Quantum Algorithm Providing Exponential Speed Increase for Finding Eigenvalues and Eigenvectors

Daniel S. Abrams*
Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California 91109-8099

Seth Lloyd†
d'Arbeloff Laboratory for Information Sciences and Technology, Department of Mechanical Engineering, MIT 3-160, Cambridge, Massachusetts 02139
(Received 27 July 1998)

We describe a new polynomial time quantum algorithm that uses the quantum fast Fourier transform to find eigenvalues and eigenvectors of a local Hamiltonian, and that can be applied in cases (commonly found in *ab initio* physics and chemistry problems) for which all known classical algorithms require exponential time. Applications of the algorithm to specific problems are considered, and we find that classically intractable and interesting problems from atomic physics may be solved with between 50 and 100 quantum bits.

PACS numbers: 03.67.Lx, 31.15.Ar

Long before Shor's ground-breaking algorithm [1]—and the resulting surge of interest in quantum computing—Feynman suggested that a quantum computer might be useful for simulating other quantum systems [2]. This suggestion was based upon the observation that quantum systems are described in a Hilbert space whose size grows exponentially with the number of particles. Thus a collection of only 100 spin-$\frac{1}{2}$ particles, each of which could be specified by only two complex amplitudes were it isolated, requires a total of 2^{100} complex amplitudes for its state to be specified completely. This exponential explosion severely limits our ability to perform true "*ab initio*" (first principles) calculations; since it is obviously not possible to even describe the state of anything but the smallest quantum systems, one must resort to various approximation techniques to calculate properties of interest.

Recent work in quantum computation has revealed various techniques for *simulating* physics on a quantum computer [3–8], and it has been demonstrated that this can, in fact, be accomplished efficiently, as Feynman supposed. However, there has been comparatively little work done on algorithms which *calculate static properties* of a physical system [8]. In particular, of all the questions which one might ask about a quantum system, there is one most frequently asked and for which one would most greatly desire an efficient algorithm: What are the energy eigenvalues and eigenstates? In this Letter, we provide a quantum algorithm that can find eigenvalues and eigenvectors of a Hamiltonian operator in cases that occur frequently in problems of physical interest. Moreover, the algorithm requires an amount of time which scales as a polynomial function of the number of particles and the desired accuracy, whereas all classical algorithms (with known complexity) require an exponential amount of time.

The problem to be solved can be precisely stated as follows. Consider the time-evolution operator $\hat{U} = e^{-(i/\hbar)\hat{H}t}$ which corresponds to the Hamiltonian $\hat{H}$, and an approximate eigenvector V_a of $\hat{U}$ (and thus of $\hat{H}$) that can be generated in quantum polynomial time; i.e., the machine can be placed into a state corresponding to V_a using a polynomial number of quantum logic operations. Call the true eigenvector V and the true eigenvalue λ_v. If the state V_a satisfies the property that $|\langle V_a \,|\, V\rangle|^2$ is not exponentially small—that is, the approximate eigenvector contains a component of the actual eigenvector that is bounded by a polynomial function of the problem size—then λ_v can be found to accuracy ϵ in time proportional to $1/|\langle V_a \,|\, V\rangle|^2$ and $1/\epsilon$. Moreover, if the eigenvalue λ_v is nondegenerate, the algorithm will also reveal the eigenvector V with polynomial accuracy. (Eigenvectors can also be found in the degenerate case, but this is slightly more complicated and will be discussed below.)

Intuitively, what the algorithm does is resolve the guess into its non-negligible components and determine the corresponding eigenvalues. If the operator $\hat{U}$ (and thus its eigenvectors) is of exponentially large dimension—which it typically is—there are no known classical algorithms that can find even the eigenvalues in polynomial time. Although the requirement that there exist an initial state vector V_a with the specified properties may appear to be overly restrictive, it is frequently (if not usually) possible to obtain such a guess for "real" problems using existing classical techniques. For example, in any physical system with discrete energy levels that are not exponentially close together near the ground state (such as an atom), if it is possible to obtain classically any state vector with expected energy merely less than the first excited state (by a nonexponentially small amount), then this state vector must contain a non-negligible component of the ground state and—although it may not remotely resemble the ground state—could be used as the approximate state V_a to determine the true ground state and ground state energy

in polynomial time. Finally, if for some problems it is not possible to obtain classically a guess with the desired properties, it may often be the case that the state vector V_a may be generated using a quantum algorithm, such as quantum simulated annealing.

We begin by describing a subroutine which can be applied to any $\hat{U}$ that can be implemented in quantum polynomial time. (It was shown in [3] that this includes the time evolution operator corresponding to any local Hamiltonian.) A similar subroutine was previously (though independently) described by Kitaev in [9] and refined by Cleve *et al.* in [10]. They show how one can obtain an eigenvalue with exponential precision if one is initially given the eigenvector and devices that can perform $\hat{U}$, $\hat{U}^2$, $\hat{U}^4, \ldots,$ and $\hat{U}^{2^m}$; they then use this subroutine for factoring (in a modified version of Shor's algorithm) by randomly sampling eigenvalues from the eigenspectrum. We employ essentially the same subroutine as Cleve *et al.* but in a different context (a physical simulation) and with a different approach, beginning with an estimate of an eigenvector, an ability to perform $\hat{U}$ only (that is, we can perform $\hat{U}$ but not $\hat{U}^2$, $\hat{U}^4, \ldots,$ and $\hat{U}^{2^m}$), and then determining both the eigenstate and eigenvalue to polynomial precision.

Consider a quantum computer consisting of $m + l + w$ qubits, where a total of m qubits (to be called the index bits) are used for a fast Fourier transform (FFT), a total of l qubits describe the Hilbert space in which the operator $\hat{U}$ acts, and w extra working qubits are required for temporary storage. Let $M = 2^m$. The accuracy of the result will grow as $1/M$—therefore, the required number of qubits will scale as the log of the accuracy. Assume that the m index qubits are initially in the state $|0\rangle$ and that the l qubits are initially in the state V_a; i.e., the initial state is

$$|\Psi\rangle = |0\rangle|V_a\rangle, \tag{1}$$

where the w work qubits are assumed to be $|0\rangle$ unless specified otherwise. We perform a $\pi/2$ rotation on each of the m index qubits to obtain the state

$$|\Psi\rangle = \frac{1}{\sqrt{M}} \sum_{j=0}^{M-1} |j\rangle|V_a\rangle. \tag{2}$$

Next, one performs a series of quantum logic operations that transform the computer into the state

$$|\Psi\rangle = \frac{1}{\sqrt{M}} \sum_{j=0}^{M-1} |j\rangle(\hat{U})^j|V_a\rangle. \tag{3}$$

This transformation is accomplished by applying the operation $\hat{U}$ to the second set of l qubits (which are initially in the state $|V_a\rangle$) a total of j times. It can be implemented easily by performing a loop (indexed by i) from 1 to M. Using standard quantum logic operations, set a flag qubit to the value $|1\rangle$ if and only if $i < j$ and perform the operation $\hat{U}$ conditioned on the value of this flag. Thus only those components of the above superposition for which $i < j$ are affected. Finally, undo the flag qubit and continue with the next iteration. After M iterations, the state above is obtained.

At this point, it is helpful to rewrite the state in a slightly different manner. Label the eigenvectors of $\hat{U}$ by the states $|\phi_k\rangle$ and the corresponding eigenvalues with λ_k. We can then write

$$|V_a\rangle = \sum_k c_k|\phi_k\rangle \tag{4}$$

in which case the state (3) above can be rewritten as

$$|\Psi\rangle = \frac{1}{\sqrt{M}} \sum_{j=0}^{M-1} |j\rangle(\hat{U})^j \sum_k c_k|\phi_k\rangle \tag{5}$$

$$= \frac{1}{\sqrt{M}} \sum_k c_k \sum_{j=0}^{M-1} |j\rangle(\lambda_k)^j|\phi_k\rangle. \tag{6}$$

If we write λ_k as $e^{i\omega_k}$ and exchange the order of the qubits so that the labels $|\phi_k\rangle$ appear first, the result is seen then most clearly:

$$|\Psi\rangle = \frac{1}{\sqrt{M}} \sum_k c_k|\phi_k\rangle \sum_{j=0}^{M-1} e^{i\omega_k j}|j\rangle. \tag{7}$$

It is now self-evident that a quantum FFT performed on the m index qubits will reveal the phases ω_k and thereby the eigenvalues λ_k. The quantum FFT requires only poly(m) operations, whereas the accuracy of the resulting eigenvalue will scale linearly with M or 2^m. Each frequency is seen to occur with amplitude $c_k = \langle V_a|\phi_k\rangle$; by performing a measurement on the m index qubits, one thus obtains each eigenvalue with probability $|c_k|^2$. Only a polynomial number of trials is therefore required to obtain any eigenvalue for which c_k is not exponentially small. If the initial guess $|V_a\rangle$ is close to the desired state (i.e., $|\langle V_a|V\rangle|^2$ is nearly 1), then only a few trials may be necessary.

Moreover, once a measurement is made and an eigenvalue λ_k is determined, the remaining l qubits "collapse" into the state of the corresponding eigenvector. One is likely to be interested in various properties of the eigenvectors, and these can be determined by making various measurements on the state. For *ab initio* quantum calculations, easily obtainable properties include those of greatest interest: charge density distributions, correlation functions, momentum distributions, etc. Of course, the state $|\phi_k\rangle$ is still in some sense "trapped" inside the computer. But since it is impossible to store as classical information the 2^l phases associated with the state, one cannot possibly do better. See [4] for a discussion of how relevant physical information can be extracted efficiently from the quantum computer.

An interesting subtlety occurs if the eigenvalue found above is degenerate or nearly degenerate, by which we

mean that there are several eigenvalues which differ by less than the accuracy $1/M$. (Note, however, that nearly degenerate states can be resolved in polynomial time, if desired, as long as they are not exponentially close together.) For degenerate or nearly degenerate eigenvalues, the measurement projects the system into the corresponding subspace. One can then determine properties of this subspace—that is, the relevant physical properties of the system—through additional measurements as described above. Furthermore, one can also use this technique to detect the presence of a degeneracy by simulating a small perturbation or by varying the initial conditions.

We now consider more precisely how we use this subroutine to find the eigenvectors and eigenvalues of a real Hamiltonian. Generally, one wishes to find energy eigenstates for a Hamiltonian of the form

$$H = \sum_{i=1}^{n} (T_i + V_i) + \sum_{i>j}^{n} V_{ij}, \tag{8}$$

where n is the number of particles, T_i is the kinetic energy, V_i is the external potential, and V_{ij} is the interaction between the particles. (Other terms can be included, as long as they operate on only a few particles at a time.) The time evolution operator is generated using the technique described in [3]; the key idea is to write $H = \sum H_i$ (where each H_i acts on only k qubits at a time) and

$$\hat{U}(t) = e^{-iHt} = (e^{-iH_1(t/m)} e^{-iH_2(t/m)} \cdots e^{-iH_k(t/m)})^m + \sum_{i<j} [H_i, H_j] \frac{t^2}{2m} + \ldots . \tag{9}$$

Let $U_i = e^{-iH_i(t/m)}$. Each term U_i can be implemented efficiently, because it acts in a space of only k quantum bits, where k is small. For large enough m, the second term on the right (and the higher order terms) approaches zero. It is therefore possible to generate $\hat{U}(t)$ by acting on the state with each U_i in series, a total of m times. In order to simulate $\hat{U}(t)$ with an accuracy ϵ, one needs to apply $\mathrm{O}(t^2/\epsilon)$ quantum logic operations [11].

For a specific problem, the form of the matrices U_i depends greatly on the basis set chosen to describe the Hilbert space. Moreover, the choice may strongly impact the size of the basis required to describe the system accurately. Virtually any basis set may be used: position space, momentum space, wavelets, single electron solutions for an effective potential, etc. As long as the single particle basis is of a fixed size, then the operators U_i can always be calculated in the chosen basis and implemented using $O(d^4)$ operations, where d is the dimension of the *single particle* basis set [12]. On the other hand, there is a tradeoff between memory and speed. By using the position or momentum space representation, one needs only $O(\mathrm{poly}(k)) = O(\mathrm{poly}(\log d))$ operations to perform each U_i; however, a large number of qubits are required to describe the eigenstates accurately. By choosing a more elaborate basis set, one can vastly reduce the required number of qubits, but a much larger number of quantum logic operations $O(d^4)$ may be necessary to implement each U_i. Thus one finds that, just as with conventional computations, the choice of basis sets in the quantum computation will depend upon the specific problem at hand and the specific capabilities of the actual computing machine.

Normally, the initial state V_a will be the result of a classical calculation, for example, a Hartree-Fock calculation or configuration interaction calculation. Any *ab initio* technique which results in a known wave function can be used. (Note that this does not include those techniques based on density functional theory, as we require a wave function, not simply a charge density distribution.) If the input wave function is not already symmetrized or antisymmetrized, we can use the algorithms described in [4] to do so efficiently.

Finally, we consider state-of-the-art *ab initio* calculations of atomic energy levels in order to compare the quantum algorithm described above with known classical techniques. Problems from atomic physics serve as a particularly good benchmark because extremely accurate experimental data are widely available. The quantum algorithm corresponds most closely to what is known as "complete active configuration interaction" or "full configuration interaction" techniques, because the many-particle basis set includes all possible products of single particle basis vectors. This approach is most valuable in situations where the correlation energy is large and where many "configurations" are of similar energy (this typically occurs when many electrons are in open shells). Unfortunately, it is difficult to state precisely the minimum size problem for which the quantum calculation surpasses the best classical calculations, because a variety of sophisticated techniques are used to avoid the exponential explosion in basis states. That is, the most *accurate* classical calculations do not employ directly the full configuration interaction method. Based on [13], however, we estimate that a calculation of the energy levels of B (five electrons), using roughly 20 angular wave functions and 40 radial wave functions per particle—for a total of 800 single particle wave functions and therefore $800^5 \approx 10^{15}$ full many-body basis states—may provide more accurate results than any classical calculation performed to date. At the very least, such a calculation would reveal scientifically interesting (and classically unobtainable) results with respect to electron correlation energies in B and the relative importance of various orders of excited configurations.

A quantum calculation of the B ground state, using a basis set as described above, can be accomplished with 60 qubits: 10 per particle to represent the state of the atom (for a total of 50 qubits), 6 or 7 qubits for the FFT, and a few additional "scratch" qubits. Unfortunately, the two particle operators (generated by the Coulomb attraction between pairs of electrons) take place in a subspace of dimension $(2^{10})^2$; they therefore are represented by matrices

with 2^{40} elements. While implementing such an operator by brute force is likely to remain intractable for the foreseeable future, it is possible to perform the necessary transformation using a quantum algorithm. One possible technique is to temporarily change basis sets for pairs of particles while calculating their Coulomb interaction; in position space, the matrices are diagonal and easy to calculate. This method will require an additional 40 qubits for temporary work space, and it follows that in order to realistically perform an "interesting" calculation using the algorithms described previously, one may possibly require a quantum computer with as many as 100 qubits. Of course, a more efficient quantum algorithm for implementing the Coulomb interaction for a specific basis might not require as many additional qubits [14].

In a real implementation, one will need to cope with errors of two forms: those that effect the FFT (and the correlations between the index qubits and the wave function qubits) and those that effect the time evolution of the simulated Hamiltonian. The first type will slightly perturb the eigenvalues, though as discussed in [15], a quantum Fourier transform is relatively insensitive to errors made during its performance. The second type will also create small perturbations, in either a consistent fashion (if the errors are systematic) or a random fashion (unsystematic errors); fortunately, the algorithm is fairly insensitive to these errors as well. Finally, we note that errors can also be handled using error correcting codes; although more than 50–100 physical qubits would then be required to realize the necessary logical qubits, this is also the case with other quantum algorithms when made fault tolerant, and it remains true that the algorithm can perform interesting calculations with only a few percent of the qubits required for an interesting factoring problem.

In conclusion, we have provided a new quantum algorithm which can be used to find eigenvectors and eigenvalues of a Hamiltonian operator. The algorithm provides an exponential speed increase when compared to the best known classical techniques. Problems from atomic physics may be the best place to perform the first real calculations, both because accurate experimental data are available to verify the resulting calculations, and because the parameters involved appear to be within the foreseeable range of small quantum computers. We estimate that 50–100 qubits are sufficient to perform interesting calculations that are classically intractable. Finally, we suggest a couple of interesting questions which remain open. First, although we have made estimates regarding numbers of required qubits, it would be interesting to calculate accurately the number of quantum logic gates required to do an interesting problem. Second, a more detailed analysis of the effects of errors would be worthwhile, as would an analysis of error correcting codes in this context.

D. S. A. acknowledges support from a NDSEG fellowship and thanks D. Lidar, C. Froese Fisher, and especially W. R. Johnson for helpful discussions. Portions of this research were supported by Grant No. N00014-95-1-0975 from the Office of Naval Research, and by ARO and DARPA under Grant No. DAAH04-96-1-0386 to QUIC, the Quantum Information and Computation initiative, and by a DARPA grant to NMRQC, the Nuclear Magnetic Resonance Quantum Computing initiative.

*Email address: daniel.s.abrams@jpl.nasa.gov
†Email address: slloyd@mit.edu

[1] P. Shor, in *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, edited by S. Goldwasser (IEEE Computer Society, Los Alamos, CA, 1994), p. 124.
[2] R. P. Feynman, Int. J. Theor. Phys. **21**, 467 (1982).
[3] S. Lloyd, Science **273**, 1073 (1996).
[4] D. S. Abrams and S. Lloyd, Phys. Rev. Lett. **79**, 2586 (1997).
[5] B. Boghosian and W. Taylor, Phys. Rev. E **57**, 54 (1998).
[6] C. Zalka, Proc. R. Soc. London Sect. A **454**, 313–322 (1998).
[7] S. Wiesner, quant-ph/9603028.
[8] D. Lidar and O. Biham, Phys. Rev. E **56**, 3661 (1997).
[9] A. Yu. Kitaev, quant-ph/9511026.
[10] R. Cleve, A. Ekert, C. Macchiavello, and M. Mosca, Proc. R. Soc. London Sect. A **454**, 339–354 (1998).
[11] Since $U(t)$ has the same eigenvalues and vectors for all t, this might lead one to falsely conclude that the number of operations necessary to find the eigenstates to a given accuracy could be reduced by choosing a shorter length of time t for the operator $U(t)$. However, the algorithm requires one to calculate U^M, and since $U(t)^M = U(Mt)$, one sees that $U = U(t)$ must be calculated with greater precision if U^M is to be calculated for a fixed precision. In fact, since the eigenvectors are determined with a precision proportional to M, the number of quantum logic operations required to calculate the energy eigenstates to a precision ϵ is seen to scale as ϵ^{-2}.
[12] A. Barenco *et al.*, Phys. Rev. A **52**, 3457 (1995).
[13] W. R. Johnson (private communication).
[14] The number of qubits required for the FFT is not as large as one might at first suppose, based on the earlier statement that the accuracy scales linearly with the size of the FFT. This statement is true only for a fixed U. By changing U—in particular, by increasing the length of time t in $U(t)$—one can obtain the eigenvalues to arbitrary precision using a fixed number of FFT points. However, the number of points in the FFT must be sufficiently large so as to separate the frequencies corresponding to distinct eigenvectors. This is how the estimate of 6 or 7 qubits (64 or 128 FFT points) is made.
[15] I. L. Chuang *et al.*, Science **270**, 1633–1635 (1995).