THE JOURNAL OF CHEMICAL PHYSICS **131**, 224102 (2009)

Quantum algorithm for molecular properties and geometry optimization

Ivan Kassal[a] and Alán Aspuru-Guzik[b]
Department of Chemistry and Chemical Biology, Harvard University, Cambridge, Massachusetts 02138, USA

(Received 13 August 2009; accepted 3 November 2009; published online 8 December 2009)

Quantum computers, if available, could substantially accelerate quantum simulations. We extend this result to show that the computation of molecular properties (energy derivatives) could also be sped up using quantum computers. We provide a quantum algorithm for the numerical evaluation of molecular properties, whose time cost is a constant multiple of the time needed to compute the molecular energy, regardless of the size of the system. Molecular properties computed with the proposed approach could also be used for the optimization of molecular geometries or other properties. For that purpose, we discuss the benefits of quantum techniques for Newton's method and Householder methods. Finally, global minima for the proposed optimizations can be found using the quantum basin hopper algorithm, which offers an additional quadratic reduction in cost over classical multi-start techniques. © *2009 American Institute of Physics*. [doi:10.1063/1.3266959]

Applying *ab initio* methods of quantum chemistry to particular problems often requires computing derivatives of the molecular energy. For instance, obtaining a molecule's electric properties relies on the ability to compute derivatives with respect to external electromagnetic fields. Likewise, computing the gradient of the molecular energy with respect to the nuclear coordinates is the most commonly used method for the proper characterization of potential energy surfaces and for optimizing the geometry of all but the smallest molecules. The computation of these kinds of derivatives, known as *molecular properties*, is nowadays a routine matter when it comes to low-order derivatives or small systems (or both). This is largely due to advances in analytical gradient techniques, which allow for explicit property evaluation without resorting to numerical differentiation.[1–8]

Nevertheless, the computation of higher-order derivatives is often prohibitively expensive, even though such derivatives are often needed. For example, third- and fourth-order anharmonic constants are sometimes required to accurately compute a vibrational absorption spectrum[4] or efficiently determine the location of transition states on complex potential energy surfaces.[7] Other properties of interest, such as hyperpolarizabilities, Raman intensities, or vibrational circular dichroism, are also cubic or quartic derivatives. In this report, we show that quantum computers, once available, will be able to bypass some of the high cost of computing these properties. In particular, we show that any molecular property can be evaluated on a quantum computer using resources that, up to a small constant, are equal to those required to compute the molecular energy once. We have previously characterized the advantage of quantum computers at both computing molecular energies[9,10] and simulating chemical reaction dynamics,[11] and the present work extends our program to molecular properties.

This paper begins with a brief overview of classical techniques for the evaluation of molecular properties, both numerical and analytical. We then introduce the quantum algorithm for molecular properties, and discuss its advantages and disadvantages with respect to classical techniques. We conclude with geometry optimization as a particular example, and we show that it can benefit from an additional quadratic speed-up through Grover's search.[12]

I. THE CLASSICAL METHODS

Given an external perturbation $\boldsymbol{\mu}$, the total molecular electronic energy can be expanded in a Taylor series

$$E(\boldsymbol{\mu}) = E^{(0)} + \boldsymbol{\mu}^{\top}\mathbf{E}^{(1)} + \tfrac{1}{2}\boldsymbol{\mu}^{\top}\mathbf{E}^{(2)}\boldsymbol{\mu} + \cdots, \tag{1}$$

where the coefficients $\mathbf{E}^{(n)}$ are called the *molecular properties* and describe the response of the system to the applied perturbation.[8] We consider time-independent properties, which can be obtained by differentiating the energy at $\boldsymbol{\mu}=\mathbf{0}$,

$$\mathbf{E}^{(n)} = \left.\frac{d^nE}{d\boldsymbol{\mu}^n}\right|_{\mathbf{0}}. \tag{2}$$

Many examples of useful derivatives can be given. For instance, the derivatives with respect to the electric field $\mathbf{F}$ are the permanent electric dipole, the static polarizability, and the static hyperpolarizabilities of various orders

$$\left.\frac{dE}{d\mathbf{F}}\right|_{\mathbf{0}} = -\mathbf{d}, \quad \left.\frac{d^2E}{d\mathbf{F}^2}\right|_{\mathbf{0}} = -\boldsymbol{\alpha}, \quad \left.\frac{d^3E}{d\mathbf{F}^3}\right|_{\mathbf{0}} = -\boldsymbol{\beta}, \ldots, \tag{3}$$

where the subscript denotes differentiation at $\mathbf{F}=\mathbf{0}$. The derivatives with respect to nuclear coordinates $\mathbf{R}$ include the forces on the nuclei and the force constants, while mixed derivatives can provide information such as Raman intensities.[3]

On a classical computer, an energy derivative can be evaluated either numerically or analytically, and we discuss each approach in turn.

[a] Electronic mail: kassal@fas.harvard.edu.
[b] Electronic mail: aspuru@chemistry.harvard.edu.

0021-9606/2009/131(22)/224102/6/$25.00

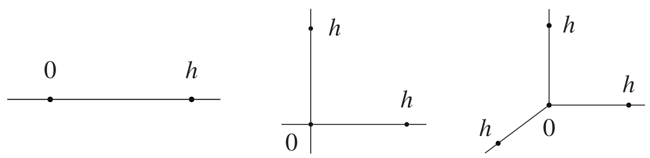

FIG. 1. Obtaining a numerical gradient of a function defined on a d-dimensional space classically requires sampling the function $d+1$ times, once at the origin and once at a distance h along each of the axes. Shown above are the sample points for the cases $d=1$ through $d=3$. The quantum gradient algorithm can evaluate many sample points in superposition, producing the same calculated gradient using only one call to the function.

Numerical derivative techniques rely on computing the value of the energy at several discrete points, and then using those values to estimate the true derivative. The simplest technique is finite difference, which for the first derivative in one dimension is the familiar formula,

$$\left.\frac{dE}{d\mu}\right|_0 \approx \frac{E(h)-E(0)}{h}. \tag{4}$$

In d dimensions, computing the gradient requires at least $d+1$ evaluations of the energy, once at the origin and once at a distance h along each axis (Fig. 1). Similarly, evaluating higher-order derivatives requires the knowledge of the energy on a particular grid, with at least d^n+1 points for the nth derivative.

While numerical gradient techniques usually require minimal effort to implement, they are occasionally susceptible to numerical instability, due to the ill-posedness of numerical differentiation in general.[13] This is particularly problematic when using finite-precision arithmetic, where various rounding errors can accumulate and be amplified upon division by the small number h. The fact that small errors in the evaluated function can lead to large errors in the derivative affects *ab initio* electronic structure methods insofar as they usually involve long calculations with many potential sources of error, including rounding and quadrature.

By contrast, analytical derivative techniques are those that compute the derivative by direct evaluation of an analytical expression. The formulas were derived by Bratož[1] and first implemented in quantum chemistry by Pulay.[2] Analytical techniques have since largely supplanted numerical procedures because they are numerically more stable and, more importantly, usually faster as well.

Analytical gradient formulas exist for most electronic structure techniques and for most kinds of perturbations. To illustrate the argument and establish the correct scaling, we will describe the analytical derivatives of fully variational wave functions (this is not a restriction: wave functions such as coupled-cluster can be made fully variationally by adding a suitable Lagrangian to the energy). We start by writing the molecular energy as a function $E(\boldsymbol{\mu},\boldsymbol{\lambda}(\boldsymbol{\mu}))$ of the external perturbation $\boldsymbol{\mu}$ and the wave function parameters $\boldsymbol{\lambda}(\boldsymbol{\mu})$. These parameters, such as the configuration interaction coefficients, completely describe the electronic wave function. Although $\boldsymbol{\lambda}(\boldsymbol{\mu})$ is a function of $\boldsymbol{\mu}$, for simplicity we will write only $\boldsymbol{\lambda}$. The energy is said to be *fully variational* with respect to $\boldsymbol{\lambda}$ if, for any given $\boldsymbol{\mu}$, $\boldsymbol{\lambda}$ assumes the value $\boldsymbol{\lambda}^*$ such that the variational condition holds

$$\left.\frac{\partial E(\boldsymbol{\mu},\boldsymbol{\lambda})}{\partial\boldsymbol{\lambda}}\right|_* = \mathbf{0}, \tag{5}$$

where $*$ indicates $\boldsymbol{\lambda}=\boldsymbol{\lambda}^*$. In that case we can write $E(\boldsymbol{\mu}) = E(\boldsymbol{\mu},\boldsymbol{\lambda}^*)$.

For fully variational wave functions, the gradient with respect to $\boldsymbol{\mu}$ is given by

$$\begin{aligned}\frac{dE(\boldsymbol{\mu})}{d\boldsymbol{\mu}} &= \left.\frac{\partial E(\boldsymbol{\mu},\boldsymbol{\lambda})}{\partial\boldsymbol{\mu}}\right|_* + \left.\frac{\partial E(\boldsymbol{\mu},\boldsymbol{\lambda})}{\partial\boldsymbol{\lambda}}\right|_* \cdot \frac{\partial\boldsymbol{\lambda}}{\partial\boldsymbol{\mu}} \\ &= \left.\frac{\partial E(\boldsymbol{\mu},\boldsymbol{\lambda})}{\partial\boldsymbol{\mu}}\right|_* \\ &= \langle\boldsymbol{\lambda}^*|\frac{\partial H}{\partial\boldsymbol{\mu}}|\boldsymbol{\lambda}^*\rangle, \end{aligned} \tag{6}$$

where we have used the variational condition and the Hellman–Feynman theorem. Since one need not know the first-order wave function response $\partial\boldsymbol{\lambda}/\partial\boldsymbol{\mu}$, computing the gradient is, to within a small constant factor,[7] as hard as computing the energy. That is, once $|\boldsymbol{\lambda}^*\rangle$ is available, calculating the expectation value of the Hamiltonian has approximately the same computational cost as calculating the expectation value of its derivative. However, computing the second derivative matrix does require the knowledge of the first-order wave function response. In fact, as a direct consequence of Wigner's $2n+1$ rule of perturbation theory, one needs to know the first n wave function responses in order to calculate the $(2n+1)$th derivative. Computing the responses often becomes the bottleneck, and it is what leads to a higher asymptotic cost of higher-order derivatives. While the gradient requires about the same resources as the energy, the second and third derivatives require resources that scale as $O(d)$ times the cost of computing the energy (where d is the number of degrees of freedom, i.e., the dimension of $\boldsymbol{\mu}$).[7] This scaling comes about because $O(d)$ time is required to compute the matrix $\partial\boldsymbol{\lambda}/\partial\boldsymbol{\mu}$. Likewise, the scaling of the $(2n+1)$th derivative is $O(d^n)$, because the bottleneck becomes the computation of the nth order wave function response. In other words, the computational cost of finding the nth analytical derivative is $O(d^{\lfloor n/2\rfloor})$, roughly a quadratic speed-up over the $O(d^n)$ numerical methods of the same degree.

It should be noted that the scaling of derivative techniques, both numerical and analytical, depends on d. For some classes of useful properties, this is not a concern because d is independent of system size. For example, if the perturbation is the electric field, then $d=3$, and indeed there are classical techniques for computing electrical properties of large molecules. However, in cases where d varies with system size, as it does whenever there is differentiation with respect to nuclear coordinates, larger systems become increasingly difficult to treat. This is most acutely true of the molecular Hessian, which is often beyond reach, even though the gradient is routinely accessible. We now show that if quantum computers were available, the cost of the higher derivatives would no longer be prohibitive.

TABLE I. Time resources required by various techniques of computing molecular properties, in terms of the cost of computing the energy. For example, the entry $d+1$ means that computing the property requires $d+1$ evaluations of the molecular energy, while the entries in the "Analytical" column indicate comparable computational effort. E is the total electronic energy, $\boldsymbol{\mu}$ is the external perturbation, and d is the dimension of $\boldsymbol{\mu}$. All the derivatives are evaluated at $\boldsymbol{\mu}=\mathbf{0}$. On classical computers, the numerical scalings correspond to the simplest finite-difference scheme. Analytical techniques are the ones that evaluate the derivative directly (the exponent $\lfloor n/2 \rfloor$ comes from Wigner's $2n+1$ rule). On a quantum computer, the quantum gradient estimation algorithm is used. It should be noted that on a quantum computer, the number of evaluations of E needed for any derivative is independent of d, and thus of the size of the system.

	Classical		Quantum
Derivative	Numerical	Analytical	Numerical
$\frac{dE}{d\boldsymbol{\mu}}$	$d+1$	$O(1)$	1
$\frac{d^2E}{d\boldsymbol{\mu}^2}$	d^2+1	$O(d)$	2
$\frac{d^3E}{d\boldsymbol{\mu}^3}$	d^3+1	$O(d)$	4
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\frac{d^nE}{d\boldsymbol{\mu}^n}$	d^n+1	$O(d^{\lfloor n/2 \rfloor})$	2^{n-1}

II. THE QUANTUM ALGORITHM

The quantum algorithm for molecular properties is based on Jordan's quantum gradient estimation algorithm.[14] Jordan's method can numerically compute the gradient of any function F, given a black box (oracle) that computes the value of F for an arbitrary input. In particular, the algorithm can evaluate the gradient using a *single* query to F, regardless of the number of dimensions d of the domain of F. By contrast, the simplest classical finite-difference scheme would require $d+1$ queries to F (see Fig. 1). The speed-up is essentially achieved by being able to sample along all the d dimensions in superposition. We apply Jordan's algorithm to the computation of molecular properties by specifying a way to compute the energy on a quantum computer as well as by outlining how to obtain higher derivatives. In this section, we describe the algorithm, its application to quantum chemistry, and finally argue that a return to numerical techniques for molecular properties would be justified if quantum computers became feasible.

We assume that the molecular energy is a smooth, bounded function of the perturbation, $E:[-h/2,h/2]^d \rightarrow [E_{\min},E_{\max}]$, where a small h is chosen so that E varies sufficiently slowly over the domain. For convenience, we express the perturbations in units such that h is unitless and such that the bounds are the same along all of the axes. We also assume that we have a black box for E, which, given a perturbation $\boldsymbol{\mu}$, outputs the energy $E(\boldsymbol{\mu})$. The precise nature of the algorithm inside the black box is not important, so long as it can be implemented on a quantum computer. In particular, any classical technique of electronic-structure theory can be converted into a quantum algorithm.[15] In Sec. III, we will discuss possible realizations of the black box, including the use of quantum simulation algorithms, which offer a significant improvement over classical ones.

We begin by choosing the number n of bits of precision that we desire in the final gradient. Jordan's algorithm starts in an equal superposition on d registers of n qubits each (nd qubits total) (Ref. 15),

$$\frac{1}{\sqrt{N^d}}\sum_{k_1=0}^{N-1}\cdots\sum_{k_d=0}^{N-1}|k_1\rangle\cdots|k_d\rangle = \frac{1}{\sqrt{N^d}}\sum_{\mathbf{k}}|\mathbf{k}\rangle, \tag{7}$$

where $N=2^n$, the states $|k_i\rangle$ are integers on n qubits represented in binary, and $|\mathbf{k}\rangle$ is a d-dimensional vector of all the $|k\rangle$'s.

We use this state as an input for the black box for E, which will, for every integer-vector $|\mathbf{k}\rangle$ in the superposition, append a phase proportional to the energy $E(\boldsymbol{\mu})$ at perturbation $\boldsymbol{\mu}=h(\mathbf{k}-\mathbf{N}/2)/N$ (where $\mathbf{N}$ is the vector $(N,N,\ldots,N)$ and serves to center the sampling region on the origin). To achieve maximum precision with fewest qubits, one needs an estimate m of the largest magnitude of any of the first derivatives of E. Then, the energy evaluated by the black box is scaled by a factor $2\pi N/hm$. Because the black box operates on all the terms in the superposition at once, a single call results in the state

$$\frac{1}{\sqrt{N^d}}\sum_{\mathbf{k}}\exp\left[\frac{2\pi i N}{hm}E\left(\frac{h}{N}(\mathbf{k}-\mathbf{N}/2)\right)\right]|\mathbf{k}\rangle$$
$$\approx \frac{1}{\sqrt{N^d}}\sum_{\mathbf{k}}\exp\left[\frac{2\pi i N}{hm}\left(E(\mathbf{0})+\frac{h}{N}(\mathbf{k}-\mathbf{N}/2)\cdot\left.\frac{dE}{d\boldsymbol{\mu}}\right|_{\mathbf{0}}\right)\right]|\mathbf{k}\rangle. \tag{8}$$

The neglect of terms quadratic in h and higher is a valid approximation for sufficiently small h (the error caused by higher-order terms is discussed in Ref. 14 and is only polynomial). The final state is separable and equals

$$\frac{e^{i\Phi(\mathbf{0})}}{\sqrt{N^d}}\sum_{k_1=0}^{N-1}\exp\left[\frac{2\pi i}{m}k_1\left.\frac{\partial E}{\partial\mu_1}\right|_{\mathbf{0}}\right]|k_1\rangle\cdots$$
$$\sum_{k_d=0}^{N-1}\exp\left[\frac{2\pi i}{m}k_d\left.\frac{\partial E}{\partial\mu_d}\right|_{\mathbf{0}}\right]|k_d\rangle, \tag{9}$$

with phase

$$\Phi(\mathbf{0}) = 2\pi\left(\frac{N}{hm}E(\mathbf{0})-\frac{\mathbf{N}}{2m}\cdot\left.\frac{dE}{d\boldsymbol{\mu}}\right|_{\mathbf{0}}\right). \tag{10}$$

Applying the inverse quantum Fourier transform[15] to each of the d registers results in the gradient

$$e^{i\Phi(\mathbf{0})}\left|\left.\frac{N}{m}\frac{\partial E}{\partial\mu_1}\right|_{\mathbf{0}}\right\rangle\cdots\left|\left.\frac{N}{m}\frac{\partial E}{\partial\mu_d}\right|_{\mathbf{0}}\right\rangle$$
$$= e^{i\Phi(\mathbf{0})}\left|\left.\frac{N}{m}\frac{dE}{d\boldsymbol{\mu}}\right|_{\mathbf{0}}\right\rangle. \tag{11}$$

The scaling factor (N/m) ensures that $N/m(dE/d\boldsymbol{\mu})$ is an integer-vector with n bits of precision along each dimension. It should be reiterated that a single call to E was made, as opposed to the $d+1$ that would be needed in the case of numerical differentiation by finite difference.

Overall, the gradient estimation algorithm produces the transformation

$$|\mathbf{0}\rangle \rightarrow e^{i\Phi(\mathbf{0})}\left| \frac{N}{m}\frac{dE}{d\boldsymbol{\mu}}\bigg|_{\mathbf{0}}\right\rangle. \tag{12}$$

We can compute the Hessian (and higher derivatives) by iterating this algorithm. If, instead of making a call to $E(\boldsymbol{\mu})$, the algorithm sought $E(\boldsymbol{\mu}-\boldsymbol{\nu})$ from the black box, we would perform, at the cost of this single additional subtraction,

$$|\mathbf{0}\rangle \rightarrow e^{i\Phi(\boldsymbol{\nu})}\left| \frac{N}{m}\frac{dE}{d\boldsymbol{\mu}}\bigg|_{\boldsymbol{\nu}}\right\rangle, \tag{13}$$

with global phase

$$\Phi(\boldsymbol{\nu}) = 2\pi\left(\frac{N}{hm}E(\boldsymbol{\nu}) - \frac{\mathbf{N}}{2m}\cdot\frac{dE}{d\boldsymbol{\mu}}\bigg|_{\boldsymbol{\nu}}\right). \tag{14}$$

Because this procedure will be used as a subroutine, the global phase will become a relative phase. To avoid undesired interference, it is important to remove (or uncompute) the phase; one additional evaluation of E (at $\boldsymbol{\nu}$) suffices for this uncomputation. Overall, this supplies another black box, which, given $\boldsymbol{\nu}$, yields $|(N/m)(dE/d\boldsymbol{\mu})|_{\boldsymbol{\nu}}\rangle$ using only two calls to the original black box for E. One can use the gradient algorithm with this new black box, producing the state

$$e^{i\Phi^{(2)}(\mathbf{0})}\left| \frac{N}{m^{(2)}}\frac{d^2E}{d\boldsymbol{\mu}^2}\bigg|_{\mathbf{0}}\right\rangle, \tag{15}$$

which is a two-dimensional array of d^2 quantum registers containing all the elements of the Hessian matrix of E. In addition, $m^{(2)}$ is an estimate for the magnitude of the largest second derivative, and the phase is

$$\Phi^{(2)}(\mathbf{0}) = 2\pi\left(\frac{N}{hm^{(2)}}\frac{dE}{d\boldsymbol{\mu}}\bigg|_{\mathbf{0}} - \frac{\mathbf{N}}{2m^{(2)}}\cdot\frac{d^2E}{d\boldsymbol{\mu}^2}\bigg|_{\mathbf{0}}\right). \tag{16}$$

Computing higher derivatives would require additional factors of two in the number of required black box calls, caused by the need to uncompute a global phase at each step (this problem is common when it comes to recursing quantum algorithms[16]). Hence, evaluating the nth derivative requires 2^{n-1} queries to E, which, although exponential in n, is much better than d^n+1, which is the minimum number of function queries required to compute the derivative by classical finite difference. We stress that the number of calls to E is independent of d, and thus of the size of the system, for the derivative of any order.

The quantum gradient algorithm is a numerical approach and therefore, just like classical numerical techniques, it could be affected by numerical instability. This implies that the gradient algorithm cannot be used indiscriminately for problems that feature errors that cannot be controllably reduced through additional computational effort. For example, finite-precision arithmetic presents the same problems to quantum computers as it does to classical ones, but the rounding errors can be brought under control by using more digits of precision (as on classical computers). Quantum chemistry techniques might present numerical problems as well, insofar as they contain uncontrolled sources of error. However, if the technique for computing the energy is numerically exact, that is to say, if the error in the energy can be controllably reduced below any level, the magnitude of the numerical error in the calculated derivative can likewise be made arbitrarily small. Fortunately, quantum computers would make it possible in principle to evaluate the numerically exact molecular energy, meaning that numerical instability would not be a problem. We turn to the topic of molecular energy evaluation next.

III. THE BLACK BOX FOR THE ENERGY

The application of Jordan's gradient algorithm to chemical problems requires a black box that can compute the value of the ground-state molecular energy at any value of the perturbation $\boldsymbol{\mu}$ in the neighborhood of $\boldsymbol{\mu}=\mathbf{0}$. Furthermore, to avoid numerical artifacts, this black box should be numerically exact, allowing the error in the energy to be controllably reduced through additional computational work.

We stress that any electronic-structure method could be used inside this black box. In particular, any one of the many well-developed classical electronic-structure techniques could be implemented on a quantum computer, allowing one to take advantage of the decreased query complexity. However, to take full advantage of the quantum computer, the energy black box could also be a quantum algorithm. This could potentially avoid the steep computational cost of exact classical electronic-structure methods, which scale exponentially with the size of the system. In the remainder of this section, we give some details of what a quantum black box might look like, although, again, the precise details of the black box do not matter for the rest of the paper.

Finding the ground-state energy on a quantum computer can be divided into two steps: preparing the ground state (encoded in some way), and then measuring its energy.

The computational cost of the first step, preparing the ground state of chemically relevant systems, is not well known. For general physical systems, and frustrated spin Hamiltonians in particular, finding the ground state is QMA-complete[17] and therefore widely believed to require superpolynomial time, even on a quantum computer. All known algorithms for the ground-state energy of general Hamiltonians require exponential time, . However, numerical evidence suggests that it might be possible to prepare ground states of certain chemically relevant systems in polynomial time using adiabatic state preparation.[9]

The second step, energy measurement, is relatively easy, and can be accomplished on a quantum computer using the phase estimation algorithm.[15,18,19] Its cost depends on the way the wave function is encoded, but is generally polynomial in the size of the system, at a given accuracy. Chemically relevant wave functions can be encoded in two ways: second or first quantization.

In a second-quantized approach, the quantum computer stores the occupation-number vectors in a chosen basis set,[9] meaning that the black box is essentially a quantum version of full configuration interaction (FCI). In this case, phase estimation for determining the energy to a given precision requires $O(M^5)$ time,[20] where M is the number of basis functions.

An alternative is a first-quantized representation, in which wave functions are encoded on grid in real space.[11,21–23] Performing phase estimation in this case requires simulating the dynamics of the relevant chemical system. For P particles interacting under a pairwise interaction, the exact dynamics can be simulated in $O(P^2)$ time and $O(P)$ space, in contrast to the classical exponential cost. Although the large prefactor of this algorithm would make it slower for small molecules than the equivalent quantum FCI calculation, it benefits from a superior asymptotic scaling as well as from the fact that only minimal modifications would need to be made to insert the perturbation $\boldsymbol{\mu}$ into the calculation. For example, simulations with different nuclear coordinates proceed in exactly the same way, while an electromagnetic field requires only a small modification of the simulated Hamiltonian.[21]

Phase estimation, the crucial ingredient of these algorithms, has been criticized as inefficient[24] because its cost grows exponentially with the number of bits of precision sought. This could be significant for gradient estimation, which might require precise energy evaluations to avoid numerical errors. We note that if the gradient is desired to n bits of precision [as in Eq. (11)], the black box needs to evaluate the energy to

$$n_E = \log_2\left[\frac{E_{\max}-E_{\min}}{(mh/2^n)(\theta/2\pi)}\right] \approx n + \log_2\frac{2\pi}{\theta}, \qquad (17)$$

bits of precision,[14] where $\cos^2\theta$ is the desired success probability of the algorithm. For example, with $\theta=\pi/8$, the algorithm succeeds 85% of the time and requires four more digits of precision in the energy than is desired in the gradient. The four additional digits present only a constant additional overhead, meaning that the computation of any molecular property at any precision is, up to a constant factor, as hard as computing the energy of the same molecule at the same precision.

IV. NEWTON'S METHOD AND GEOMETRY OPTIMIZATION

Perhaps the single most common use of molecular derivatives is molecular geometry optimization. We can therefore use it to illustrate some of the advantages of a quantum algorithm over a classical one, including a quantum version of Newton's method, which offers an additional quadratic speedup over its classical counterpart.

A simple way for finding the locally optimal geometry is to perform the standard Newton iterations,

$$\mathbf{R}_{n+1} = \mathbf{R}_n - \left(\left.\frac{dE}{d\mathbf{R}}\right|_{\mathbf{R}_n}\right)\cdot\left(\left.\frac{d^2E}{d\mathbf{R}^2}\right|_{\mathbf{R}_n}\right)^{-1}, \qquad (18)$$

until convergence is reached. Here, $\mathbf{R}_n$ are the nuclear coordinates at the n^{th} iteration, and $dE/d\mathbf{R}|_{\mathbf{R}_n}$ and $d^2E/d\mathbf{R}^2|_{\mathbf{R}_n}$ are, respectively, the gradient and Hessian of E with respect to nuclear displacement (the "molecular gradient" and the "molecular Hessian"). If a quantum computer were used to compute the derivatives, one would require exactly three calls to the energy black box per iteration: one for the gradient and two for the Hessian. A classical approach, on the other hand, would be much slower, requiring at least d^2+1 function calls for finite difference, and approximately $O(d)$ effort in the analytical case. For a molecule with N nuclei, $d=3N$, which means that this improvement in query complexity would be most significant in the case of large molecules.

We do not consider here quasi-Newton methods or simpler methods such as gradient descent, which, classically, offer the advantage of not needing to compute the molecular Hessian at each step, or at all. While such schemes are useful and widely applied, they are typically slower or, for a given number of steps, less accurate than Newton's method. On a quantum computer, their classical computational savings would be superceded by the ability to rapidly compute the exact Hessian at each step.

There are many classical tricks available for speeding up the convergence of Newton's method if the initial guess is not close to a local minimum, in which case the usual Newton step might be inappropriately large. Techniques such as trust regions and level shifts[25] are still available to quantum computers, or they can be implemented as classical postprocessing.

In addition, we remark that Newton's method is the first in the class of Householder methods, which offer a rate of convergence of $\ell+1$, provided that derivatives up to order $\ell+1$ exist and can be calculated. A quantum computer could be used to accelerate Householder methods of any degree, requiring $\sum_{m=1}^{\ell+1}2^{m-1}=2^{\ell+1}-1$ calls to the black box for order-ℓ Householder optimization method. Although exponential in ℓ, this expression is independent of system dimension d.

Of course, Newton's method is only useful for local minimization, and we are often interested in global optimization. Here, we can use a quantum version of the multistart technique, called the quantum basin hopper.[26–29] A number of points is selected at random, and each is followed, using a local search, to its local basin (if a quantum version of Newton's method is used for the local search, such as the one we propose above, we can get the usual quadratic convergence). Then, the minima of all the basins are compared and the least one chosen as the global minimum. Quantum computers could add a quadratic speed-up to such a multistart technique, since the resulting local minima form an unstructured database that can be searched using Grover's algorithm[12,15] with a quadratic speed-up. As Dürr and Høyer pointed out,[30] a Grover search can find the minimum of an unstructured database with $O(\sqrt{K}\log K)$ calls to the database (where K is the number of database entries, i.e., multistart points), as opposed to the classically required $O(K\log K)$ queries.

V. CONCLUSION

We have shown that Jordan's quantum gradient estimation algorithm can be applied to the estimation of time-independent, non-relativistic molecular properties. Doing so requires a quantum electronic-structure black box, for which known quantum simulation algorithms are well suited. The quantum algorithm offers a speed-up from the classical cost of $O(d^{\lfloor n/2 \rfloor})$ for analytical derivatives to the quantum query complexity of 2^{n-1} (Table I). That is, the number of energy calculations required on the quantum computer is independent of d, and thus of the system size, which could offer a significant advantage for the computation of properties of large systems. In particular, it would make the molecular Hessian of any molecule only twice as expensive as its molecular gradient, enabling a fast, local geometry optimization using Newton's method. Finally, global optimization could combine the local Newton's method with Grover search to offer an additional quadratic speed-up over the classical multi-start technique.

An avenue for future work is investigating whether there are quantum versions of analytical gradient techniques. It is possible that such algorithms could outperform the numerical methods outlined here, for example by not requiring the computationally expensive recursion.

ACKNOWLEDGMENTS

We acknowledge support from the Army Research Office under Contract No. W911NF-07-0304. I.K. thanks the Joyce and Zlatko Baloković Scholarship.

[1] S. Bratož, Colloq. Int. C. N. R. S. **82**, 287 (1958).
[2] P. Pulay, Mol. Phys. **17**, 197 (1969).
[3] P. Pulay, Adv. Chem. Phys. **69**, 241 (1987).
[4] T. Helgaker and P. Jørgensen, Adv. Quantum Chem. **19**, 183 (1988).
[5] Y. Yamaguchi, J. D. Goddard, Y. Osamura, and H. Schaefer, *A New Dimension to Quantum Chemistry: Analytic Derivative Methods in Ab Initio Molecular Electronic Structure Theory* (Oxford University Press, New York, 1994).
[6] R. Shepard, in *Modern Electronic Structure Theory*, edited by D. Yarkony (World Scientific, Singapore, 1995), pp. 345–458.
[7] P. Pulay, in *Modern Electronic Structure Theory*, edited by D. Yarkony (World Scientific, Singapore, 1995), pp. 1191–1240.
[8] T. Helgaker, in *The Encyclopedia of Computational Chemistry*, edited by P. v. R. Schleyer, N. L. Allinger, T. Clark, J. Gasteiger, P. A. Kollman, H. F. Schaefer III, and P. R. Schreiner (Wiley, New York, 1998), pp. 1157–1169.
[9] A. Aspuru-Guzik, A. D. Dutoi, P. J. Love, and M. Head-Gordon, Science **309**, 1704 (2005).
[10] H. Wang, S. Kais, A. Aspuru-Guzik, and M. R. Hoffmann, Phys. Chem. Chem. Phys. **10**, 5388 (2008).
[11] I. Kassal, S. P. Jordan, P. J. Love, M. Mohseni, and A. Aspuru-Guzik, Proc. Natl. Acad. Sci. U.S.A. **105**, 18681 (2008).
[12] L. Grover, Proceedings of the 28th Annual ACM Symposium on the Theory of Computing, 1996 (unpublished), pp. 212–219.
[13] A. N. Tikhonov and V. Y. Arsenin, *Solutions of Ill-Posed Problems* (Winston, Washington, 1977).
[14] S. P. Jordan, Phys. Rev. Lett. **95**, 050501 (2005).
[15] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, New York, 2000).
[16] S. Aaronson, Quantum Inf. Comput. **3**, 165 (2003).
[17] J. Kempe, A. Kitaev, and O. Regev, SIAM J. Comput. **35**, 1070 (2006).
[18] A. Y. Kitaev, e-print arXiv:quant-ph/9511026.
[19] D. S. Abrams and S. Lloyd, Phys. Rev. Lett. **83**, 5162 (1999).
[20] B. P. Lanyon, J. D. Whitfield, G. G. Gillet, M. E. Goggin, M. P. Almeida, I. Kassal, J. D. Biamonte, M. Mohseni, B. J. Powell, M. Barbieri, A. Aspuru-Guzik, and A. G. White, "Towards Quantum Chemistry on a Quantum Computer," Nature Chem. (in press).
[21] C. Zalka, Proc. R. Soc. London, Ser. A **454**, 313 (1998).
[22] D. A. Lidar and H. Wang, Phys. Rev. E **59**, 2429 (1999).
[23] N. J. Ward, I. Kassal, and A. Aspuru-Guzik, J. Chem. Phys. **130**, 194105 (2009).
[24] K. R. Brown, R. J. Clark, and I. L. Chuang, Phys. Rev. Lett. **97**, 050504 (2006).
[25] R. Fletcher, *Practical Methods of Optimization*, 2nd ed. (Wiley, New York, 2000).
[26] D. Bulger, e-print arXiv:quant-ph/0507193.
[27] D. Bulger, J. Optim. Theory Appl. **133**, 289 (2007).
[28] W. P. Baritompa, D. Bulger, and G. Wood, SIAM J. Optim. **15**, 1170 (2005).
[29] J. Zhu, Z. Huang, and S. Kais, e-print arXiv:0906.2375.
[30] C. Dürr and P. Høyer, e-print arXiv:quant-ph/9607014.