps of this algorithm are shown in Fig. 4. The Hadamard gate produces

$$H^{\otimes N} |0\rangle^{\otimes N} = \frac{1}{\sqrt{2^N}} \bigotimes_{\ell=1}^N \sum_{\delta_\ell=0}^1 |\delta_\ell\rangle \equiv \frac{1}{\sqrt{2^N}} \sum_{\delta} |\delta\rangle, \quad (\text{D13})$$

where  $N$  is the number of qubits contained in the second register.

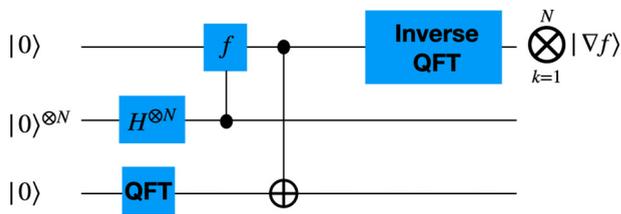


FIG. 4. Circuit diagram of the quantum gradient algorithm. One could write the function  $f$  with a control to a fourth register with input  $c$  for the point that  $f$  is evaluated on.

When calling the oracle query as controlled on the equal superposition in Eq. (D13),  $f$  is evaluated on all arguments  $\delta_\ell$  giving  $f(c + \delta)$ , having perturbed an initial coordinate by  $\delta$  or some similar function [82].

On the third register in Fig. 4, a QFT has been applied on an initial value of 1, giving an output of

$$\frac{1}{2^{N_p/2}} \sum_w e^{i2\pi w/2^{N_p}} |w\rangle, \quad (\text{D14})$$

which is similar to Eq. (D3). The number of qubits in this register,  $N_p$ , are enough to allow a pending bitwise addition to be carried out properly.

The state of the full quantum wavefunction is then

$$|\psi\rangle = \frac{1}{\sqrt{2^N 2^{N_p}}} \sum_\delta \sum_w e^{i2\pi w/2^{N_p}} |w\rangle |f(c + \tilde{\delta})\rangle |\delta\rangle, \quad (\text{D15})$$

where

$$\tilde{\delta} = L(\delta - \mathbf{N}/2)/2^N \quad (\text{D16})$$

and  $N$  is a vector of  $(2^N, 2^N, \dots)$  and provides the offset factor to convert integers to real numbers.  $M$  and  $L$  are assigned meaning in the following. The factor  $M$  scales the maximum amount of  $\nabla f$  to keep this quantity expressed as an integer. The parameter  $L$  is a neighborhood over which the derivative is accurate to first order, for example in one dimension

$$\partial_x f \approx \frac{f(x + L/2) - f(x - L/2)}{L} \quad (\text{D17})$$

is the decomposition in one dimension.

The next step is to add (bitwise addition denoted by  $\oplus$ ) the first register to the third under the addition  $w \rightarrow w \oplus (2^N 2^{N_p} f)/(ML) \bmod 2^{N_p}$  so that the register and phase are shifted as

$$|\psi\rangle = \frac{1}{\sqrt{2^N 2^{N_p}}} \sum_\delta \sum_w e^{i2\pi [w + \frac{2^N 2^{N_p}}{ML} f(c + \tilde{\delta})]/2^{N_p}} \times |w\rangle |f(c + \tilde{\delta})\rangle |\delta\rangle, \quad (\text{D18})$$

where this trick is often called a phase kickback. In a small neighborhood,  $L$ , around the central point  $c$  of the oracle query the vectors  $\delta$  can be thought of as perturbations on this point. Expanding the function according to a Taylor expansion gives

$$\begin{aligned} & \sum_\delta e^{i2\pi \frac{2^N}{ML} [f(c) + \frac{L}{2^N} (\delta - \frac{\mathbf{N}}{2}) \cdot \nabla f]} |f(c + \tilde{\delta})\rangle |\delta\rangle \\ & = e^{i2\pi \frac{2^N}{ML} f(c)} e^{i\frac{2\pi}{M} \frac{\mathbf{N}}{2} \cdot \nabla f} \sum_\delta e^{i\frac{2\pi}{M} \delta \cdot \nabla f} |f(c + \tilde{\delta})\rangle |\delta\rangle. \end{aligned} \quad (\text{D19})$$

Recall that, by inspection from Eq. (D3), applying the inverse Fourier transform will give

$$\bigotimes_k \left| \frac{2^N}{M} (\nabla_{x_k} f) \Big|_c \right\rangle. \quad (\text{D20})$$

So, given a continuous function  $f$ , we obtain a gradient  $\nabla f$ . The representation on the qubits for both  $f$  and  $\nabla f$  is in the binary representation of their continuous values. Improvements to this algorithm have been noted [82].

#### a. Functional derivatives with quantum gradient algorithms

The QGA can be used to evaluate the functional derivative, Eq. (B6). The test field  $\Upsilon(\mathbf{x})$  can be provided by hand or by Hadamard transformation with each resulting state giving the same result. Evaluating the derivative as before on  $\eta$  produces the functional derivative. This could be applied to a variety of functionals, such as the density functional or the partition function (although this last object would be very difficult to compute before taking the functional derivative due again to the curse of dimensionality) [223]. There are other ways to get the functional derivative, such as with a chain rule if an alternative form is required.

#### 4. Real-time evolution

While it is possible in theory to implement a more advanced classical algorithm on the quantum computer, there may be sizable overhead. It is generally accepted in the literature to use the real-time evolution (RTE) method which we choose to introduce here [224].

The initial state of qubits can be initialized into a state consisting of a single-particle Hamiltonian's,  $\mathcal{H}_0$ , eigenstate. This could be the Hartree-Fock solution for some number of electrons,  $N_e$  [11]. The Hamiltonian is then given a time dependence such that  $t = 0$  is  $\mathcal{H}_0$  and  $t = t_{\max}$  is the full Hamiltonian,

$$\mathcal{H}(t) = \mathcal{H}_0 + \lambda(t)\mathcal{H}_1 + \mathcal{C} \quad (\text{D21})$$

for some time-dependent function  $\lambda(t)$  and interaction term  $\mathcal{H}_1$ . By tuning the time parameter slowly enough, the new ground state can be found. A constant  $\mathcal{C}$  is added in anticipation of the QPE and is simply taken into account when converting the output phase to the energy [224].

The final state of the RTE must be a close approximation to the true ground state for QPE to work properly [1,225]. In order to evolve the initial state to the ground state, the error in the Trotter step must be no more than the allowed accuracy for a computation [23]. For the case of molecular systems, this is 1 mHa (although this can be even lower for some applications). A variable number of steps is required to fully evolve the initial wavefunction to the ground state, but this may be on the order of a number of thousands and the entire process can take months or much, much longer [24].

There are also other algorithms that could be used [138,155–159], but these may have a large overhead. Just as with phase estimation, we present the most widely known algorithm here for ease of presentation. The RWMP method of the main text can interchange subroutines for the best algorithm.

### 5. Quantum amplitude estimation (quantum counting)

A way to obtain useful quantities from a wavefunction without measuring it is to use QAE [226] as described by Ref. [5] (although we follow the state-preserving quantum counting algorithm used in Ref. [58] and also note Ref. [59,227]). One can envision the application of an operator  $\hat{O}$  onto the wavefunction (e.g.,  $\hat{c}_i^\dagger \hat{c}_j$  onto  $|\Psi\rangle$ ) as being represented in a superposition of the original function  $\Psi$  and all other states that are perpendicular,  $\Psi^\perp$ , as

$$\hat{O}|\Psi\rangle = \alpha_0|\Psi\rangle + \alpha^\perp|\Psi^\perp\rangle. \quad (\text{D22})$$

We will simply assume some process exists to apply the operator of interest is available. The goal is to estimate  $\alpha_0$  which is the expectation value. A series of steps is required to find this coefficient. The fraction of times that the following is successful will give  $\alpha_0$ . Before performing any of the subsequent steps, we assume that the energy of  $\Psi$  has been obtained via phase estimation beforehand on a separate register. In total, four registers are required: one for  $\Psi$ , one for the saved ground-state energy, one for the check energy, and one for the pointer qubit.

One iteration of the full algorithm is the following:

- (1) An operator is applied to  $\Psi$
- (2) The energy of the resulting state is determined; the energy is in a superposition over all states
- (3) The newly found energy is compared to the saved energy of the original wavefunction
- (4) The difference is represented as a single bit (called a pointer qubit)

(5) (Accept) The pointer qubit is measured. The algorithm then branches: if the measurement of the single pointer qubit gives a success, implying the energies match and the original  $\Psi$  was recovered, we repeat the steps starting from step 1 here after returning the state to the original configuration. A separate counter is incremented by one each time this step is reached.

(6) (Reject) If the pointer measurement results in a failure, then the wavefunction found is not the original. We must recover the original wavefunction by undoing the QPE, undoing the operator applied ( $\hat{O}^\dagger$ ), applying the operator again, and again finding the energy difference. We return to step 1.

More details and diagrams can be found in Refs. [5,58,228]. Intuitively, we are counting the number of times the starting wavefunction is recovered when applying an operator. The ratio of accepted counts to the total number of times the operator is applied is related to the coefficient on the ground state,  $\alpha_0$ . While the description here involves a measurement and therefore gives classical data, the algorithm can be used as an oracle query for the QGA (as stated in Ref. [58] at the cost of additional auxiliary qubits).

In order to see how the algorithm will converge when the reject step is activated, we can analogize with the half-life of radioactive isotopes to envision when the rejection procedure must eventually recover the correct ground state. A detailed analysis of the convergence of the rejection step shows the number of steps required to find the original  $\Psi$  is related to  $1/\varepsilon$  for some probability of failure,  $\varepsilon$ , and is found in Refs. [5,58].

- 
- [1] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, 2010).
- [2] D. Deutsch and R. Jozsa, Rapid solution of problems by quantum computation, *Proc. R. Soc. Lond A* **439**, 553 (1992).
- [3] P. W. Shor, in *Proceedings of the 35th Annual Symposium on Foundations of Computer Science* (IEEE, Piscataway, 1994).
- [4] L. K. Grover, Quantum Mechanics Helps in Searching for a Needle in a Haystack, *Phys. Rev. Lett.* **79**, 325 (1997).
- [5] G. Brassard, P. Hoyer, M. Mosca, and A. Tapp, Quantum amplitude amplification and estimation, *Contemp. Math.* **305**, 53 (2002).
- [6] P. W. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, *SIAM Rev.* **41**, 303 (1999).
- [7] L. K. Grover, From Schrödinger's equation to the quantum search algorithm, *Pramana* **56**, 333 (2001).
- [8] A. Aspuru-Guzik, A. D. Dutoi, P. J. Love, and M. Head-Gordon, Simulated quantum computation of molecular energies, *Science* **309**, 1704 (2005).
- [9] K. L. Brown, W. J. Munro, and V. M. Kendon, Using quantum computers for quantum simulation, *Entropy* **12**, 2268 (2010).
- [10] B. P. Lanyon, J. D. Whitfield, G. G. Gillett, M. E. Goggin, M. P. Almeida, I. Kassal, J. D. Biamonte, M. Mohseni, B. J. Powell, M. Barbieri *et al.*, Towards quantum chemistry on a quantum computer, *Nat. Chem.* **2**, 106 (2010).
- [11] J. D. Whitfield, J. Biamonte, and A. Aspuru-Guzik, Simulation of electronic structure Hamiltonians using quantum computers, *Mol. Phys.* **109**, 735 (2011).
- [12] Y. 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 (2019).
- [13] S. McArdle, S. Endo, A. Aspuru-Guzik, S. C. Benjamin, and X. Yuan, Quantum computational chemistry, *Rev. Mod. Phys.* **92**, 015003 (2020).
- [14] K. Kang, Y. S. Meng, J. Bréger, C. P. Grey, and G. Ceder, Electrodes with high power and high capacity for rechargeable lithium batteries, *Science* **311**, 977 (2006).
- [15] M. G. Mavros, T. Tsuchimochi, T. Kowalczyk, A. McIsaac, L.-P. Wang, and T. V. Voorhis, What can density functional theory tell us about artificial catalytic water splitting? *Inorg. Chem.* **53**, 6386 (2014).
- [16] P. Cudazzo, G. Profeta, A. Sanna, A. Floris, A. Continenza, S. Massidda, and E. K. U. Gross, Ab initio Description of High-Temperature Superconductivity in Dense Molecular Hydrogen, *Phys. Rev. Lett.* **100**, 257001 (2008).
- [17] T. H. Rod, A. Logadottir, and J. K. Nørskov, Ammonia synthesis at low temperatures, *J. Chem. Phys.* **112**, 5343 (2000).
- [18] J. K. Nørskov, M. Scheffler, and H. Toulhoat, Density functional theory in surface science and heterogeneous catalysis, *MRS Bull.* **31**, 669 (2006).

- [19] J. A. Flores-Livas, A. Sanna, and E. K. U. Gross, High temperature superconductivity in sulfur and selenium hydrides at high pressure, *Eur. Phys. J. B* **89**, 63 (2016).
- [20] P. Rydberg, F. S. Jørgensen, and L. Olsen, Use of density functional theory in drug metabolism studies, *Expert Opinion Drug Metab. Toxicol.* **10**, 215 (2014).
- [21] N. W. Ashcroft and N. D. Mermin, *Solid State Physics* (Saunders College, New York, 1976).
- [22] S. Jansen, M.-B. Ruskai, and R. Seiler, Bounds for the adiabatic approximation with applications to quantum computation, *J. Math. Phys.* **48**, 102111 (2007).
- [23] D. Wecker, B. Bauer, B. K. Clark, M. B. Hastings, and M. Troyer, Gate-count estimates for performing quantum chemistry on small quantum computers, *Phys. Rev. A* **90**, 022305 (2014).
- [24] D. Poulin, M. B. Hastings, D. Wecker, N. Wiebe, A. C. Doherty, and M. Troyer, The Trotter step size required for accurate quantum simulation of quantum chemistry, *Quantum Inf. Comput.* **15**, 361 (2015).
- [25] J. Lemieux, G. Duclos-Cianci, D. Sénéchal, and D. Poulin, Resource estimate for quantum many-body ground state preparation on a quantum computer, [arXiv:2006.04650](https://arxiv.org/abs/2006.04650) (2020).
- [26] J. L. Park, The concept of transition in quantum mechanics, *Found. Phys.* **1**, 23 (1970).
- [27] S. Aaronson, Shadow tomography of quantum states, in *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing* (IEEE, 2018), p. 325.
- [28] G. 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).
- [29] P. Hohenberg and W. Kohn, Inhomogeneous electron gas, *Phys. Rev.* **136**, B864 (1964).
- [30] W. Kohn and L. J. Sham, Self-consistent equations including exchange and correlation effects, *Phys. Rev.* **140**, A1133 (1965).
- [31] J. C. Snyder, M. Rupp, K. Hansen, K.-R. Müller, and K. Burke, Finding Density Functionals with Machine Learning, *Phys. Rev. Lett.* **108**, 253002 (2012).
- [32] J. Behler and M. Parrinello, Generalized Neural-Network Representation of High-Dimensional Potential-Energy Surfaces, *Phys. Rev. Lett.* **98**, 146401 (2007).
- [33] L. Li, T. E. Baker, S. R. White, and K. Burke, Pure density functional for strong correlation and the thermodynamic limit from machine learning, *Phys. Rev. B* **94**, 245129 (2016).
- [34] F. Brockherde, L. Vogt, L. Li, M. E. Tuckerman, K. Burke, and Klaus-Robert Müller, Bypassing the Kohn-Sham equations with machine learning, *Nat. Commun.* **8**, 872 (2017).
- [35] M. Bogojeski, L. Vogt-Maranto, M. E. Tuckerman, K.-R. Müller, and K. Burke, Density functionals with quantum chemical accuracy: From machine learning to molecular dynamics, [ChemRxiv](https://arxiv.org/abs/1903.05550) (2019), doi: [10.26434/chemrxiv.8079917.v1](https://doi.org/10.26434/chemrxiv.8079917.v1).
- [36] M. Bogojeski, F. Brockherde, L. Vogt-Maranto, L. Li, M. E. Tuckerman, K. Burke, and K.-R. Müller, Efficient prediction of 3D electron densities using machine learning, [arXiv:1811.06255](https://arxiv.org/abs/1811.06255) (2018).
- [37] R. Nagai, R. Akashi, S. Sasaki, and S. Tsuneyuki, Neural-network kohn-sham exchange-correlation potential and its out-of-training transferability, *J. Chem. Phys.* **148**, 241737 (2018).
- [38] L. Li, J. C. Snyder, I. M. Pelaschier, J. Huang, U.-N. Niranjan, P. Duncan, M. Rupp, K.-R. Müller, and K. Burke, Understanding machine-learned density functionals, *Int. J. Quantum Chem.* **116**, 819 (2016).
- [39] A. Grisafi, A. Fabrizio, B. Meyer, D. M. Wilkins, C. Corminboeuf, and M. Ceriotti, Transferable machine-learning model of the electron density, *ACS Central Sci.* **5**, 57 (2018).
- [40] A. Fabrizio, A. Grisafi, B. Meyer, M. Ceriotti, and C. Corminboeuf, Electron density learning of non-covalent systems, *Chem. Sci.* **10**, 9424 (2019).
- [41] M. M. Denner, M. H. Fischer, and T. Neupert, Efficient learning a one-dimensional density functional theory, *Phys. Rev. Research* **2**, 033388 (2020).
- [42] R. Nagai, R. Akashi, and O. Sugino, Completing density functional theory by machine learning hidden messages from molecules, *npj Comput. Mater.* **6**, 1 (2020).
- [43] S. Manzhos, Machine learning for the solution of the Schrödinger equation, *Machine Learning: Sci. Technol.* **1**, 013002 (2020).
- [44] Y. Suzuki, R. Nagai, and J. Haruyama, Machine learning exchange-correlation potential in time-dependent density functional theory, *Phys. Rev. A* **101**, 050501(R) (2020).
- [45] J. Wetherell, A. Costamagna, M. Gatti, and L. Reining, Insights into one-body density matrices using deep learning, *Faraday Discussions* (2020), doi: [10.1039/D0FD00061B](https://doi.org/10.1039/D0FD00061B).
- [46] J. C. Snyder, M. Rupp, K. Hansen, L. Blooston, K.-R. Müller, and K. Burke, Orbital-free bond breaking via machine learning, *J. Chem. Phys.* **139**, 224104 (2013).
- [47] J. C. Snyder, M. Rupp, K.-R. Müller, and K. Burke, Nonlinear gradient denoising: Finding accurate extrema from inaccurate functional derivatives, *Int. J. Quant. Chem.* **115**, 1102 (2015).
- [48] K. Vu, J. C. Snyder, L. Li, M. Rupp, B. F. Chen, T. Khelif, K.-R. Müller, and K. Burke, Understanding kernel ridge regression: Common behaviors from simple functions to density functionals, *Int. J. Quant. Chem.* **115**, 1115 (2015).
- [49] J. Hollingsworth, L. Li, T. E. Baker, and K. Burke, Can exact conditions improve machine-learned density functionals? *J. Chem. Phys.* **148**, 241743 (2018).
- [50] E. Engel and R. M. Dreizler, *Density Functional Theory: An Advanced Course* (Springer Science & Business Media, 2011).
- [51] E. K. U. Gross and R. M. Dreizler, *Density Functional Theory*, NATO ASI Series Vol. 337 (Springer Science & Business Media, 2013).
- [52] R. Hatcher, J. A. Kittl, and C. Bowen, A method to calculate correlation for density functional theory on a quantum processor, [arXiv:1903.05550](https://arxiv.org/abs/1903.05550) (2019).
- [53] J. D. Whitfield, M. H. Yung, D. G. Tempel, S. Boixo, and A. Aspuru-Guzik, Computational complexity of time-dependent density functional theory, *New. J. Phys.* **16**, 083035 (2014).
- [54] J. Brown, J. Yang, and J. D. Whitfield, Solver for the electronic V-representation problem of time-dependent density functional theory, *J. Chem. Theory Comput.* **16**, 6014 (2020).
- [55] J. Yang, J. Brown, and J. D. Whitfield, Measurement on quantum devices with applications to time-dependent density functional theory, [arXiv:1909.03078](https://arxiv.org/abs/1909.03078) (2019).
- [56] P. Rall, Quantum algorithms for estimating physical quantities using block encodings, *Phys. Rev. A* **102**, 022408 (2020).

- [57] C. A. Ullrich, *Time-Dependent Density-Functional Theory: Concepts and Applications* (Oxford University Press, Oxford, 2011).
- [58] K. Temme, T. J. Osborne, K. G. Vollbrecht, D. Poulin, and F. Verstraete, Quantum metropolis sampling, *Nature (London)* **471**, 87 (2011).
- [59] T. E. Baker, Lanczos recursion on a quantum computer for the Green's function, [arXiv:2008.05593](https://arxiv.org/abs/2008.05593) (2020).
- [60] R. A. Servedio and S. J. Gortler, Equivalences and separations between quantum and classical learnability, *SIAM J. Comput.* **33**, 1067 (2004).
- [61] E. Tang, A quantum-inspired classical algorithm for recommendation systems, in *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing* (Association for Computing Machinery, NY, USA, 2019), p. 217.
- [62] E. Tang, Quantum-inspired classical algorithms for principal component analysis and supervised clustering, [arXiv:1811.00414](https://arxiv.org/abs/1811.00414) (2018).
- [63] A. Gilyén, S. Lloyd, and E. Tang, Quantum-inspired low-rank stochastic regression with logarithmic dependence on the dimension, [arXiv:1811.04909](https://arxiv.org/abs/1811.04909) (2018).
- [64] N.-H. Chia, A. Gilyén, T. Li, H.-H. Lin, E. Tang, and C. Wang, Sampling-based sublinear low-rank matrix arithmetic framework for dequantizing quantum machine learning, in *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing* (Association for Computing Machinery, NY, USA, 2020), p. 387.
- [65] G. H. Low and I. L. Chuang, Hamiltonian simulation by qubitization, *Quantum* **3**, 163 (2019).
- [66] W. Yang, P. W. Ayers, and Q. Wu, Potential Functionals: Dual to Density Functionals and Solution to the  $v$ -Representability Problem, *Phys. Rev. Lett.* **92**, 146404 (2004).
- [67] J. P. Perdew, What do the Kohn-Sham orbital energies mean? How do atoms dissociate? in *Density Functional Methods in Physics*, NATO ASI Series (Springer, 1985), p. 265.
- [68] J. P. Perdew and M. Levy, Comment on "Significance of the highest occupied Kohn-Sham eigenvalue," *Phys. Rev. B* **56**, 16021 (1997).
- [69] M. Levy, Universal variational functionals of electron densities, first-order density matrices, and natural spin-orbitals and solution of the  $v$ -representability problem, *Proc. Natl. Acad. Sci. U. S. A.* **76**, 6062 (1979).
- [70] W. Kohn,  $v$ -Representability and Density Functional Theory, *Phys. Rev. Lett.* **51**, 1596 (1983).
- [71] J. T. Chayes, L. Chayes, and M. B. Ruskai, Density functional approach to quantum lattice systems, *J. Stat. Phys.* **38**, 497 (1985).
- [72] L. O. Wagner, T. E. Baker, E. M. Stoudenmire, K. Burke, and S. R. White, Kohn-Sham calculations with the exact functional, *Phys. Rev. B* **90**, 045109 (2014).
- [73] N. I. Gidopoulos, Progress at the interface of wave-function and density-functional theories, *Phys. Rev. A* **83**, 040502(R) (2011).
- [74] T. J. Callow and N. I. Gidopoulos, Optimal power series expansions of the Kohn-Sham potential, *Eur. Phys. J. B* **91**, 209 (2018).
- [75] T. J. Callow, N. N. Lathiotakis, and N. I. Gidopoulos, Density-inversion method for the Kohn-Sham potential: Role of the screening density, *J. Chem. Phys.* **152**, 164114 (2020).
- [76] D. S. Jensen and A. Wasserman, Numerical density-to-potential inversions in time-dependent density functional theory, *Phys. Chem. Chem. Phys.* **18**, 21079 (2016).
- [77] D. S. Jensen and A. Wasserman, Numerical methods for the inverse problem of density functional theory, *Int. J. Quant. Chem.* **118**, e25425 (2018).
- [78] B. Kanungo, P. M. Zimmerman, and V. Gavini, Exact exchange-correlation potentials from ground-state electron densities, *Nat. Commun.* **10**, 1 (2019).
- [79] A. Kumar and M. K. Harbola, A general penalty method for density-to-potential inversion, [arXiv:2004.03219](https://arxiv.org/abs/2004.03219) (2020).
- [80] T. G. Draper, Addition on a quantum computer, [arXiv:quant-ph/0008033](https://arxiv.org/abs/quant-ph/0008033) (2000).
- [81] S. P. Jordan, Fast Quantum Algorithm for Numerical Gradient Estimation, *Phys. Rev. Lett.* **95**, 050501 (2005).
- [82] A. Gilyén, S. Arunachalam, and N. Wiebe, Optimizing quantum optimization algorithms via faster quantum gradient computation, in *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms* (Society for Industrial and Applied Mathematics, 2019), p. 1425.
- [83] M. G. Medvedev, I. S. Bushmarinov, J. Sun, J. P. Perdew, and K. A. Lyssenko, Density functional theory is straying from the path toward the exact functional, *Science* **355**, 49 (2017).
- [84] T. E. Baker, Methods of calculation with the exact density functional using the renormalization group, Ph.D. thesis, University of California, Irvine, 2017; J. P. Perdew and T. E. Baker, IPAM Book of DFT: The Generalized Gradient Approximation, Thesis, 2017 (unpublished), Chap. 6.
- [85] A. Kumar and M. K. Harbola, Using random numbers to obtain Kohn-Sham potential for a given density, [arXiv:2006.00324](https://arxiv.org/abs/2006.00324) (2020).
- [86] E. K. U Gross and C. R. Proetto, Adiabatic connection and the Kohn-Sham variety of potential-functional theory, *J. Chem. Theory Comput.* **5**, 844 (2009).
- [87] M. F. Langer, A. Goeßmann, and M. Rupp, Representations of molecules and materials for interpolation of quantum-mechanical simulations via machine learning, [arXiv:2003.12081](https://arxiv.org/abs/2003.12081) (2020).
- [88] M. B. Hastings, Locality in Quantum and Markov Dynamics on Lattices and Networks, *Phys. Rev. Lett.* **93**, 140402 (2004).
- [89] W. Kohn, Density Functional and Density Matrix Method Scaling Linearly with the Number of Atoms, *Phys. Rev. Lett.* **76**, 3168 (1996).
- [90] E. Prodan and W. Kohn, Nearsightedness of electronic matter, *Proc. Nat. Acad. Sci. U.S.A.* **102**, 11635 (2005).
- [91] T. E. Baker, S. Desrosiers, M. Tremblay, and M. P. Thompson, Méthodes de calcul avec réseaux de tenseurs en physique (Basic tensor network computations in physics), Canadian J. Phys. (to be published), [arXiv:1911.11566](https://arxiv.org/abs/1911.11566).
- [92] G. Vidal, Class of Quantum Many-Body States that can be Efficiently Simulated, *Phys. Rev. Lett.* **101**, 110501 (2008).
- [93] L. O. Wagner, E. M. Stoudenmire, K. Burke, and S. R. White, Guaranteed Convergence of the Kohn-Sham Equations, *Phys. Rev. Lett.* **111**, 093003 (2013).
- [94] Z. Li, J. R. Kermode, and A. De Vita, Molecular Dynamics with On-The-Fly Machine Learning of Quantum-Mechanical Forces, *Phys. Rev. Lett.* **114**, 096405 (2015).
- [95] T. L. Gilbert, Hohenberg-Kohn theorem for nonlocal external potentials, *Phys. Rev. B* **12**, 2111 (1975).

- [96] N. I. Gidopoulos and E. K. U Gross, Electronic non-adiabatic states: Towards a density functional theory beyond the Born–Oppenheimer approximation, *Phil. Trans. R. Soc. A* **372**, 20130059 (2014).
- [97] E. Runge and E. K. U. Gross, Density-Functional Theory for Time-Dependent Systems, *Phys. Rev. Lett.* **52**, 997 (1984).
- [98] R. van Leeuwen, Causality and Symmetry in Time-Dependent Density-Functional Theory, *Phys. Rev. Lett.* **80**, 1280 (1998).
- [99] P. Elliott, F. Furche, and K. Burke, 3 excited states from time-dependent density functional theory, *Rev. Comput. Chem.* **26**, 91 (2009).
- [100] N. D. Mermin, Thermal properties of the inhomogeneous electron gas, *Phys. Rev.* **137**, A1441 (1965).
- [101] W. Kohn, E. K. U Gross, and L. N. Oliveira, Orbital magnetism in the density functional theory of superconductors, *J. Phys.* **50**, 2601 (1989).
- [102] K. Capelle and E. K. U Gross, Density functional theory for triplet superconductors, *Int. J. Quant. Chem.* **61**, 325 (1997).
- [103] M. Ruggenthaler, F Mackenroth, and D. Bauer, Time-dependent Kohn-Sham approach to quantum electrodynamics, *Phys. Rev. A* **84**, 042107 (2011).
- [104] I. V. Tokatly, Time-Dependent Density Functional Theory for Many-Electron Systems Interacting with Cavity Photons, *Phys. Rev. Lett.* **110**, 233001 (2013).
- [105] M. Ruggenthaler, J. Flick, C. Pellegrini, H. Appel, I. V. Tokatly, and A. Rubio, Quantum-electrodynamical density-functional theory: Bridging quantum optics and electronic-structure theory, *Phys. Rev. A* **90**, 012508 (2014).
- [106] M. Filatov, Ensemble dft approach to excited states of strongly correlated molecular systems, in *Density-Functional Methods for Excited States*, Topics in Current Chemistry (Springer, 2015), p. 97.
- [107] P. Jouzdani, S. Bringuier, and M. Kostuk, A method of determining excited-states for quantum computation, [arXiv:1908.05238](https://arxiv.org/abs/1908.05238) (2019).
- [108] M. H. Cohen and A. Wasserman, On hardness and electronegativity equalization in chemical reactivity theory, *J. Stat. Phys.* **125**, 1121 (2006).
- [109] M. H. Cohen and A. Wasserman, On the foundations of chemical reactivity theory, *J. Phys. Chem. A* **111**, 2229 (2007).
- [110] P. Elliott, K. Burke, M. H. Cohen, and A. Wasserman, Partition density-functional theory, *Phys. Rev. A* **82**, 024501 (2010).
- [111] A. Cangi, D. Lee, P. Elliott, K. Burke, and E. K. U. Gross, Electronic Structure via Potential Functional Approximations, *Phys. Rev. Lett.* **106**, 236404 (2011).
- [112] A. Cangi, E. K. U Gross, and K. Burke, Potential functionals versus density functionals, *Phys. Rev. A* **88**, 062505 (2013).
- [113] J. R. Schrieffer, *Theory of Superconductivity* (W. A. Benjamin, New York, 1964).
- [114] A. N. Chowdhury and R. D. Somma, Quantum algorithms for Gibbs sampling and hitting-time estimation, *Quantum Info. Comput.* **17**, 41 (2017).
- [115] G. H. Low and I. L. Chuang, Optimal Hamiltonian Simulation by Quantum Signal Processing, *Phys. Rev. Lett.* **118**, 010501 (2017).
- [116] T. R. Mattsson and M. P. Desjarlais, Phase Diagram and Electrical Conductivity of High Energy-Density Water from Density Functional Theory, *Phys. Rev. Lett.* **97**, 017801 (2006).
- [117] D. Poulin and P. Wocjan, Sampling from the Thermal Quantum Gibbs State and Evaluating Partition Functions with a Quantum Computer, *Phys. Rev. Lett.* **103**, 220502 (2009).
- [118] E. M. Stoudenmire, L. O. Wagner, S. R. White, and K. Burke, One-Dimensional Continuum Electronic Structure with the Density-Matrix Renormalization Group and its Implications for Density-Functional Theory, *Phys. Rev. Lett.* **109**, 056402 (2012).
- [119] L. Shulenburger, M. Casula, G. Senatore, and R. M. Martin, Spin resolved energy parametrization of a quasi-one-dimensional electron gas, *J. Phys. A: Math. Theor.* **42**, 214021 (2009).
- [120] L. O. Wagner, E. M. Stoudenmire, K. Burke, and S. R. White, Reference electronic structure calculations in one dimension, *Phys. Chem. Chem. Phys.* **14**, 8581 (2012).
- [121] T. E. Baker, E. M. Stoudenmire, L. O. Wagner, K. Burke, and S. R. White, One-dimensional mimicking of electronic structure: The case for exponentials, *Phys. Rev. B* **91**, 235141 (2015); Erratum: One-dimensional mimicking of electronic structure: The case for exponentials [*Phys. Rev. B* **91**, 235141 (2015)], **93**, 119912(E) (2016).
- [122] N. Helbig, I. V. Tokatly, and A. Rubio, Exact Kohn–Sham potential of strongly correlated finite systems, *J. Chem. Phys.* **131**, 224105 (2009).
- [123] N. Helbig, J. I. Fuks, M. Casula, M. J. Verstraete, M. A. L. Marques, I. V. Tokatly, and A. Rubio, Density functional theory beyond the linear regime: Validating an adiabatic local density approximation, *Phys. Rev. A* **83**, 032503 (2011).
- [124] P. Elliott, J. I. Fuks, A. Rubio, and N. T. Maitra, Universal Dynamical Steps in the Exact Time-Dependent Exchange-Correlation Potential, *Phys. Rev. Lett.* **109**, 266404 (2012).
- [125] J. I. Fuks, K. Luo, E. D. Sandoval, and N. T. Maitra, Time-Resolved Spectroscopy in Time-Dependent Density Functional Theory: An Exact Condition, *Phys. Rev. Lett.* **114**, 183002 (2015).
- [126] N. A. Lima, M. F. Silva, L. N. Oliveira, and K. Capelle, Density Functionals not Based on the Electron Gas: Local-Density Approximation for a Luttinger Liquid, *Phys. Rev. Lett.* **90**, 146402 (2003).
- [127] J. I. Fuks and N. T. Maitra, Challenging adiabatic time-dependent density functional theory with a Hubbard dimer: The case of time-resolved long-range charge transfer, *Phys. Chem. Chem. Phys.* **16**, 14504 (2014).
- [128] J. I. Fuks and N. T. Maitra, Charge transfer in time-dependent density-functional theory: Insights from the asymmetric Hubbard dimer, *Phys. Rev. A* **89**, 062502 (2014).
- [129] A. J. Cohen and P. Mori-Sánchez, Landscape of an exact energy functional, *Phys. Rev. A* **93**, 042511 (2016).
- [130] D. J. Carrascal, J. Ferrer, J. C. Smith, and K. Burke, The Hubbard dimer: A density functional case study of a many-body problem, *J. Phys.: Condens. Matter* **27**, 393001 (2015).
- [131] D. J. Carrascal, J. Ferrer, N. Maitra, and K. Burke, Linear response time-dependent density functional theory of the Hubbard dimer, *Eur. Phys. J. B* **91**, 142 (2018).
- [132] J. C. Smith and K. Burke, Thermal stitching: Combining the advantages of different quantum fermion solvers, *Phys. Rev. B* **98**, 075148 (2018).
- [133] F. Sagredo and K. Burke, Accurate double excitations from ensemble density functional calculations, *J. Chem. Phys.* **149**, 134103 (2018).

- [134] J. C. Smith, A. Pribram-Jones, and K. Burke, Exact thermal density functional theory for a model system: Correlation components and accuracy of the zero-temperature exchange-correlation approximation, *Phys. Rev. B* **93**, 245131 (2016).
- [135] M. Herrera, K. Zawadzki, and I. D'Amico, Melting a Hubbard dimer: Benchmarks of *alda'* for quantum thermodynamics, *Eur. Phys. J. B* **91**, 248 (2018).
- [136] W. A. Vigor, J. S. Spencer, M. J. Bearpark, and A. J. W. Thom, Minimising biases in full configuration interaction quantum Monte Carlo, *J. Chem. Phys.* **142**, 104101 (2015).
- [137] J. R. McClean, R. Babbush, P. J. Love, and A. Aspuru-Guzik, Exploiting locality in quantum computation for quantum chemistry, *J. Phys. Chem. Lett.* **5**, 4368 (2014).
- [138] U. Schollwöck, The density-matrix renormalization group in the age of matrix product states, *Ann. Phys.* **326**, 96 (2011).
- [139] G. K.-L. Chan and S. Sharma, The density matrix renormalization group in quantum chemistry, *Annu. Rev. Phys. Chem.* **62**, 465 (2011).
- [140] J. Yang, W. Hu, D. Usvyat, D. Matthews, M. Schütz, and G. K.-L. Chan, *Ab initio* determination of the crystalline benzene lattice energy to sub-kilojoule/mole accuracy, *Science* **345**, 640 (2014).
- [141] O. A. Arodola and M. E. S. Soliman, Quantum mechanics implementation in drug-design workflows: Does it really help? *Drug Design Develop. Therapy* **11**, 2551 (2017).
- [142] W. M. C. Foulkes, L. Mitas, R. J. Needs, and G. Rajagopal, Quantum Monte Carlo simulations of solids, *Rev. Mod. Phys.* **73**, 33 (2001).
- [143] H. Eshuis, J. E. Bates, and F. Furche, Electron correlation methods based on the random phase approximation, *Theor. Chem. Acc.* **131**, 1084 (2012).
- [144] G. P. Chen, V. K. Voora, M. M. Agee, S. G. Balasubramani, and F. Furche, Random-phase approximation methods, *Annu. Rev. Phys. Chem.* **68**, 421 (2017).
- [145] T. Helgaker, P. Jorgensen, and J. Olsen, *Molecular Electronic-Structure Theory* (John Wiley & Sons, 2014).
- [146] J. A. Pople, Nobel lecture: Quantum chemical models, *Rev. Mod. Phys.* **71**, 1267 (1999).
- [147] R. J. Bartlett and M. Musiał, Coupled-cluster theory in quantum chemistry, *Rev. Mod. Phys.* **79**, 291 (2007).
- [148] R. Babbush, N. Wiebe, J. McClean, J. McClain, H. Neven, and G. K. L. Chan, Low-depth quantum simulation of materials, *Phys. Rev. X* **8**, 011044 (2018).
- [149] J. C. Kimball, Short-range correlations and the structure factor and momentum distribution of electrons, *J. Phys. A: Math. Gen.* **8**, 1513 (1975).
- [150] B. Klahn and J. D. Morgan III, Rates of convergence of variational calculations and of expectation values, *J. Chem. Phys.* **81**, 410 (1984).
- [151] R. N. Hill, Rates of convergence and error estimation formulas for the Rayleigh–Ritz variational method, *J. Chem. Phys.* **83**, 1173 (1985).
- [152] T. E. Baker, K. Burke, and S. R. White, Accurate correlation energies in one-dimensional systems from small system-adapted basis functions, *Phys. Rev. B* **97**, 085139 (2018).
- [153] G. Beylkin and M. J. Mohlenkamp, Numerical operator calculus in higher dimensions, *Proc. Natl. Acad. Sci. U. S. A.* **99**, 10246 (2002).
- [154] R. Oliveira and B. M. Terhal, The complexity of quantum spin systems on a two-dimensional square lattice, *Quant. Inf. Comput.* **8**, 900 (2008).
- [155] S. McArdle, T. Jones, S. Endo, Y. Li, S. C. Benjamin, and X. Yuan, Variational ansatz-based quantum simulation of imaginary time evolution, *npj Quantum Inf.* **5**, 1 (2019).
- [156] A. Hackl and S. Kehrein, Real time evolution in quantum many-body systems with unitary perturbation theory, *Phys. Rev. B* **78**, 092303 (2008).
- [157] M. Schwarz, K. Temme, and F. Verstraete, Preparing Projected Entangled Pair States on a Quantum Computer, *Phys. Rev. Lett.* **108**, 110502 (2012).
- [158] A. Gilyén, Y. Su, G. H. Low, and N. Wiebe, Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics, in *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing* (Association for Computing Machinery, NY, USA, 2019), p. 193.
- [159] N. S. Yanofsky and M. A. Mannucci, *Quantum Computing for Computer Scientists* (Cambridge University Press, 2008).
- [160] Y.-K. Liu, M. Christandl, and F. Verstraete, Quantum Computational Complexity of the  $n$ -Representability Problem: Qma Complete, *Phys. Rev. Lett.* **98**, 110503 (2007).
- [161] N. Schuch and F. Verstraete, Computational complexity of interacting electrons and fundamental limitations of density functional theory, *Nat. Phys.* **5**, 732 (2009).
- [162] M. Szegedy, Quantum speed-up of Markov chain based algorithms, in *45th Annual IEEE Symposium on Foundations of Computer Science* (IEEE, 2004), p. 32.
- [163] J. Lemieux, B. Heim, D. Poulin, K. Svore, and M. Troyer, Efficient quantum walk circuits for Metropolis-Hastings algorithm, *Quantum* **4**, 287 (2020).
- [164] A. 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).
- [165] D. Ristè, M. P. Da Silva, C. A. Ryan, A. W. Cross, A. D. Córcoles, J. A. Smolin, J. M. Gambetta, J. M. Chow, and B. R. Johnson, Demonstration of quantum advantage in machine learning, *npj Quantum Inf.* **3**, 16 (2017).
- [166] S. Arunachalam and R. De Wolf, Optimal quantum sample complexity of learning algorithms, *J. Machine Learning Res.* **19**, 2879 (2018).
- [167] Theorems listed here are so heavily dependent on preexisting proofs that they are probably more accurately called a corollary of those theorems, but this naming convention is used in several physics works and here to match.
- [168] J. D. Whitfield, P. J. Love, and A. Aspuru-Guzik, Computational complexity in electronic structure, *Phys. Chem. Chem. Phys.* **15**, 397 (2013).
- [169] L. G. Valiant, A theory of the learnable, *Commun. ACM* **27**, 1134 (1984).
- [170] J. B. Amsterdam, The valiant learning model: Extensions and assessment, Ph.D. thesis, Department of Electrical Engineering, Massachusetts Institute of Technology, 1988.
- [171] F. Bergadano and L. Saitta, On the error probability of Boolean concept descriptions, in *Proceedings of the 1989 European Working Session on Learning* (Morgan Kaufmann Publishers Inc., CA, United States, 1989), p. 25.

- [172] D. Haussler, Probably Approximately Correct Learning, *Encyclopedia of Machine Learning and Data Mining*, edited by C. Sammut and G. I. Webb (Springer, Boston, US, 2017), pp. 1017–1017.
- [173] W. L. Buntine, A theory of learning classification rules, Ph.D. thesis, University of Technology, Sydney, 1990.
- [174] M. J. Pazzani and W. Sarrett, A framework for average case analysis of conjunctive learning algorithms, *Mach. Learn.* **9**, 349 (1992).
- [175] A. L. Fetter and J. D. Walecka, *Quantum Theory of Many-Particle Systems* (Courier Corporation, 2012).
- [176] S. Raimes, *Many-electron theory* (North-Holland, 1972).
- [177] J. Hubbard, Electron correlations in narrow energy bands, *Proc. R. Soc. Lond. A* **276**, 238 (1963).
- [178] M. Born and R. Oppenheimer, Zur Quantentheorie der Molekeln, *Ann. Phys.* **389**, 457 (1927).
- [179] S. F. Boys, Electronic wave functions. I. A general method of calculation for the stationary states of any molecular system, in *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, Vol. 200 (Royal Society, London, 1950), p. 542–554.
- [180] F. A. Bischoff, R. J. Harrison, and E. F. Valeev, Computing many-body wave functions with guaranteed precision: The first-order Møller-Plesset wave function for the ground state of helium atom, *J. Chem. Phys.* **137**, 104103 (2012).
- [181] R. J. Harrison, G. Beylkin, F. A. Bischoff, J. A. Calvin, G. I. Fann, J. Fosso-Tande, D. Galindo, J. R. Hammond, R. Hartman-Baker, J. C. Hill *et al.*, MADNESS: A multiresolution, adaptive numerical environment for scientific simulation, *SIAM J. Sci. Comput.* **38**, S123 (2016).
- [182] M. Reed and B. Simon, *Methods of Modern Mathematical Physics: Functional Analysis* (Elsevier, 2012).
- [183] M. Levy and J. P. Perdew, The constrained search formulation of density functional theory, in *Density Functional Methods in Physics* (Springer, 1985), p. 11.
- [184] E. Fermi, Un metodo statistico per la determinazione di alcune priorietà dell'atome, *Rend. Accad. Naz. Lincei* **6**, 32 (1927).
- [185] L. H. Thomas, The calculation of atomic fields, in *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 23 (Cambridge University Press, 1927), p. 542.
- [186] C. V. Weizsäcker, Zur Theorie der Nernmassen, *Z. Phys. A* **96**, 431 (1935).
- [187] R. F. Ribeiro, D. Lee, A. Cangi, P. Elliott, and K. Burke, Corrections to Thomas-Fermi Densities at Turning Points and Beyond, *Phys. Rev. Lett.* **114**, 050401 (2015).
- [188] E. H. Lieb and S. Oxford, Improved lower bound on the indirect Coulomb energy, *Int. J. Quant. Chem.* **19**, 427 (1981).
- [189] S. Pittalis, C. R. Proetto, A. Floris, A. Sanna, C. Bersier, K. Burke, and E. K. U. Gross, Exact Conditions in Finite-Temperature Density-Functional Theory, *Phys. Rev. Lett.* **107**, 163001 (2011).
- [190] J. Sun, A. Ruzsinszky, and J. P. Perdew, Strongly Constrained and Appropriately Normed Semilocal Density Functional, *Phys. Rev. Lett.* **115**, 036402 (2015).
- [191] P. Mori-Sánchez, A. J. Cohen, and W. Yang, Localization and Delocalization Errors in Density Functional Theory and Implications for Band-Gap Prediction, *Phys. Rev. Lett.* **100**, 146401 (2008).
- [192] A. J. Cohen, P. Mori-Sánchez, and W. Yang, Insights into current limitations of density functional theory, *Science* **321**, 792 (2008).
- [193] H. Goldstein, C. P. Poole, and J. L. Safko, *Classical Mechanics* (Pearson Higher Ed, 2014).
- [194] E. K. U. Gross and N. T. Maitra, Introduction to TDDFT, in *Fundamentals of Time-Dependent Density Functional Theory* (Springer, 2012), p. 53.
- [195] M. Seidl, Strong-interaction limit of density-functional theory, *Phys. Rev. A* **60**, 4387 (1999).
- [196] D. C. Langreth and J. P. Perdew, The exchange-correlation energy of a metallic surface, *Solid State Commun.* **17**, 1425 (1975).
- [197] M. Seidl, J. P. Perdew, and M. Levy, Strictly correlated electrons in density-functional theory, *Phys. Rev. A* **59**, 51 (1999).
- [198] J. S. Townsend, *A Modern Approach to Quantum Mechanics* (University Science Books, 2000).
- [199] A. Görling and M. Levy, Exact Kohn-Sham scheme based on perturbation theory, *Phys. Rev. A* **50**, 196 (1994).
- [200] A. Pribram-Jones, S. Pittalis, E. K. U. Gross, and K. Burke, Thermal density functional theory in context, in *Frontiers and Challenges in Warm Dense Matter* (Springer, 2014), p. 25.
- [201] D. M. Ceperley and B. J. Alder, Ground State of the Electron Gas by a Stochastic Method, *Phys. Rev. Lett.* **45**, 566 (1980).
- [202] S. H. Vosko, L. Wilk, and M. Nusair, Accurate spin-dependent electron liquid correlation energies for local spin density calculations: A critical analysis, *Can. J. Phys.* **58**, 1200 (1980).
- [203] A. D. Becke, A new mixing of Hartree-Fock and local density-functional theories, *J. Chem. Phys.* **98**, 1372 (1993).
- [204] J. P. Perdew, R. G. Parr, M. Levy, and J. L. Balduz Jr, Density-Functional Theory for Fractional Particle Number: Derivative Discontinuities of the Energy, *Phys. Rev. Lett.* **49**, 1691 (1982).
- [205] S. Ruder, An overview of gradient descent optimization algorithms, [arXiv:1609.04747](https://arxiv.org/abs/1609.04747) (2016).
- [206] V. Černý, Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm, *J. Optim. Theory Appl.* **45**, 41 (1985).
- [207] D. Bertsimas and J. Tsitsiklis, Simulated annealing, *Stat. Sci.* **8**, 10 (1993).
- [208] L. Tierney, Markov chains for exploring posterior distributions, *Ann. Stat.* **22**, 1701 (1994).
- [209] S. V. N. Vishwanathan, K. M. Borgwardt, and N. N. Schraudolph, Fast computation of graph kernels, in *Advances in Neural Information Processing Systems*, edited by B. Schölkopf and J. Platt, and T. Hoffman (MIT Press, 2006), Vol. 19, pp. 1449–1456.
- [210] R. M. Neal, *Bayesian Learning for Neural Networks*, Lecture Notes in Statistics Vol. 118 (Springer Science & Business Media, 2012).
- [211] L. M. Rios and N. V. Sahinidis, Derivative-free optimization: A review of algorithms and comparison of software implementations, *J. Global Optim.* **56**, 1247 (2013).
- [212] N. S. Pillai, A. M. Stuart, and A. H. Thiéry, Noisy gradient flow from a random walk in hilbert space, *Stoch. Partial Diff. Eq.: Anal. Comput.* **2**, 196 (2014).
- [213] B. Perozzi, V. Kulkarni, and S. Skiena, Walklets: Multiscale graph embeddings for interpretable network classification, [arXiv:1605.02115](https://arxiv.org/abs/1605.02115) (2016).

- [214] V. Tejedor, A dimensional acceleration of gradient descent-like methods, using persistent random walkers, [arXiv:1801.04532](#) (2018).
- [215] S. Ghosh, N. Das, T. Gonçalves, P. Quaresma, and M. Kundu, The journey of graph kernels through two decades, *Comput. Sci. Rev.* **27**, 88 (2018).
- [216] V. Tejedor, R. Voituriez, and O. Bénichou, Optimizing Persistent Random Searches, *Phys. Rev. Lett.* **108**, 088103 (2012).
- [217] D. Maclaurin, D. Duvenaud, and R. Adams, Gradient-based hyperparameter optimization through reversible learning, in *Proceedings of the 32nd International Conference on Machine Learning*, edited by F. Bach and D. Blei (PMLR, 2015), Vol. 37, pp. 2113–2122.
- [218] A. Blais, Algorithmes et architectures pour ordinateurs quantiques supraconducteurs, in *Annales de Physique* (EDP Sciences, 2003), Vol. 28, p. 1.
- [219] C. Kittel, *Quantum Theory of Solids* (Wiley, 1987).
- [220] L. Hales and S. Hallgren, An improved quantum Fourier transform algorithm and applications, in *Foundations of Computer Science, 2000, Proceedings, 41st Annual Symposium on* (IEEE, 2000), p. 515.
- [221] M. Suzuki, Improved Trotter-like formula, *Phys. Lett. A* **180**, 232 (1993).
- [222] D. Poulin, A. Kitaev, D. S. Steiger, M. B. Hastings, and M. Troyer, Quantum Algorithm for Spectral Measurement with a Lower Gate Count, *Phys. Rev. Lett.* **121**, 010501 (2018).
- [223] E. Novak and H. Woźniakowski, *Tractability of Multivariate Problems: Standard Information for Functionals*, Vol. 12 (European Mathematical Society, 2008).
- [224] S. Oh, Quantum computational method of finding the ground-state energy and expectation values, *Phys. Rev. A* **77**, 012326 (2008).
- [225] I. Kassal, J. D. Whitfield, A. Perdomo-Ortiz, M.-H. Yung, and A. Aspuru-Guzik, Simulating chemistry using quantum computers, *Annu. Rev. Phys. Chem.* **62**, 185 (2011).
- [226] This algorithm can go by other names, notably quantum counting, but we use “QAE” instead of the abbreviated “QC” for quantum counting since this would overlap with “quantum chemistry” and “quantum computing” if we chose to also use abbreviations for those.
- [227] E. Knill, G. Ortiz, and R. D. Somma, Optimal quantum measurements of expectation values of observables, *Phys. Rev. A* **75**, 012328 (2007).
- [228] C. Marriott and J. Watrous, Quantum Arthur–Merlin games, *Comput. Complex.* **14**, 122 (2005).