PHYSICAL REVIEW E VOLUME 59, NUMBER 2 FEBRUARY 1999

Calculating the thermal rate constant with exponential speedup on a quantum computer

Daniel A. Lidar* and Haobin Wang†
Department of Chemistry, The University of California at Berkeley, Berkeley, California 94720

(Received 14 July 1998)

It is shown how to formulate the ubiquitous quantum chemistry problem of calculating the thermal rate constant on a quantum computer. The resulting exact algorithm scales exponentially faster with the dimensionality of the system than all known ''classical'' algorithms for this problem. [S1063-651X(99)07302-X]

PACS number(s): 89.80.+h, 89.70.+c, 82.20.Pm, 82.90.+j

I. INTRODUCTION

It is well known that an exact calculation of the thermal rate constant is a problem that scales exponentially with the number of degrees of freedom [1]. But what if we had a quantum computer [2,3] (QC) at our disposal? As we will show here, then the calculation can be speeded up exponentially. Exponential speedups on QC's have been demonstrated in a number of problems, the most famous of which are Shor's algorithm for the factoring problem [4–6], and Grover's algorithm for database search [7]. In the context of physics problems, exponential speedup has been demonstrated mostly in the context of simulation of the many-body Schrödinger equation [8–10]. Other physics applications have also been proposed, such as studying quantum chaos [11] and Ising spin glasses [12]. At the experimental level QC's are still in a stage of infancy, although very impressive first steps toward implementation have been taken using ions in ion traps [13], atoms in high-finesse microwave cavities [14], and molecular spins in NMR [15]. In particular, Chuang, Gershenfeld, and Kubinec, using a chloroform NMR-QC, recently implemented for the first time a quantum algorithm (Grover's) which outperforms any classical algorithm designed to solve the same task [16]. Another algorithm for which QC's offer an exponential speedup compared to classical computers, known as ''Deutsch's problem'' [17], has also been implemented on a chloroform NMR-QC [18], and by Jones and co-workers on a cytosine NMR-QC [19,20]. These impressive achievements signal quite clearly, albeit for very simple applications at this point, that QC's may perhaps sooner than expected play an important role in simulations. For a comprehensive introduction to quantum computation, we refer the interested reader to a number of recent reviews [21].

The extensive body of work in quantum computation has, however, to date not addressed a computational problem of direct relevance to the field of quantum chemistry (apart, of course, from the general simulation of the Schrödinger equation). In this work we show how to formulate the ubiquitous quantum chemistry problem of calculating the thermal rate constant on a QC, and, by doing so, how the calculation can be speeded up exponentially with the number of degrees of freedom. The rate constant is the single most important number characterizing chemical reactions, and thus great efforts have been invested in designing efficient and *exact* ''classical'' computational ways to obtain it. Important progress along this line has been made by Miller and co-workers [22], Light and co-workers [23], and Manthe and co-workers [24], based on the efficient evaluation of the flux correlation function [25,26]. Approximate methods have also been developed for obtaining the rate constants. For example, the popular mixed quantum-classical model [27,28] whereby one integrates the time-dependent Schrödinger equation for (a few) degrees of freedom that are treated quantum mechanically, simultaneously with the classical equations of motion for the (many) degrees of freedom that are treated by classical mechanics; the semiclassical initial value representation [29] (SC-IVR), that has had a rebirth of interest [30–32] as a way for including quantum effects in molecular dynamics simulations; and the (further) linearizing approximation to the SC-IVR which leads to a much simpler form for the rate expression [33]. While these methods enjoy favorable computational scaling properties, they are inherently approximate and thus not in the context of the present paper.

In spite of these significant advances, exact classical algorithms can at most achieve a polynomial speedup in a problem that inherently scales exponentially. Indeed, when classical algorithms are described as $O(N^3)$ instead of $O(N^2)$, it should be remembered that N is itself exponentially large. The anticipated advance in quantum computation would therefore have revolutionary consequences for quantum chemistry, rendering ''classical'' simulation methods essentially obsolete. Of course, QC's are still at best many years away from reaching the point of replacing classical computers. Nevertheless, it is of considerable interest to exhibit an explicit QC algorithm for a problem as central as the computation of the rate constant, and this is the task we undertake here. The paper is organized as follows: In Sec. II we introduce some pertinent concepts of quantum computation. In Sec. III we briefly rederive the exact quantum expression for the rate constant, and discuss the computational difficulties associated with it. The next sections are the heart of the paper, where the QC algorithm for calculation of the rate constant is described in detail. Section VI concludes.

II. A BRIEF INTRODUCTION TO QUANTUM COMPUTATION

Let us denote all the degrees of freedom of the problem at hand by the collective variable $\vec{q} \equiv (q_1, \ldots, q_M)$, and let us

---

*Author to whom correspondence should be addressed. Electronic address: dani@holmium.cchem.berkeley.edu

†Electronic address: whb@neon.cchem.berkeley.edu

assume for simplicity that after discretization the number of points per degree of freedom is 2^l. Then the Hilbert space for a given Hamiltonian $\mathbf{H}(\vec{q},\vec{p})$ is of dimension

$$N=2^{\nu}, \quad \nu=lM, \tag{1}$$

from which we obtain the exponential scaling. To set up the problem on a QC one introduces a ''register'' of ν ''qubits'' (two-level systems), which can be in a superposition state $|\phi_i\rangle=a_i|0_i\rangle+b_i|1_i\rangle$ (with $|a_i|^2+|b_i|^2=1$). Each group of l qubits corresponds to one of the degrees of freedom q_j. Initially the quantum register $|\Phi\rangle$ is in a direct product state with all qubits in the $|0\rangle$ state:

$$|\Phi\rangle=\bigotimes_{i=1}^{\nu}|0_i\rangle. \tag{2}$$

Allowed operations on the register are all the *unitary transformations* (corresponding to propagation of the register), and all *measurements*, i.e., projections onto subspaces of the full register Hilbert space $\mathcal{H}$. However, by convention the unitary transformations should be explicitly given in terms of operations on a single qubit and two qubits at the most, since it is such one- and two-qubit ''gates'' that one can expect to construct in practice. Also, allowing arbitrarily large gates would not constitute a general-purpose computer (this is similar to the situation with classical computers, where the number of distinct logical elements is a small and finite set). Simplifying an early construction by Deutsch [3], it has been proven [34] that the set of *all single-qubit gates* [the group U(2)] with, in addition, the ''*controlled-not*'' (CNOT)

$$|\epsilon_1,\epsilon_2\rangle\mapsto|\epsilon_1,(\epsilon_1+\epsilon_2)\,\mathrm{mod}2\rangle \quad (\epsilon_i=0,1), \tag{3}$$

form a *universal* set of gates: all unitary transformations, of arbitrary size, can be constructed using a polynomially large set of one- and two-qubit gates. Therefore, the restriction to this set of gates is sufficient to simulate *any* computable function. This, of course, includes (and potentially exceeds) anything that is computable on a classical computer.

Now, let us see how to set up a superposition state on the quantum register corresponding to all possible initial classical positions. In the basis $|0\rangle=\binom{1}{0}$ and $|1\rangle=\binom{0}{1}$, one applies the one-qubit unitary ''Hadamard transform''

$$\mathbf{R}=\frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}. \tag{4}$$

Thus

$$\begin{aligned}|\Phi\rangle\mapsto|\Phi'\rangle&=\bigotimes_{i=1}^{\nu}\mathbf{R}|0_i\rangle=\bigotimes_{i=1}^{\nu}\frac{(|0_i\rangle+|1_i\rangle)}{\sqrt{2}}\\ &=\frac{1}{\sqrt{N}}\sum_{j=0}^{N-1}|j\rangle,\end{aligned} \tag{5}$$

where j is the decimal representation of the register state with the corresponding binary value, so $|j\rangle$ is a convenient shorthand notation for a tensor product of ν single-particle states. Equation (5) represents the desired superposition over all initial positions. For example, let $l=2$ (so the number of grid points per degree of freedom is 4) and $M=2$ (e.g., $q_1=x$ and $q_2=y$ in a two-dimensional problem involving a linear triatomic vibrating molecule), so $N=16$; how is $\vec{q}=(2\Delta x,3\Delta y)$ represented in the register? In binary, $\vec{q}=(\{1,0\}\Delta x,\{1,1\}\Delta y)$, so that $j=11$, i.e., $|j\rangle=|1\rangle\otimes|0\rangle\otimes|1\rangle\otimes|1\rangle$ corresponds to $x=2\Delta x$ and $y=3\Delta y$. In this way the quantum register supports a superposition over all discretized values of the degrees of freedom $\vec{q}$. By linearity, unitary evolution of the register amounts to parallel propagation on all of the exponentially many grid points. We will find it convenient to work from now on with j as the collective degrees of freedom variable, instead of $\vec{q}$. The transformation between the two is straightforward. Note further that the register states $|j\rangle$ are *position eigenstates*.

Suppose now that the initial wave function $|\Psi(0)\rangle$ in our scattering problem has amplitude α_j to be at position j. As explained below, it is in fact not essential to utilize this initial condition, and an equal superposition of all possible position eigenstates will be sufficient in most cases. Nevertheless, as was shown by Zalka [9] (and will not be repeated here), it is possible to initialize the register to the state

$$|\Phi'\rangle\mapsto|\Phi''\rangle=\sum_{j=0}^{N-1}\alpha_j|j\rangle \tag{6}$$

by means of a suitable unitary transformation. This then represents $|\Psi(0)\rangle$ on the QC. The dynamics in the scattering problem is determined by the unitary propagator $\mathbf{U}=e^{-i\mathbf{H}t/\hbar}$: $|\Psi(t)\rangle=\mathbf{U}|\Psi(0)\rangle$. The crucial advantage offered by a QC is that, as will be shown below, it is possible to implement this propagator *effficiently* on the quantum register, so that

$$|\Phi''\rangle\mapsto|\Phi'''\rangle=\mathbf{U}|\Phi''\rangle. \tag{7}$$

In this way we have set up a one-to-one correspondence between the QC ($|\Phi\rangle$) and the dynamics of the problem of interest ($|\Psi\rangle$). All the relevant information can be extracted from this simulation by observing the states of the qubits.

III. THERMAL RATE CONSTANT VIA THE FLUX CORRELATION FUNCTION FORMALISM

Let us now turn to the scattering problem, and define a flux operator

$$\mathbf{F}=\frac{i}{\hbar}[\mathbf{H},\mathbf{h}(s(\vec{q}))], \tag{8}$$

where $\mathbf{h}$ is the Heaviside function and the condition $s(\vec{q})=0$ defines the dividing surface.

The thermal rate constant is written as the time integral of the flux-flux autocorrelation function [25]:

$$k(T)=\frac{1}{Q_r(T)}\int_0^{\infty}dt\,C_f(t), \tag{9}$$

where $Q_r(T)$ is the reactant partition function per unit volume, and

$$C_f(t)=\mathrm{Tr}[e^{-\beta\mathbf{H}/2}\mathbf{F}e^{-\beta\mathbf{H}/2}e^{i\mathbf{H}t/\hbar}\mathbf{F}e^{-i\mathbf{H}t/\hbar}]$$
$$=\mathrm{Tr}[\mathbf{F}e^{i\mathbf{H}t/\hbar-\beta\mathbf{H}/2}\mathbf{F}e^{-i\mathbf{H}t/\hbar-\beta\mathbf{H}/2}]$$
$$=\mathrm{Tr}[\mathbf{F}e^{i\mathbf{H}\tau^*/\hbar}\mathbf{F}e^{-i\mathbf{H}\tau/\hbar}], \tag{10}$$

and where, for convenience, we have also defined the complex ''time''

$$\tau=t-i\hbar\beta/2. \tag{11}$$

Evaluating the trace in the energy eigenbasis $\{|n\rangle\}$ (with $\mathbf{H}|n\rangle=E_n|n\rangle$), we obtain

$$C_f(t)=\sum_n \langle n|\mathbf{F}e^{i\mathbf{H}\tau^*/\hbar}\mathbf{F}e^{-i\mathbf{H}\tau/\hbar}|n\rangle$$
$$=\sum_{n,m}\langle n|\mathbf{F}e^{i\mathbf{H}\tau^*/\hbar}|m\rangle\langle m|\mathbf{F}e^{-i\mathbf{H}\tau/\hbar}|n\rangle$$
$$=\sum_{n,m} e^{iE_m\tau^*/\hbar}e^{-iE_n\tau/\hbar}\langle n|\mathbf{F}|m\rangle\langle m|\mathbf{F}|n\rangle$$
$$=\sum_{n,m} e^{-\beta(E_m+E_n)/2}e^{i(E_m-E_n)t/\hbar}|\langle n|\mathbf{F}|m\rangle|^2. \tag{12}$$

Using the commutator form for **F** [Eq. (8)], we find:

$$\langle n|\mathbf{F}|m\rangle=\frac{i}{\hbar}\langle n|[\mathbf{H}\mathbf{h}(s(\vec{q}))-\mathbf{h}(s(\vec{q}))\mathbf{H}]|m\rangle$$
$$=\frac{i}{\hbar}(E_n-E_m)\langle n|\mathbf{h}(s(\vec{q}))|m\rangle. \tag{13}$$

Recall from Eq. (5) that the quantum register naturally supports a superposition over position eigenstates. Accordingly, let us represent $|n\rangle$ in the discretized position basis $\{|j\rangle\}$, which we from now on identify with the QC's ''computational basis'' $\{|j\rangle\}$ (indeed, the correspondence is one to one). Thus let us expand the energy eigenstates as

$$|n\rangle=\sum_{j=0}^{N-1} a_j(n)|j\rangle. \tag{14}$$

Clearly, the Heaviside function $\mathbf{h}(s(\vec{q}))$ is diagonal in this basis, so that, from Eq. (13),

$$\langle n|\mathbf{F}|m\rangle=\frac{i}{\hbar}(E_n-E_m)\sum_{j=0}^{N-1} a_j^*(n)a_j(m)h(s(j)). \tag{15}$$

Hence, finally,

$$C_f(t)=\frac{1}{\hbar^2}\sum_{n\neq m} e^{-\beta(E_m+E_n)/2}e^{i(E_m-E_n)t/\hbar}(E_n-E_m)^2 \times\left|\sum_{j=0}^{N-1} a_j^*(n)a_j(m)h(s(j))\right|^2. \tag{16}$$

Equation (16) is the *formal* quantum-mechanical expression for obtaining the thermal rate constant. At first glance it may seem that one has to sum over every eigenstate of the Hamiltonian operator within at least a certain energy range. In reality this is hardly the case, nor is it possible, with the obvious reason being that for a scattering problem, one usually deals with a continuum in energy spectrum and there could be an infinite number of such eigenstates. As a result, the summations in Eq. (16) are actually integrations over the energy, which can be carried out by employing various quadrature schemes. The number of energy quadrature points [the number of eigenstates in Eq. (16)] determines the accuracy of the integral, which often can be kept rather small. For example, in quantum scattering calculations for small gas-phase chemical reactions, when obtaining a thermal rate constant from microcanonical cumulative reaction probabilities [analogous to Eq. (16)], the integration over energy is usually done by employing less than 100 quadrature points. In treating more complex systems, one may need significantly more quadrature points. The lesson from such classical calculations is that the number of quadrature points in many cases of chemical interest does not scale exponentially with the size of the problem, and thus does not form a bottleneck for the computation on a QC. More specifically, for the purpose of achieving numerical convergence, one does not have to include *all* the energy eigenstates, but only to obtain a reasonable resolution of the energy spectrum. This statement is universal and is certainly valid also for the case of quantum computation.

Our task is therefore to find an algorithm that calculates the spectrum $\{E_n\}$, and the position amplitudes $\{a_j(n)\}_{j=0}^{N-1}$ for each of the eigenstates $|j\rangle$. With these in hand the rest of the calculation [summations in Eq. (16)] can be efficiently implemented on a classical computer. The summation over *energy* eigenstates in Eq. (16) should only extend over a polynomially large number of energies. There are two reasons for this: first, from the discussion above one is only interested in obtaining representative (quadrature) points of the energy spectrum in order to perform this summation; and second, since the exponential decrease due to the Boltzman factors will effectively eliminate the higher end of the spectrum $E\gtrsim k_BT+$ (barrier height) even if there are many peaks in the spectrum. As for the summation over the *position* eigenstates in Eq. (16), the point is again that one only needs to obtain a reasonable sample of the distribution. The number of position states required for numerical convergence is polynomially related to the number of energy states (in the optimal case, the relation is one to one), and thus also scales polynomially with the size of the problem. Some numerical tricks can of course help to further improve the situation. For example, if the distribution is badly behaved in position space, one could Fourier transform to momentum space and sample there. At any rate, once the problem of dealing with exponentially many states has been eliminated, the problem is essentially reduced to the one usually faced by any classical simulation, where a variety of efficient methods are available. Note, however, that unlike the classical case where attention has to be given to the issue of generating statistically *independent* Monte Carlo samples, the quantum simulation *automatically* generates truly independent samples by the projection postulate [35].

We emphasize here that though the summations in Eq. (16) are performed over a polynomial number of energy and position states and quadrature points, the solution of the Schrödinger equation is still carried out in an exponentially large Hilbert space. Such exponential scaling is inherent to performing quantum mechanical simulations. It is *this* property that forms the computational bottleneck on a classical computer. At the same time, this is also the part of the simulation that can be solved efficiently on a quantum computer.

How can we efficiently calculate the eigenstates and the spectrum? Solving the time-*in*dependent Schrödinger equation (SE)

$$\mathbf{H}|\psi\rangle = E|\psi\rangle \tag{17}$$

can be done on a QC by transforming the problem to the time-dependent SE and propagating the dynamics with the unitary time-evolution operator $\mathbf{U}=e^{-i\mathbf{H}t}$. This can also be understood from the relation between the time-dependent and independent pictures through the formal definition of the Green's function,

$$\begin{aligned}\mathbf{G}(E) &= \lim_{\epsilon\to 0}(E+i\epsilon-\mathbf{H})^{-1} \\ &= \lim_{\epsilon\to 0}(i\hbar)^{-1}\int_0^\infty dt\, e^{i(E+i\epsilon-\mathbf{H})t/\hbar},\end{aligned} \tag{18}$$

where ϵ, a formal convergence factor, is the absorbing potential in practical calculations [1]. Once the problem is transformed into the time-dependent picture, each energy eigenvalue and eigenstate can be obtained via known quantum algorithms, to be detailed below, in polynomial time.

The procedure outlined above is suitable for performing quantum rate constant calculations on a QC, because *it casts the problem in terms of a discrete Fourier transform, which is known to be efficiently implementable on a QC*. Great efforts and ingenuity have been applied in developing fast methods for calculating the rate constant on a classical computer, most of which use iterative techniques to exploit the low rank feature of certain operators. For completeness, we briefly summarize two representative such methods in the Appendix. We emphasize, however, that while such methods are highly successful for computations on a classical computer, it is at present not clear how they can be implemented with equal efficiency on a quantum computer. This is because these low-rank methods, together with many other similar approaches, rely on the efficient implementation of matrix algebra, for which no fast quantum algorithms are currently available. It is fair to say that at this point, all known efficient quantum algorithms [4–10] require the discrete Fourier transform, or its generalizations [36]. At any rate, the speedup achieved by the efficient classical methods is, as mentioned above, polynomial in a problem that inherently scales exponentially.

IV. GENERAL OUTLINE OF THE ALGORITHM

We now give our algorithm in general terms, to be defined more precisely in Sec. V.

(1) Prepare a register as in Eq. (2), and attach some ''ancilla'' qubits to it, also in the $|\mathbf{0}\rangle$ state. These will serve as a quantum scratch pad to record the results of intermediate measurements. From now on we will distinguish between the ''main'' and ancillary registers.

(2) If a good guess for the initial wave function is known, initialize the register to it as in Eq. (6). Otherwise initialize the register to an equal superposition. Since the computational basis states are *position* eigenstates, in all likelihood they are not energy eigenstates, so will not be stationary under the SE dynamics. Thus except if the equal superposition corresponds to some undesirable position—such as very high above the barrier, so that dissociation sets in immediately—this is as good a guess as any. In fact, any random (but reproducible) initial distribution will do.

(3) ''Propagate'' the register in parallel for a time t. This corresponds to a parallel evolution of all the position eigenstates. The propagation is done very much in analogy to the classical fast Fourier transform (FFT) method [37], in particular the split time propagation scheme [38]. That is, the potential part is diagonal and can be implemented directly, whereas for the kinetic part it is necessary to Fourier transform to and back from momentum space.

(4) Perform a ''von Neumann'' measurement (see Sec. V D) on the *ancillary* register using the Hamiltonian (energy) as the observable. This accomplishes a double purpose: (a) It allows one to obtain an energy E_n by measuring the ancillas. (b) It provides a means to sample the energy-position amplitudes $a_j(n)$.

(5) Repeat steps (1)–(4) many times until the distribution is converged to the desired accuracy for all relevant eigenstates. The number of required repetitions is proportional to this accuracy.

(6) Calculate (classically) the sums in Eq. (16).

V. ALGORITHM IN DETAIL

A. Initialization

Here the register is initialized to the state $|\Phi\rangle = \otimes_{i=1}^{2\nu}|0_i\rangle$, where the last ν qubits are ancillas. The physics of this initialization step depends on the QC implementation. One conceivable way is cooling to the ground state.

B. Inputting the initial wave function

If necessary, one inputs the initial wavefunction by the technique of Zalka [9]. Otherwise one employs the Hadamard rotations technique to create an equal superposition over position states, as in Eq. (5). In the former case the register will be in the state:

$$|\Phi''\rangle = \left(\sum_{j=0}^{N-1}\alpha_j|j\rangle\right)\bigotimes_{i=\nu}^{2\nu}|0_i\rangle. \tag{19}$$

In the latter case, all $\alpha_j = 1$.

C. Quantum propagation algorithm

This subsection is the heart of the algorithm. Assume for simplicity that we have a single particle of mass m in an external potential $V(\vec{q})$. The full Green's function for arbitrary time t is

$$G(x_1,x_2;t) = \langle x_1|e^{-i\mathbf{H}t/\hbar}|x_2\rangle. \tag{20}$$

For short time steps $\Delta t \ll 1/E$(E is a typical energy of the system), this becomes approximately

$$G(x_1,x_2;\Delta t)=\kappa \exp\left[im\frac{(x_1-x_2)^2}{2\Delta t}-iV(x_1)\Delta t\right], \tag{21}$$

where κ is a normalization factor. Applying this to the amplitudes is equivalent to acting on the basis states with the inverse transformation. Thus the position eigenstates, properly normalized, transform as:

$$|j\rangle \mapsto \mathbf{U}|j\rangle=\frac{1}{\sqrt{N}}\sum_{j'=0}^{N-1}\exp\left[-im\frac{(j-j')^2\Delta x^2}{2\Delta t}\right.$$
$$\left.+iV(j\Delta x)\Delta t\right]|j'\rangle. \tag{22}$$

This is carried out in parallel on the entire superposition $\Sigma_{j=0}^{N-1}|j\rangle$. Suppose the time step and spatial resolution are adjusted so that

$$\frac{m\Delta x^2}{\Delta t}=\frac{2\pi}{N}. \tag{23}$$

Then, by expanding the exponent, Eq. (22) can be written as a succession of a diagonal transformation, Fourier transform, and another diagonal transformation, all unitary:

$$\mathbf{U}|j\rangle=\exp[iF_2(j)]\mathcal{F}(j,j')\exp[iF_1(j')]|j'\rangle, \tag{24}$$

where

$$F_1(j)=-\pi\frac{j^2}{N},$$

$$F_2(j)=-\pi\frac{j^2}{N}+V(j\Delta x)\Delta t,$$

$$\mathcal{F}(j,j')|j\rangle=\frac{1}{\sqrt{N}}\sum_{j'=0}^{N-1}\exp\left[2\pi i\frac{jj'}{N}\right]|j'\rangle. \tag{25}$$

Equation (23) tells us how many qubits $\nu=\log_2 N$ are needed for given Δx and Δt:

$$\nu=\log_2\frac{2\pi\Delta t}{m\Delta x^2}. \tag{26}$$

The special form of Eq. (24), involving diagonal transformations and a Fourier transform, is due to the structure of the Hamiltonian operator as a sum of operators diagonal in coordinate and momentum space. As mentioned above, this is very similar to the situation that arises in the classical FFT method for solving the SE [37].

1. Diagonal transformations

Consider first executing the diagonal unitary transformations $|j\rangle\mapsto\exp[iF(j)]|j\rangle$, which can be done as follows, using the ancillary register [Eq. (19)], in the state $|\mathbf{0}\rangle\equiv\otimes_{i=\nu}^{2\nu}|0_i\rangle$. The number ν of qubits in this register depends on the accuracy with which F needs to be evaluated (see immediately below). Then the following steps are applied.

(1) $|j,\mathbf{0}\rangle\mapsto|j,F(j)\rangle$: evaluation of F and storage of the result in the ancillary register;

(2) $|j,F(j)\rangle\mapsto\exp[iF(j)]|j,F(j)\rangle$: introducing the phase; and

(3) $\exp[iF(j)]|j,F(j)\rangle\mapsto\exp[iF(j)]|j,\mathbf{0}\rangle$: inversion of step (1) in order to clear the ancillary register.

Step (1) requires that it is possible to evaluate an arbitrary function and store the result. This is very similar to the equivalent classical problem, for which algorithms are known using just the elementary classical gates. The same can be done in the quantum case, by breaking up the evaluation into elementary arithmetic operations, for which quantum algorithms have been designed [4,39]. We will not dwell on this issue here. Step (3) is just the reverse of step (1), and can therefore be implemented by running the inverse unitary transformation.

Step (2) has no classical analog since it involves phases. It can be implemented if one knows how to do $|x\rangle\mapsto\exp[iax]|x\rangle$. This can be done by simple single-qubit phase shifts. Let $\nu=2k$. Using a binary expansion $x=\Sigma_{l=-k}^{k-1}x_l 2^l$, we have $|x\rangle=|x_{-k}\rangle\otimes|x_{-k+1}\rangle\otimes\cdots\otimes|x_{k-1}\rangle$, where $x_l=0,1$. In the standard basis $|0\rangle=\binom{1}{0}$, $|1\rangle=\binom{0}{1}$, consider the unitary operation

$$\mathbf{Q}=\bigotimes_{l=-k}^{k-1}\begin{pmatrix}1 & 0\\ 0 & e^{i2^l}\end{pmatrix}. \tag{27}$$

The lth 2×2 matrix is a unitary operation in the Hilbert space of qubit number l. Thus

$$\begin{pmatrix}1 & 0\\ 0 & e^{i2^l}\end{pmatrix}|x_l\rangle=e^{ix_l 2^l}|x_l\rangle. \tag{28}$$

Therefore the full result is

$$\mathbf{Q}|x\rangle=\bigotimes_{l=-k}^{k-1}e^{ix_l 2^l}|x_l\rangle=e^{i\Sigma_{l=-k}^{k-1}x_l 2^l}\bigotimes_{l=-k}^{k-1}|x_l\rangle=e^{ix}|x\rangle, \tag{29}$$

as required.

2. Quantum Fourier transform

The quantum Fourier transform (QFT) algorithm has been discussed extensively [5,40–42], and some beautiful connections to group theory have been made [42]. In view of its central importance in the algorithm for solving the SE (and indeed in *all* efficient quantum algorithm found so far), we present a brief derivation here, using the approach of Cleve *et al.* [40].

The QFT was defined in Eq. (25). Using the binary notation $j/N=0.j_1 j_2\ldots j_\nu$ (recall that $N=2^\nu$) where $j_1=0,1$ etc., we note first that

$$e^{2\pi ijj'/2^\nu}|j'_1,j'_2,\ldots,j'_\nu\rangle=e^{2\pi i(0.j_\nu)j'_1}|j'_1\rangle\otimes e^{2\pi i(0.j_{\nu-1}j_\nu)j'_2}|j'_2\rangle\otimes\cdots\otimes e^{2\pi i(0.j_1 j_2\ldots j_\nu)j'_\nu}|j'_\nu\rangle. \tag{30}$$

It follows that

$$\sum_{j'=0}^{N-1}\exp\left[2\pi i\frac{jj'}{N}\right]|j'\rangle=(|0\rangle+e^{2\pi i(0.j_\nu)}|1\rangle)\otimes(|0\rangle+e^{2\pi i(0\cdot j_{\nu-1}j_\nu)}|1\rangle)\otimes\cdots\otimes(|0\rangle+e^{2\pi i(0\cdot j_1j_2\cdots j_\nu)}|1\rangle), \tag{31}$$

by expanding out the product on the right-hand-side and a term-by-term comparison. Thus the Fourier-transformed state in Eq. (31) is in fact an ‘‘unentangled’’ direct product. This fact greatly simplifies the implementation of the QFT.

To perform the QFT, one first applies a Hadamard rotation [Eq. (4)] to $|j_1\rangle$ (the first qubit of $|j\rangle$), with the result

$$\mathbf{R}|j_1\rangle=(|0\rangle+(-1)^{j_1}|1\rangle)=(|0\rangle+e^{2\pi i(0\cdot j_1)}|1\rangle), \tag{32}$$

so $|j\rangle\mapsto(|0\rangle+e^{2\pi i(0.j_1)}|1\rangle)|j_2,\dots,j_\nu\rangle$. Let us now define a new single-qubit operation, similar to $\mathbf{Q}$ from Eq. (27):

$$\mathbf{Q}_l=\begin{pmatrix}1&0\\0&e^{2\pi i/2^l}\end{pmatrix}. \tag{33}$$

This operation is applied on the first qubit $|j_1\rangle$, subject to a *control* by a second qubit $|j_l\rangle$ (which itself does not change): a ‘‘controlled rotation.’’ That is, if $j_l=0$, one does nothing; if it is 1, one applies $\mathbf{Q}_l$. This can be written as the following unitary transformation in the four-dimensional Hilbert space of the two qubits, in the standard basis $|j_1j_l\rangle=|00\rangle=(1,0,0,0)$, $|01\rangle=(0,1,0,0)$, $|10\rangle=(0,0,1,0)$, $|11\rangle=(0,0,0,1)$:

$$C\mathbf{Q}_l=\begin{pmatrix}1&0&&\\0&1&&\\&&\multicolumn{2}{c}{\mathbf{Q}_l}\end{pmatrix}. \tag{34}$$

After applying $C\mathbf{Q}_2$, one obtains

$$(|0\rangle+e^{2\pi i(0.j_1j_2)}|1\rangle). \tag{35}$$

Next a ‘‘controlled-$\mathbf{Q}_3$’’ is applied, yielding

$$(|0\rangle+e^{2\pi i(0.j_1j_2j_3)}|1\rangle). \tag{36}$$

Clearly, this process will eventually generate the desired phase in the superposition state of the first qubit [corresponding to the last qubit in Eq. (31)]:

$$\left[\left(\prod_{l=2}^{\nu}C\mathbf{Q}_l\right)\mathbf{R}\right]_1|j_1\rangle=(|0\rangle+e^{2\pi i(0\cdot j_1j_2\cdots j_\nu)}|1\rangle), \tag{37}$$

where the terms in the product from here onwards are applied *low index first*.

Now we turn to the second qubit. Again, a Hadamard rotation on it has the effect of: $\mathbf{R}|j_2\rangle=(|0\rangle+e^{2\pi i(0.j_2)}|1\rangle)$. This is followed by a controlled-$\mathbf{Q}_2$, conditioned upon $|j_3\rangle$: $(|0\rangle+e^{2\pi i(0.j_2)}|1\rangle)\mapsto(|0\rangle+e^{2\pi i(0.j_2j_3)}|1\rangle)$. After the full operation on $|j_2\rangle$, one obtains

$$\left[\left(\prod_{l=2}^{\nu-1}C\mathbf{Q}_l\right)\mathbf{R}\right]_2|j_2\rangle=(|0\rangle+e^{2\pi i(0.j_2j_3\cdots j_\nu)}|1\rangle)|j_2\rangle. \tag{38}$$

which corresponds to the one before last qubit in Eq. (31).

The method to generate the entire product in Eq. (31) should now be clear; collecting all the transformations yields

$$|j\rangle\mapsto\prod_{p=1}^{\nu-1}\left[\left(\prod_{l=2}^{\nu-p}C\mathbf{Q}_l\right)\mathbf{R}\right]_p|j_1,\dots,j_\nu\rangle=(|0\rangle+e^{2\pi i(0.j_1j_2\cdots j_\nu)}|1\rangle)\otimes\cdots\otimes(|0\rangle+e^{2\pi i(0\cdot j_{\nu-1}j_\nu)}|1\rangle)\otimes(|0\rangle+e^{2\pi i(0\cdot j_\nu)}|1\rangle). \tag{39}$$

Up to an unimportant bit reversal (which can easily be rectified by permuting the role of the qubits in the transformations above), this is exactly the desired result. In other words, the QFT is simply

$$\mathcal{F}=\prod_{p=1}^{\nu-1}\left[\left(\prod_{l=2}^{\nu-p}C\mathbf{Q}_l\right)\mathbf{R}\right]_p. \tag{40}$$

This will be applied in parallel, by virtue of the superposition principle, on all position eigenstates $|j\rangle$. Most importantly, the number of operations (single- and two-qubit) needed to implement the QFT is seen to be a mere $\nu(\nu-1)/2$. This is to be compared to the $\nu 2^\nu$ operations required classically, and, as emphasized above, is the ‘‘secret’’ behind the quantum speedup.

D. von Neuman measurements

Combining Eqs. (7) and (22), at this point the register is in the state

$$|\Phi'''\rangle=\sum_j\alpha_j\mathbf{U}|j\rangle=\sum_{j'}\psi_{j'}(t)|j'\rangle,$$

$$\psi_{j'}(t)=\sum_j\alpha_jG^{-1}(j,j';t). \tag{41}$$

A parallel propagation has occurred on all the position eigenstates. By measuring the qubits one by one, i.e., projecting onto a random position eigenstate $|j'\rangle$, and repeating this process many times while collecting the statistics, one can sample the electronic density function $|\psi_{j'}(t)|^2$. Our goal was to find the *energy*-spectrum and energy-position amplitudes $a_j(n)$, so these should be obtained from the simulation. This can be done using the so-called ''von Neuman measurement'' trick [9]. We will require an additional propagation step.

A ''measurement apparatus'' that can be made to interact with the QC is introduced, and is assumed to be equivalent to a one-dimensional quantum mechanical particle. That is, its Hilbert space is spanned by the basis vectors $|x\rangle$, x real, with $\mathbf{X}|x\rangle = x|x\rangle$. In practice this will be another ancillary quantum register, consisting of, say, K qubits. Now, let us expand the position eigenstates $|j'\rangle$ in terms of the complete set of energy eigenstates [recall Eq. (14)]:

$$|j'\rangle = \sum_n a_{j'}^*(n)|n\rangle. \tag{42}$$

Consider next the joint evolution of an energy eigenstate $|n\rangle$ and the apparatus state $|x\rangle$ (x is arbitrary), under the unitary operator $\tilde{\mathbf{U}} = \exp(i\mathbf{HP}t)$, where $[\mathbf{X},\mathbf{P}] = i$. Here $\mathbf{H}$ acts on the main register, and $\mathbf{X}$ and $\mathbf{P}$ act on the apparatus, so $[\mathbf{X},\mathbf{H}] = [\mathbf{P},\mathbf{H}] = 0$. We will shortly discuss the implementation of $\tilde{\mathbf{U}}$. Consider first a formal Taylor expansion of $\exp(i\mathbf{HP}t)$, which yields

$$\tilde{\mathbf{U}}|n\rangle|x\rangle = \sum_{l=0}^{\infty} \frac{1}{l!}(tE_n)^l|n\rangle \frac{\partial^l}{\partial x^l}|x\rangle = |n\rangle|x + tE_n\rangle. \tag{43}$$

Thus $\tilde{\mathbf{U}}$ does not change the energy eigenstate, but has the effect of ''shifting the dial x'' by an amount proportional to the energy E_n. The effect on the position eigenstate $|j'\rangle$ will be

$$\tilde{\mathbf{U}}|j'\rangle|x\rangle = \sum_n a_{j'}^*(n)|n\rangle|x + tE_n\rangle, \tag{44}$$

and the effect on the full superposition of Eq. (41) is

$$\begin{aligned}\tilde{\mathbf{U}}|\Phi'''\rangle|x\rangle &= \sum_{j'} \psi_{j'}(t) \sum_n a_{j'}^*(n)|n\rangle|x + tE_n\rangle \\ &= \sum_n \xi_n(t)|n\rangle|x + tE_n\rangle, \end{aligned} \tag{45}$$

$$\xi_n(t) = \sum_{j'} a_{j'}^*(n)\psi_{j'}(t).$$

Now suppose we *observe* the state of the apparatus. From Eq. (45) it is clear that the apparatus has become entangled with the QC, and by performing the observation the superposition will collapse onto a particular state $|m\rangle|x + tE_m\rangle$. This happens with probability $|\xi_m(t)|^2$. Recall that $|x + tE_m\rangle$ is represented in binary by the qubits of the apparatus. Since t is a parameter of the simulation and x is known, all that remains is to measure the apparatus qubit by qubit, to obtain the energy eigenvalue E_m. The accuracy with which these numbers are obtained is proportional to the number of simulation steps [9].

To implement $\tilde{\mathbf{U}}$ it is necessary to Fourier transform the $|x\rangle$ register, just as in the classical FFT case. Specifically, let us define the Fourier transform pair

$$\begin{aligned}\mathcal{F}|x\rangle &= |p\rangle = \frac{1}{\sqrt{2^K}} \sum_{x=0}^{2^K-1} e^{-ixp}|x\rangle, \\ \mathcal{F}|p\rangle &= |x\rangle = \frac{1}{\sqrt{2^K}} \sum_{p=0}^{2^K-1} e^{ixp}|p\rangle. \end{aligned} \tag{46}$$

Then starting from the initial apparatus state $|x\rangle$, $\tilde{\mathbf{U}}$ can be implemented as follows:

$$\begin{aligned}&|n\rangle|x\rangle \overset{\mathcal{F}(x)}{\mapsto} |n\rangle|p\rangle \\ &\overset{\tilde{\mathbf{U}}}{\mapsto} e^{iE_n pt}|n\rangle|p\rangle \\ &\overset{\mathcal{F}^{-1}(p)}{\mapsto} |n\rangle \frac{1}{\sqrt{2^K}} \sum_{p=0}^{2^K-1} e^{i(x+E_n t)p}|p\rangle \\ &= |n\rangle|x + E_n t\rangle, \end{aligned} \tag{47}$$

in agreement with Eq. (43).

E. Extracting the amplitudes from the measurements

Note further that after observation of the apparatus, the state of the main register has been projected onto $|m\rangle$, an energy eigenstate. The $\tilde{\mathbf{U}}$ propagation had a remarkable outcome: it transformed the information in the main register from a mixture over position eigenstates $|j'\rangle$ to one over energy eigenstates $|n\rangle$. The QC is consequently in an energy eigenstate, which can in principle be used in a further propagation step. However, our next task is to obtain the amplitudes $a_j(n)$ needed to complete the calculation in Eq. (16). Note first that, by Eq. (45),

$$\sum_n \xi_n(t) a_j(n) = \psi_j(t). \tag{48}$$

Now, by performing the whole procedure a sufficient number of times, the simulation yields an estimate of the *probabilities* $|\psi_j(t)|^2$ [Eq. (41)] and $|\xi_n(t)|^2$ [Eq. (45)]. Thus, to specify fully the complex numbers $a_j(n)$, it is necessary to also know their phases, as well as those of the $\xi_n(t)$ and $\psi_j(t)$.

To obtain the phases, we note first that it is sufficient to know only the *signs*, since no generality is lost by employing a *real* initial wave function $|\Psi(0)\rangle$. The signs can then be obtained with the help of a simple trick (new, as far as we know), which we will illustrate on a generic two-qubit register state $|\psi\rangle = a_0|00\rangle + a_1|01\rangle + a_2|10\rangle + a_3|11\rangle$. Given repeated preparations of this $|\psi\rangle$, we perform the following set of measurements:

(i) Observation of the two qubits in $|\psi\rangle$.

(ii) A Hadamard transform on the first qubit, followed by observation of the two qubits.

(iii) A Hadamard transform on the second qubit, followed by observation of the two qubits.

Step (i) yields an estimate of the $|a_i|$. Step (ii) yields an estimate of $|a_0 \pm a_1|$ and $|a_2 \pm a_3|$, since under the Hadamard transform $|\psi\rangle \mapsto (1/\sqrt{2})[(a_0+a_1)|00\rangle + (a_0-a_1)|01\rangle + (a_2+a_3)|10\rangle + (a_2-a_3)|11\rangle]$. Similarly, step (iii) yields an estimate of $|a_0 \pm a_2|$ and $|a_1 \pm a_3|$, since $|\psi\rangle \mapsto (1/\sqrt{2})[(a_0+a_2)|00\rangle + (a_0-a_2)|01\rangle + (a_1+a_3)|10\rangle + (a_1-a_3)|11\rangle]$. Clearly, this provides sufficient information for extraction of the signs of all amplitudes. The generalization to a ν-bit register is obvious: one performs Hadamard rotations on all ν qubits. This then yields $\{|a_0 \pm a_1|, |a_2 \pm a_3|, |a_4 \pm a_5|, \dots\}$ (after Hadamard on first qubit), $\{|a_0 \pm a_2|, |a_1 \pm a_3|, |a_4 \pm a_6|, \dots\}$ (after Hadamard on second qubit), etc. After each Hadamard rotation there are 2^ν coefficients to be estimated. This exponential ''Monte Carlo scaling'' is the same as the one we encountered before, and is not considered a slowdown for the reasons detailed above. The additional computational cost is in the Hadamard rotations, ν of which must be performed. This does therefore not affect the efficiency of the algorithm. At the end of the process, if the whole phase space has been sampled, one is left with $\nu 2^\nu$ absolute values equations, which contain sufficient information to solve for the signs of all the amplitudes. In practice one will of course sample only a small (polynomial) portion of the phase space, and care must then be taken to obtain sufficient equations of the type above to determine uniquely the signs of the amplitudes of interest.

F. Repetition

The steps outlined above generate the energies $\{E_n\}$ and estimates of amplitudes $\{a_j(n)\}$ needed to perform the sum in Eq. (16). The whole process must now be repeated many times, on the order of the required accuracy, in order to complete the calculation. Due to the speedup in the implementation of the propagation step, the algorithm performs exponentially faster than any exact classical algorithm designed to solve the same task.

VI. DISCUSSION AND CONCLUSIONS

Quantum computers are still far from being a panacea, and serious doubts have been raised whether they will ever replace ordinary, classical computers [43]. Such worries are invariably based on the immense difficulties associated with maintaining phase coherence throughout the computation, i.e., the ''decoherence problem.'' However, a remarkable theory of quantum error correction codes has recently been constructed [44], in which a ''logical qubit'' is encoded in the larger Hilbert space of several physical qubits [45]. It has been shown that as long as the error rate is sufficiently small, it is possible to perform *fault-tolerant* quantum computation, i.e., the computation can be stabilized and be made fully robust to errors [46]. These advances greatly enhance the prospects of the eventual construction of useful QC's, beyond the current highly rudimentary prototypes. Building on these hopes, here we have presented an algorithm for calculating the thermal rate constant on a QC. The algorithm involves an initialization step of the QC into an equal superposition of position eigenstates; a propagation using an adaptation to QC's of the well-known FFT technique; and finally, a sequence of measurements yielding the energy spectrum and amplitudes. Under reasonable assumptions about the distribution of energy eigenvalues the algorithm runs in polynomial time. The algorithm thus outperforms any exact classical simulation, which is bound to be exponential. This clearly demonstrates the potential utility of QC's in future applications to quantum chemistry problems.

Our approach was somewhat of a ''brute force'' one, in that we did not attempt to optimize the algorithm using such fruitful concepts as ''direct and correct'' low-rank expressions for the rate constant [1] (also see the Appendix). Such optimizations, while ineffectual in altering the essential exponential speedup achieved by use of a QC, may still be important in practice, especially in the early stages of the application on a small-scale QC of an algorithm such as described here. Further work is hence desirable to optimize the algorithm.

Finally, it would be interesting to check the effect of noise and other types of errors affecting the evolution of the QC on the present algorithm. It has been shown, e.g., in the case of the ion trap QC, that factoring becomes impossible once random phase fluctuations in the laser pulses exceed a certain threshold [47]. We intend to study similar noise-related issues using numerical simulations in the context of the present algorithm.

ACKNOWLEDGMENTS

We wish to thank Professor William H. Miller for a critical reading of the manuscript and for useful comments, and Dr. Christof Zalka and Dr. Daniel Abrahams for helpful discussions. D.A.L. gratefully acknowledges support from the U.S. Army Research Office under Grant No. DAAG55-98-1-0371, by the NSF under Grant Nos. CHE 96-16615 and CHE 97-32758. H.W. was supported by the Laboratory Directed Research and Development (LDRD) Project from the National Energy Research Scientific Computing (NERSC) Center, Lawrence Berkeley National Laboratory, and also by NSF Grant No. CHE 97-32758.

APPENDIX

Several efficient classical methods have been developed to calculate the thermal rate constant. We briefly summarize two of the more popular ones here.

For the thermal rate constant calculation, the flux operator set in Boltzmann form

$$\mathbf{F}(\beta) \equiv e^{-\beta \mathbf{H}/2} \mathbf{F} e^{-\beta \mathbf{H}/2}, \tag{A1}$$

is of low rank, and a Lanczos iteration [48,49] can be used to find the nonzero eigenvalues $\{f_m\}$ and the corresponding eigenvectors $\{|u_m\rangle\}$ [22–24]. Thus $\mathbf{F}(\beta)$ can be represented in its eigenstate expansion as follows:

$$\mathbf{F}(\beta) = \sum_m f_m |u_m\rangle\langle u_m|, \tag{A2}$$

and the trace in the flux correlation function expression becomes

$$C_f(t)=\sum_m f_m\langle u_m(t)|\mathbf{F}|u_m(t)\rangle, \tag{A3a}$$

where $\{|u_m(t)\rangle\}$ are the *time-evolved* eigenvectors of $\mathbf{F}(\beta)$:

$$|u_m(t)\rangle=e^{-i\mathbf{H}t/\hbar}|u_m\rangle. \tag{A3b}$$

The number of Lanczos iterations required for the convergence of this procedure is essentially the number of nonzero eigenvalues of $\mathbf{F}(\beta)$ and is usually several orders of magnitude smaller than the size of the full basis used in the calculation. Thus the trace calculation can be carried out much more efficiently than conventional methods.

Another way of obtaining the thermal rate constant $k(T)$ is through the Boltzmann integration of the cumulative reaction probability $N(E)$,

$$k(T)=[2\pi\hbar Q_r(T)]^{-1}\int_{-\infty}^{\infty}dE\, e^{-\beta E}N(E), \tag{A4}$$

where $N(E)$ is given by [25]

$$N(E)=\tfrac{1}{2}(2\pi\hbar)^2\,\mathrm{Tr}[\mathbf{F}\delta(E-\mathbf{H})\mathbf{F}\delta(E-\mathbf{H})]. \tag{A5}$$

The microcanonical density operator $\delta(E-\mathbf{H})$ can be related to the Green's function of Eq. (18) as

$$\delta(E-\mathbf{H})=-\frac{1}{\pi}\mathrm{Im}[\mathbf{G}(E)]. \tag{A6}$$

After some manipulations, $N(E)$ can be re-expressed in a somewhat different form [50]

$$\begin{aligned}N(E)&=\mathrm{Tr}[\mathbf{P}(E)]\\&=\mathrm{Tr}[4\epsilon_r^{1/2}\mathbf{G}(E)^{+}\epsilon_p\mathbf{G}(E)\epsilon_r^{1/2}],\end{aligned} \tag{A7}$$

where $\epsilon_r(\epsilon_p)$ denotes the absorbing boundary condition for reactant(product) region.

Equation (A7), though formally equivalent to Eq. (A5), offers a significant numerical improvement over the conventional methods due to the fact that the ''reaction probability operator'' $\mathbf{P}(E)$ is a low-rank Hermitian operator. Therefore, Hermitian Lanczos iteration method (here actually a special case of Arnoldi iteration method) [49] can be used to find the nonzero eigenvalues which then constitute the trace in Eq. (A7). During the Lanczos-Arnoldi iteration, the application of the Green's function can be achieved by either time-dependent or time-independent methods. If a time-independent method is used, as has been done previously [50], operating the Green's function on a trial vector corresponds to solving linear equations for a large system. Again, iterative methods for large (and sparse) linear systems, such as the generalized minimal residual [51] and quasiminimal residual [52] methods, can be used here to solve the problem efficiently.

[1] W. H. Miller, J. Phys. Chem. A **102**, 794 (1998).

[2] R. P. Feynman, Int. J. Theor. Phys. **21**, 467 (1982); P. Benioff, Phys. Rev. Lett. **48**, 1581 (1982); D. Deutsch, Proc. R. Soc. London, Ser. A **400**, 97 (1985).

[3] D. Deutsch, Proc. R. Soc. London, Ser. A **425**, 73 (1989).

[4] P.W. Shor, in *Proceedings of the 35th Annual Symposium on the Foundations of Computer Science*, edited by S. Goldwasser (IEEE Computer Society, Los Alamitos, CA, 1994), p. 124.

[5] P. W. Shor, SIAM J. Comput. **26**, 1484 (1997).

[6] A. Ekert and R. Jozsa, Rev. Mod. Phys. **68**, 733 (1996).

[7] L. K. Grover, *Proceedings of the 28th Annual ACM Symposium on the Theory of Computing* (ACM, New York, 1996), p. 212; L. K. Grover, Phys. Rev. Lett. **79**, 4709 (1997); **79**, 325 (1997); M. Boyer, G. Brassard, P. Hoyer, and A. Tapp, Fortschr. Phys.**46**, 493 (1998); E. Biham, O. Biham, D. Biron, M. Grassl, and D. A. Lidar, LANL e-print, quant-ph/9807027.

[8] S. Wiesner, LANL e-print, quant-ph/9603028.

[9] C. Zalka, Proc. R. Soc. London, Ser. A **454**, 313 (1998).

[10] S. Lloyd, Science **273**, 1073 (1996); B. M. Boghosian and W. Taylor, Phys. Rev. E **57**, 54 (1998); Int. J. Mod. Phys. C. **8**, 705 (1997); D. S. Abrams and S. Lloyd, Phys. Rev. Lett. **79**, 2586 (1997).

[11] R. Schack, Phys. Rev. A **57**, 1634 (1998).

[12] D. A. Lidar and O. Biham, Phys. Rev. E **56**, 3661 (1997).

[13] J. I. Cirac and P. Zoller, Phys. Rev. Lett. **74**, 4091 (1995); C. Monroe, D. M. Meekhof, B. E. King, W. M. Itano, and D. J. Wineland, *ibid.* **75**, 4714 (1995).

[14] Q. A. Turchette, C. J. Hood, W. Lange, H. Mabuchi, and H. J. Kimble, Phys. Rev. Lett. **75**, 4710 (1995).

[15] N. Gershenfeld and I. L. Chuang, Science **275**, 350 (1997); D. G. Cory, A. F. Fahmy, and T. F. Havel, Proc. Natl. Acad. Sci. USA **94**, 1634 (1997); I. L. Chuang, N. Gershenfeld, M. G. Kubinec, and D. W. Leung, Proc. R. Soc. London, Ser. A **454**, 447 (1998).

[16] I. L. Chuang, N. Gershenfeld, and M. Kubinec, Phys. Rev. Lett. **80**, 3408 (1998).

[17] D. Deutsch and R. Jozsa, Proc. R. Soc. London, Ser. A **439**, 553 (1992).

[18] I. L. Chuang, L. M. K. Vandersypen, X. Zhou, D. W. Leung, and S. Lloyd, Nature (London) **393**, 143 (1998).

[19] J. A. Jones, M. Mosca, and R. H. Hansen, Nature (London) **393**, 344 (1998); J. A. Jones and M. Mosca, J. Chem. Phys. **109**, 1648 (1998).

[20] J. A. Jones, Science **280**, 229 (1998).

[21] C. H. Bennett, Phys. Today **48** (10), 24 (1995); S. Lloyd, Sci. Am. **273**, 44 (1995); D. P. DiVincenzo, Science **270**, 255 (1995); A. M. Steane, Rep. Prog. Phys. **61**, 117 (1998); C. P. Williams and S. H. Clearwater, *Explorations in Quantum Computing* (Springer-Verlag, New York, 1998).

[22] W. H. Thompson and W. H. Miller, J. Chem. Phys. **106**, 142 (1997); **107**, 2164(E) (1997); W. H. Miller, in *Dynamics of Molecules and Chemical Reactions*, edited by Z. J. Zhang and R. Wyatt (Dekker, New York, 1996), p. 389; W. H. Miller, J. Phys. Chem. **102**, 793 (1998).

[23] T. J. Park and J. C. Light, J. Chem. Phys. **88**, 4897 (1988); D.

Brown and J. C. Light, *ibid.* **97**, 5465 (1992).

[24] U. Manthe, J. Chem. Phys. **102**, 9205 (1995); F. Matzkies and U. Manthe, *ibid.* **106**, 2647 (1997).

[25] W. H. Miller, J. Chem. Phys. **61**, 1823 (1974); W. H. Miller, S. D. Schwartz, and J. W. Tromp, *ibid.* **79**, 4889 (1983).

[26] T. Yamamoto, J. Chem. Phys. **33**, 281 (1960).

[27] G. D. Billing, Chem. Phys. Lett. **30**, 391 (1975); J. Chem. Phys. **99**, 5849 (1993); R. B. Gerber, V. Buch, and M. A. Ratner, *ibid.* **77**, 3022 (1982); V. Buch, R. B. Gerber, and M. A. Ratner, Chem. Phys. Lett. **101**, 44 (1983).

[28] Some recent examples include N. P. Blake and H. Metiu, J. Chem. Phys. **101**, 223 (1994); M. Ben-Nun and R. D. Levine, Chem. Phys. **201**, 163 (1995); Z. Li and R. B. Gerber, J. Chem. Phys. **102**, 4056 (1995); J. Cao, C. Minichino, and G. A. Voth, *ibid.* **103**, 1391 (1995); L. Liu and H. Guo, *ibid.* **103**, 7851 (1995); C. Scheurer and P. Saalfrank, *ibid.* **104**, 2869 (1995); J. Fang and C. C. Martens, *ibid.* **104**, 3684 (1996); S. Consta and R. Kapral, *ibid.* **104**, 4581 (1996); P. Bala, P. Grochowski, B. Lesyng, and J. A. McCammon, J. Phys. Chem. **100**, 2535 (1996); A. B. McCoy, J. Chem. Phys. **103**, 986 (1995); H. J. C. Berendsen and J. Mavri, Int. J. Quantum Chem. **57**, 975 (1996).

[29] W. H. Miller, J. Chem. Phys. **53**, 3578 (1970).

[30] M. F. Herman and E. Kluk, Chem. Phys. **91**, 27 (1984); E. Kluk, M. F. Herman, and H. L. Davis, J. Chem. Phys. **84**, 326 (1986); E. J. Heller, *ibid.* **94**, 2723 (1991); **95**, 9431 (1991); F. Grossman and E. J. Heller, Chem. Phys. Lett. **241**, 45 (1995); K. G. Kay, J. Chem. Phys. **100**, 4377 (1994); **100**, 4432 (1994); **101**, 2250 (1994).

[31] D. Provost and P. Brumer, Phys. Rev. Lett. **74**, 250 (1995); G. Campolieti and P. Brumer, Phys. Rev. A **50**, 997 (1994); S. Garashchuk and D. J. Tannor, Chem. Phys. Lett. **262**, 477 (1996); A. R. Walton and D. E. Manolopoulos, Mol. Phys. **87**, 961 (1996); A. R. Walton and D. E. Manolopoulos, Chem. Phys. Lett. **244**, 448 (1995); M. L. Brewer, J. S. Hulme, and D. E. Manolopoulos, J. Chem. Phys. **106**, 4832 (1997).

[32] W. H. Miller, J. Chem. Phys. **95**, 9428 (1991); B. W. Spath and W. H. Miller, *ibid.* **104**, 95 (1996); X. Sun and W. H. Miller, *ibid.* **106**, 916 (1997); **106**, 6346 (1997); J. Chem. Phys. **108**, 8870 (1998).

[33] H. Wang, X. Sun, and W. H. Miller, J. Chem. Phys. **108**, 9726 (1998).

[34] D. P. DiVincenzo, Phys. Rev. A **51**, 1015 (1995); A. Barenco, D. Deutsch, A. Ekert, and R. Jozsa, Phys. Rev. Lett. **74**, 4083 (1995); T. Sleator and H. Weinfurter, *ibid.* **74**, 4087 (1995); S. Lloyd, *ibid.* **75**, 346 (1995).

[35] J. von Neumann, *Mathematical Foundations of Quantum Mechanics* (Princeton University Press, Princeton, 1955).

[36] A. Fijany and C. P. Williams, LANL e-print quant-ph/9809004.

[37] D. Kosloff and R. Kosloff, J. Comput. Phys. **52**, 35 (1983); R. Kosloff, J. Phys. C **92**, 2087 (1988).

[38] M. D. Feit, J. A. Fleck, and A. Steiger, J. Comput. Phys. **47**, 412 (1982).

[39] V. Vedral, A. Barenco, and A. Ekert, Phys. Rev. A **54**, 147 (1996), LANL e-print quant-ph/9511018; D. Beckman, A. N. Chari, S. Devabhaktuni, and J. Preskill, Phys. Rev. A **54**, 1034 (1996).

[40] R. Cleve, A. Ekert, C. Macchiavello, and M. Mosca, Proc. R. Soc. London, Ser. A **454**, 339 (1998).

[41] D. Coppersmith (unpublished); R. B. Griffiths and C.-S. Niu, Phys. Rev. Lett. **76**, 3228 (1996).

[42] P. Hoyer, LANL e-print quant-ph/9702028; R. Jozsa, Proc. R. Soc. London Ser. A **454**, 323 (1998); A. Ekert and R. Jozsa, Philos. Trans. R. Soc. London Ser. A **356**, 1765 (1998).

[43] W. G. Unruh, Phys. Rev. A **51**, 992 (1995); S. Haroche and J. M. Raimond, Phys. Today **49** (8), 51 (1996); W. S. Warren, Science **277**, 1688 (1997).

[44] A. R. Calderbank and P. W. Shor, Phys. Rev. A **54**, 1098 (1996); A. M. Steane, Phys. Rev. Lett. **77**, 793 (1996); D. Gottesman, Phys. Rev. A **54**, 1862 (1996).

[45] P. Zanardi and M. Rasetti, Mod. Phys. Lett. B **11**, 1085 (1997); D. A. Lidar, I. L. Chuang, and K. B. Whaley, Phys. Rev. Lett. **81**, 2594 (1998).

[46] P. W. Shor, in *Proceedings of the 37th Conference on Foundations of Computer Science* (IEEE, Los Alamitos, CA, 1996); D. Gottesman, Phys. Rev. A **57**, 127 (1997); J. Preskill, Proc. R. Soc. London, Ser. A **454**, 385 (1998); R. Laflamme, E. Knill and W. Zurek, Science **279**, 342 (1998).

[47] C. Miquel, J. P. Paz, and W. H. Zurek, Phys. Rev. Lett. **78**, 3971 (1997).

[48] C. Lanczos, J. Res. Natl. Bur. Stand. **45**, 255 (1950).

[49] Y. Saad, *Numerical Methods for Large Eigenvalue Problems: Theory and Algorithms* (Wiley, New York, 1992).

[50] U. Manthe and W. H. Miller, J. Chem. Phys. **99**, 3411 (1993).

[51] Y. Saad and M. H. Schultz, SIAM (Soc. Ind. Appl. Math.) J. Sci. Stat. Comput. **7**, 856 (1986).

[52] R. W. Freund, SIAM (Soc. Ind. Appl. Math.) J. Sci. Stat. Comput. **13**, 425 (1992).