Quantum computing of semiclassical formulas

B. Georgeot and O. Giraud

Laboratoire de Physique Théorique, Université de Toulouse, UPS, CNRS, 31062 Toulouse, France
(Received 30 January 2008; published 25 April 2008)

We show that semiclassical formulas such as the Gutzwiller trace formula can be implemented on a quantum computer more efficiently than on a classical device. We give explicit quantum algorithms which yield quantum observables from classical trajectories, and which alternatively test the semiclassical approximation by computing classical actions from quantum evolution. The gain over classical computation is in general quadratic, and can be larger in some specific cases.

I. INTRODUCTION

It is now widely recognized that the principles of quantum mechanics allow realization of new computational devices which can be more efficient than their classical counterparts [1–4]. Quantum algorithms have been proposed which take advantage of the quantum mechanical properties of these devices to perform specific tasks faster than a classical computer. The most famous such algorithm is due to Shor [5] and factors large integers exponentially faster than any known classical algorithm. Another algorithm, for which the gain is only quadratic, enables the search of an unsorted database [6]. Efforts have been devoted also to using such quantum computers to simulate the behavior of complex physical systems, a task of much practical interest. Algorithms have been set up enabling simulation of certain quantum mechanical systems efficiently [7–11], as was originally envisioned by Feynman. However, as quantum algorithms use procedures different from classical algorithms, it is by no means obvious which problems can be sped up by using a quantum computer. It is therefore important to precisely specify the class of problems that can be solved efficiently on a quantum computer, especially among problems that have been implemented by scientists on classical devices because of their practical interest.

On classical computers, a great deal of activity in the past decades has been devoted to the numerical implementation of *semiclassical formulas*. Such formulas approximate quantum mechanics through classical quantities, and have been used since the beginning of quantum mechanics. Although they have been much studied, their application to practical computation of quantum observables is often hampered by the exponential proliferation of classical orbits involved when the system is chaotic. Semiclassical formulas enable the approximation of the exact quantum mechanics for small $\hbar$, and give an insight into the relationship between classical and quantum mechanics. For integrable systems with n degrees of freedom, classical dynamics takes place on n-dimensional tori in the $2n$-dimensional phase space. In this case, semiclassical formulas quantize individual tori. They are relatively straightforward to implement and were constructed and used early in the development of quantum mechanics. In contrast, for chaotic systems this quantization of tori is not valid, as pointed out by Einstein as early as in 1917 [12], and individual wave functions cannot be built

from a single classical structure. As a substitute, various formulas have been constructed, which express the quantum quantities in terms of an (infinite) set of classical orbits. The most famous such formula is the *Gutzwiller trace formula* [13], where the quantum density of states $d(E) = \sum_n \delta(E - E_n)$ (where E_n are the energy levels) is written as a function of all classical periodic orbits of the system. It has the general form $d(E) \equiv \sum_p A_p e^{i\varphi_p/\hbar}$, where the sum runs over all periodic orbits, A_p is related to the stability of the orbit, and φ_p to its action. It can be viewed as a Fourier-type duality between the set of all eigenenergies of the system on the one hand and the set of all actions of periodic orbits on the other hand. Other formulas of the same kind give the quantum propagator $G(x, x')$ in terms of all classical orbits from x to x' (Van Vleck formula) [14] or scattering amplitudes in terms of scattering orbits (Miller's formula) [15]. Many works have implemented such formulas numerically by truncating the sum over classical orbits (see, e.g., [16–21]), e.g., to obtain the semiclassical spectrum, but because of the exponential proliferation of classical orbits typical of chaotic systems only a few semiclassical eigenvalues can be extracted. Several methods have been devised to reduce the number of orbits entering the sum [22–24], but they all require summing up contributions from a still exponential number of orbits.

In this paper, we study the implementation of semiclassical formulas on quantum computers. We show that, for certain dynamical systems, such formulas can be computed more efficiently on a quantum computer than on a classical device. From the quantum information point of view, this gives additional examples of algorithms where a gain can be reached compared to classical algorithms. From the point of view of quantum chaos, this would enable these formulas to become more practical on a quantum computer if such a device becomes available, and thus to explore the quantum-classical correspondence in regimes that are difficult to reach on a classical computer. The paper is organized as follows. In Sec. II, we present in detail the most famous semiclassical formula which relates the density of states to classical periodic orbits (the Gutzwiller trace formula), in the specific case of quantum maps. We then discuss in Sec. III a quantum algorithm that implements this semiclassical formula in the form where it is most difficult classically, i.e., in summing up classical orbits and extracting quantum observables. In Sec. IV we implement the same formula but in the reverse direction, i.e., using quantum observables to extract classical

quantities. Our method can be considered as a new way of extracting information from the quantum simulation of quantum systems. Indeed, while many quantum systems can be simulated efficiently on a quantum computer, a crucial point to get a complete algorithm and make the gain effective is to devise a readout method once the simulation is performed. It has been shown that the gain over classical computation can depend critically on the observable measured at the end of the simulation [25–27]. In the present paper we show that in general we can expect a quadratic gain over classical computation using the algorithms of Secs. III and IV, and that this gain can be quartic for some quantities. The original hope of this study was to use the quantum Fourier transform, which is exponentially faster than the classical Fourier transform, to ensure an exponential gain for this type of problem. It turned out that for most systems a count of the total number of gates involved shows that only a polynomial gain can be reached. However, in Sec. V, we give an example of a related problem where exponential gain can be reached.

II. SEMICLASSICAL TRACE FORMULA FOR QUANTUM MAPS

Classical and quantum maps represent a particularly simple class of dynamical systems. Indeed, such systems, where one iteration of the map corresponds to a discrete time step, are easier to handle and yield simpler formulas. In what follows, we will restrict ourselves to such systems. This does not entail a major loss of generality, since it is known that Hamiltonian systems can in general be represented by maps through the construction of Poincaré surfaces of section [28,29]. Furthermore, most of the phenomena observed in more complicated systems can be reproduced in well-known models of quantum maps. This explains why many works on semiclassical formulas have used classical and quantum maps as test beds.

Here we consider two-dimensional maps on a toroidal phase space. Let us first give examples of well-known classical maps that we will use later on. A much studied instance is the family of cat maps [28,30–32], i.e., linear automorphisms of the torus characterized by 2×2 matrices of $SL(2, \mathbb{Z})$. For a matrix $M = \begin{pmatrix} t_{11} & t_{12} \\ t_{21} & t_{22} \end{pmatrix}$, the corresponding map is

$$\bar{p} = t_{11}p + t_{12}q \pmod 1,$$

$$\bar{q} = t_{21}p + t_{22}q \pmod 1, \tag{1}$$

where (p,q) are phase-space variables and the overbars denote new variables after one iteration of the map.

Another well-known example is the baker's map [28]:

$$(\bar{q}, \bar{p}) = \begin{cases} \left(2q, \dfrac{p}{2}\right) & \text{for } 0 \le q \le \dfrac{1}{2}, \\[2mm] \left(2q-1, \dfrac{p+1}{2}\right) & \text{for } \dfrac{1}{2} < q \le 1. \end{cases} \tag{2}$$

Maps (1) [for matrices M with $|\text{Tr}(M)| > 2$] and (2) are instances of strongly chaotic systems, with homogeneous ex-

ponential divergence of trajectories, positive Kolmogorov-Sinai entropy, and exponential proliferation of periodic orbits with the length.

More generally, many classical maps can be written in the form

$$\bar{p} = p - kV'(q),$$

$$\bar{q} = q + T\bar{p}, \tag{3}$$

where the potential $V(q)$ is a function of position. Such maps correspond to the integration over one period of a free rotator periodically kicked by a potential $V(q)$. They include the standard map (the classical version of the kicked rotator) [33] for $V(q) = \cos q$, or the sawtooth map [10] for $V(q) = -(p-\pi)^2/2$. These maps display a wide range of different behaviors depending on the parameters. In particular, for the standard map the dynamics changes from being close to integrability for small values of the parameter kT to fully developed chaos for large values of kT.

The quantum version of the classical maps acts on a Hilbert space of dimension N corresponding to the inverse of Planck's constant $2\pi\hbar$. It is represented by an $N \times N$ matrix U [31]. In the case of a cat map (1), the quantization yields [31,32]

$$U_{Q_1, Q_2} = \sqrt{\frac{it_{12}}{N}} \langle e^{2i\pi NS(Q_1/N, Q_2/N+m)} \rangle_m, \tag{4}$$

where $S(q_1, q_2) = (t_{11}q_1^2 - 2q_1q_2 + t_{22}q_2^2)/(2t_{12})$ and the average is taken over all integers m.

The quantized baker's map [34] is even simpler. The evolution operator on an N-dimensional space is given by

$$F_n^{-1} \begin{pmatrix} F_{n-1} & 0 \\ 0 & F_{n-1} \end{pmatrix}, \tag{5}$$

where F_n is the $N \times N$ matrix with $(F_n)_{kj} = \frac{1}{\sqrt{N}} e^{-2i\pi kj/N}$ (discrete Fourier transform).

Finally, maps of the form (3) yield, upon quantization, quantum maps of the form

$$\hat{U} = e^{-iT\hat{p}^2/2\hbar} e^{-ikV(\hat{q})/\hbar}. \tag{6}$$

These evolution operators can be implemented efficiently on a quantum computer. This was shown for (5) in [35] using the quantum Fourier transform instead of the classical one, and in [9,10] for maps of the form (6).

One of the advantages of maps over generic systems is that some of the steps leading to the trace formula linking the spectrum to periodic orbits can be made exact. Indeed, the spectral density for an $N \times N$ quantum map U with eigenphases θ_k, $1 \le k \le N$, is given by

$$d(\theta) \equiv \sum_{m=-\infty}^{\infty} \sum_{k=1}^{N} \delta(\theta - \theta_k + 2\pi m)$$

$$= \frac{N}{2\pi} + \frac{1}{2\pi} \sum_{t=1}^{\infty} (e^{-it\theta} \text{tr } U^t + e^{it\theta} \text{tr } U^{-t}). \tag{7}$$

This expression, obtained by the Poisson summation for-

mula, is exact and depends only on the traces of iterates of the quantum map. Similarly, one can express the coefficients of the characteristic polynomial $\det(I-xU)=\Sigma_k \beta_k x^k$ only in terms of traces of powers of U by using the recurrence relation

$$\beta_k = -\frac{1}{k}\sum_{t=1}^{k} \beta_{k-t}\mathrm{tr}\, U^t, \quad \beta_0 = 1. \tag{8}$$

This relation can be easily proved by expanding $\det(I+zU) = \exp \mathrm{tr}\, \ln(I+zU)$ into powers of z. Unitarity of the operator U implies the symmetry relation

$$\beta_{N-k} = \det(-U)\bar{\beta}_k. \tag{9}$$

Thanks to this resurgence relation the computation of $\mathrm{tr}\, U^t$ for $t \le N/2$ suffices to calculate the characteristic polynomial.

The semiclassical approximation of the spectrum can be obtained by calculating the coefficients of the characteristic polynomial (8) using semiclassical expressions for the traces. For large N the main contribution to $\mathrm{tr}\, U^t$ comes from periodic orbits. For a classical map ϕ mapping the phase space onto itself, a periodic orbit of length t is a fixed point of ϕ^t. It is given by a sequence $(p_0, q_0, p_1, q_1, \ldots, p_t, q_t)$ of phase-space points such that $(p_i, q_i) = \phi(p_{i-1}, q_{i-1})$ for all $i, 1 \le i \le t$, and $(p_t, q_t) = (p_0, q_0)$. If t_p is the smallest integer such that $(p_{t_p}, q_{t_p}) = (p_0, q_0)$, then t_p divides t and the periodic orbit is the repetition of $r = t/t_p$ times a primitive periodic orbit. A given primitive periodic orbit is characterized by its monodromy matrix M_p (which is the linearized version of the map ϕ in the vicinity of the periodic orbit), its action $S_p = \Sigma_{i=1}^{t_p} S(q_{i-1}, q_i)$ where $S(q, q')$ is the classical action from q to q', and its Maslov index ν_p. The semiclassical expansion of $\mathrm{tr}\, U^t$ reads

$$\mathrm{tr}\, U^t \approx \tau_t = \sum_{p \in \mathcal{P}_t} \frac{t_p e^{ir(S_p/\hbar - \nu_p \pi/2)}}{|\det(I - M_p^r)|^{1/2}}, \tag{10}$$

where the sum runs over the set $\mathcal{P}_t$ of all periodic orbits of length (number of time steps) $t = rt_p$. The action, Maslov index, and monodromy matrix correspond to the associated primitive periodic orbit [36].

For maps (1)–(3), the classical dynamics displays some form of chaos, up to the strongest types with exponential divergence of nearby trajectories and exponential proliferation of periodic orbits with increasing length. Such properties make difficult the practical use of semiclassical formulas, which need enormous numbers of orbits to be accurate. As we will show in the next section, this task can be made easier on a quantum computer. The fact already mentioned that the quantum evolution operator of these maps can often be implemented efficiently on a quantum computer opens the way to the use of semiclassical formulas in the reverse direction, using quantum observables to infer results on classical quantities. This will be the subject of Sec. IV.

III. SPECTRUM FROM CLASSICAL QUANTITIES

We first discuss an algorithm allowing us to obtain semi-classically the set of eigenvalues of the quantum map, or

equivalently the coefficients (8) of the characteristic polynomial of the map.

In order to calculate the traces using (10) we need to be able to characterize periodic orbits of the classical map. There are instances of systems where this task is very easy. For instance for cat maps (1), the iterates of the classical map can be calculated analytically, and therefore periodic orbits are entirely characterized. This is also the case for perturbed cat maps that are Anosov maps of the form $\phi = \phi_0 \circ \chi_\epsilon$, where ϕ_0 is a cat map and χ_ϵ is a perturbation close to the identity. It was shown [37] that for sufficiently weak perturbations orbits of Anosov maps remain topologically conjugate to periodic orbits of the unperturbed cat map. Thus periodic orbits can be described completely [38]. More generically, we will consider systems in which periodic orbits can be described by a symbolic dynamics associated with a finite Markov partition. That is, phase space can be partitioned into sets R_k, $1 \le k \le m$, and intersections of the images of the R_k under the (forward and backward) iterates of the classical map define finer and finer partitions, so that at infinity the intersections contain either no point or a single one [39]. Thus a given (infinite) sequence of labels corresponds to at most one point of phase space. The mapping rules between the R_k under one iteration of the map can be summarized in an $m \times m$ transition matrix T such that $T_{ij} = 1$ if the image of R_i has a non-empty intersection with R_j, and 0 otherwise. This transition matrix sums up the grammar rules that discriminate between allowed words and forbidden ones. There is a one-to-one correspondence between phase-space points and allowed symbolic sequences, and periodic orbits correspond to periodic sequences of symbols.

Simple examples of quantum maps with symbolic dynamics are perturbed cat maps or the baker's map [40]. In the latter example, symbolic dynamics is described by only two symbols 0 and 1, and all sequences of symbols are allowed. Dynamical systems such as the three-disk [41] or motion on surfaces with constant negative curvature [42] also provide examples where a symbolic dynamics exists with all sequences allowed. In such examples, periodic trajectories are in one-to-one correspondence with periodic strings of 0 and 1.

We will now sketch the steps of a quantum algorithm allowing us to compute the semiclassical traces (10) in a parallel way. To simplify notation, for each periodic orbit p of length $t = rt_p$ we define the amplitude $\mathcal{A}_p = t_p/|\det(I - M_p^r)|^{1/2}$ and the phase $\phi_p = r(S_p/\hbar - \nu_p \pi/2)$. Thus we have to calculate the quantities $\tau_t = \Sigma_p \mathcal{A}_p e^{i\phi_p}$. Let us consider a system whose symbolic dynamics is described by a finite Markov partition. For simplicity we assume that the partition consists of only two sets. Then only two symbols 0 and 1 are required (if there are more than two sets in the partition we code labels by binary strings). The trace formula (10) will be truncated at $t_{\max}$, which means that only periodic orbits of length $t < t_{\max}$ will be considered. We distinguish five registers in the computational state. Register A will hold the lengths t, $0 \le t < t_{\max}$, of the periodic orbits. It requires $n_A = \log_2(t_{\max})$ qubits. Register B will hold, on its last t qubits, the 2^t code words corresponding to a given orbit length t. This register has to contain $n_B = t_{\max}$ qubits. Register C is used to store the phases ϕ_p, and register D is used for the
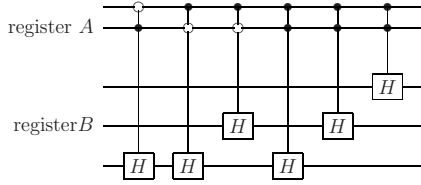
FIG. 1. Circuit for step II and two-letter symbolic dynamics. Register A codes for lengths t, $0 \le t \le 3$, on two qubits. The Hadamard gates are controlled by the values of t, and on register B the state $|t\rangle|0\rangle$ becomes $2^{-t/2}\sum_{i=0}^{2^t-1}|t\rangle|i\rangle$.

"tuning" of the amplitudes $\mathcal{A}_p$ associated with each orbit. Additional registers will serve as work space. We will make use of the following one-qubit operations: rotation of the kth qubit $R_k(\theta) = \exp(-i\theta\sigma_y^{(k)})$ and phase shifts $P_k(\theta) = \exp(-i\theta\sigma_z^{(k)})$. The steps are as follows.

Step I. Let λ be such that the number of allowed code words scales exponentially with t as $\exp(\lambda t)$. We define Λ such that, for each t, the amplitude $\mathcal{A}_p = t_p / |\det(I - M_p^r)|^{1/2}$ of each periodic orbit p of length $t = r t_p$ is upper bounded by $\exp(-\Lambda t)$. The amplitude $\mathcal{A}_p$ in many cases will be actually close to $\exp(-\Lambda t)$. Let us set $\mu = \Lambda - \lambda/2$. We define angles $\theta_k \in [0, \pi/2]$ by $\cos\theta_k = 1/\sqrt{1 + e^{-2\mu2^k}}$. Applying n_A rotations $R_k(\theta_k)$ to register A of the initial state gives (up to a normalization factor) the state $\sum_{t < t_{\max}} \exp(-\mu t)|t\rangle_A|0\rangle_B|0\rangle_C|0\rangle_D$.

Step II. All allowed code words are generated on register B. In the simplest case where there is no grammar rule one wants, for each value t on register A, to put the last t qubits of B into a uniform superposition. This is performed by applying, for each t, t Hadamard gates controlled by register A on the last t qubits of B (see Fig. 1). This gives (up to normalization) the state

$$\sum_t \sum_p e^{-\Lambda t}|t\rangle_A|p\rangle_B|0\rangle_C|0\rangle_D, \qquad (11)$$

where the second sum runs over all code words p, $0 \le p \le 2^t - 1$. If there is a finite number of grammar rules, the allowed code words can be generated by replacing the Hadamard gates by rotations R_k. For each value of t these rotations are controlled not only by register A (as in Fig. 1) but also by qubits of register B. Steps I and II are polynomial in $t_{\max}$.

Step III. The values of ϕ_p and $\ln \mathcal{A}_p + \Lambda t$ are calculated and written on registers C and D. Indeed, from each code word p of length t it is possible to recover the phase-space coordinates $(p_0, q_0, p_1, q_1, \ldots, p_t, q_t)$ of the trajectory coded by this code word, as well as the characteristics of this trajectory: action S_p, monodromy matrix M_p, and Maslov index ν_p. In many systems this can be done by a sequence of elementary operations, which just correspond to the sequence of operations that would be performed on a classical computer. This is clearly the case for all systems where the periodic orbit sum was numerically simulated in the literature [16–21], although there is no general procedure. Such elementary operations are based ultimately on the four arithmetic operations, whose classical complexity is polynomial in the number of bits of the input. It is known (see, e.g., [4],

Chaps. 1 and 3) that a classical process $x \to f(x)$ can be transformed in a quantum routine parallelized on all x as $\sum_x|x\rangle|0\rangle \to \sum_x|x\rangle|f(x)\rangle$ with comparable efficiency. In general, for the kind of systems considered here, this step is thus polynomial in $t_{\max}$. After erasing intermediate steps we get a state

$$\sum_t \sum_{p \in \mathcal{P}_t} e^{-\Lambda t}|t\rangle_A|p\rangle_B|\phi_p\rangle_C|\ln \mathcal{A}_p + \Lambda t\rangle_D. \qquad (12)$$

Step IV. As in step I we use rotations $R_k(\theta_k)$ to transfer the value stored in register D into an exponential prefactor $\exp(\ln \mathcal{A}_p + \Lambda t)$. The angles θ_k are now given by $\cos\theta_k = 1/\sqrt{1 + e^{-2\kappa2^k}}$, where the constant κ sets the precision that one wants to achieve on the prefactor. For each value of t and p, register D is the sum of two orthogonal components $|0\rangle_D$ and $|\psi_p\rangle_D$. As the prefactor $e^{-\Lambda t}$ in (12) is meant to yield a rough estimate of the amplitudes $\mathcal{A}_p$, it can be expected that the quantities $\ln \mathcal{A}_p + \Lambda t$ are small and that the relative weight of $\langle\psi_p|\psi_p\rangle$ is small.

By controlled phase shifts on register C, the states are then multiplied by the phase factor $e^{i\phi_p}$, and step III is run backward to erase register C. This yields

$$\sum_t \sum_{p \in \mathcal{P}_t} \mathcal{A}_p e^{i\phi_p}|t\rangle_A|p\rangle_B|0\rangle_C(|0\rangle_D + |\psi_p\rangle_D). \qquad (13)$$

Step V. In order to get the semiclassical traces $\tau_t = \sum_p \mathcal{A}_p e^{i\phi_p}$, we perform $t_{\max}$ quantum Fourier transforms (QFTs) on register B. Each QFT corresponds to a given value of t and operates on the last t qubits of register B. That is, the gates of the QFT are controlled by register A (as in step II; see Fig. 1). This yields a state

$$\sum_t \sum_{k=0}^{2^t-1} \sum_{p \in \mathcal{P}_t} \mathcal{A}_p e^{i\phi_p} e^{-2i\pi kp/2^t}|t\rangle_A|k\rangle_B|0\rangle_C(|0\rangle_D + |\psi_p\rangle_D).$$
$$(14)$$

Step VI. In (14) the amplitude of the $|k=0\rangle_B$ term corresponds to the semiclassical traces τ_t. Therefore we now just have to perform a quantum search of $|0\rangle_B|0\rangle_D$ in (14). This is done by amplitude amplification performed on registers B and D. This process, which is the slowest part of our algorithm, requires $O(2^{t_{\max}/2})$ operations (as $\langle\psi_p|\psi_p\rangle$ is small, the search on register D is expected to contribute only a prefactor). It brings the state (14) into a state

$$\sum_t \sum_{p \in \mathcal{P}_t} \mathcal{A}_p e^{i\phi_p}|t\rangle_A|0\rangle_B|0\rangle_C|0\rangle_D = \sum_t \tau_t|t\rangle_A|0\rangle_B|0\rangle_C|0\rangle_D.$$
$$(15)$$

Quantum state tomography then gives the relative values of all semiclassical traces τ_t. The knowledge of τ_1 (easily computed classically) allows us to obtain the absolute values of the τ_t, and thus the characteristic polynomial. Because of the symmetry relation (9), only traces up to $t_{\max} = N/2$ are required. Therefore the cost of our quantum algorithm (which is essentially the cost of amplitude amplification in step VI) is $O(2^{N/4})$. This is to be compared with the classical cost of $O(2^{N/2})$ required for the calculation of the semiclassical characteristic polynomial.

As already mentioned our algorithm aims at estimating the accuracy of the semiclassical approximation. Obviously, the cost of calculating the exact characteristic polynomial, with a scaling in $O(N^3)$, is far less. Thus for systems where the trace formula is exact, the result of the semiclassical sum should only yield with much more efforts the same result as the exact diagonalization. For instance, for cat maps the exact equality tr $U^t = \tau_t$ holds in Eq. (10), and thus cat maps are not suited to studying discrepancies between exact and semiclassical energy levels if the full semiclassical sum is used. There are, however, instances of systems for which characterization of classical periodic orbits remains easy while the traces obtained through the trace formula (10) are truly approximations, such as, e.g., the perturbed cat maps described above. In such cases, our algorithm yields the semiclassical spectrum with quadratic efficiency compared to classical computation. In addition, even when the trace formula is exact, its truncation is not, and therefore its implementation has some interest and has been done classically in [16,17,19]. Indeed, it enables the convergence properties of the sum over periodic orbits in (7) to be understood.

IV. CLASSICAL ORBITS FROM QUANTUM OPERATOR

Another way of estimating the accuracy of the semiclassical approximation is to calculate how well the classical actions of the periodic orbits are reproduced when calculated from the spectrum through the trace formula. In the semiclassical approximation, the trace of the iterates of the quantum operator can be written as a sum over periodic orbits. This sum can be put in the form $\tau_t = \sum_p \mathcal{A}_p e^{2i\pi NS_p} \approx$ tr U^t [see Eq. (10)]. The actions S_p calculated from the quantum spectrum through the semiclassical formula (10) are obtained by performing a fast Fourier transform (FFT) on the set of traces tr U^t calculated for all matrix sizes $0 \leq N < N_{max}$. The number of traces N_{max} to evaluate depends on the precision required for the actions.

We now discuss a quantum algorithm allowing us to calculate each trace tr U^t, for any matrix size $0 \leq N < N_{max}$. Let m be the smallest integer such that $N \leq 2^m$ and $M = 2^m$. We distinguish three registers in the state vector on which the computation is performed. Register A stores the lengths t of the orbits, $0 \leq t < t_{max}$, on $n_A = \log_2 t_{max}$ qubits; here t_{max} is some fixed integer specifying the highest period that we want to consider. The two other registers, each of length m, will store the computational basis vectors. Additional work-space registers will be used as well in the course of the computations. Starting from the state $|0\rangle_A|0\rangle_B|0\rangle_C$, we perform the following steps.

Step I. We first apply Hadamard gates on registers A, B, and C to put them in an equal superposition of basis vectors. We obtain

$$\sum_t \sum_{i=0}^{M-1} |t\rangle_A |i\rangle_B |i\rangle_C. \tag{16}$$

What we want is in fact a sum running over a range $0 \leq i \leq N-1$. To obtain this from (16) we use an auxillary qubit (register D) that is set to $|0\rangle$ if $i - N < 0$ and to $|1\rangle$ if $i - N \geq 0$. The relative weight of the state,

$$\sum_t \sum_{i=0}^{N-1} |t\rangle_A |i\rangle_B |i\rangle_C |0\rangle_D, \tag{17}$$

is greater than 1/2.

Step II. The $N \times N$ matrix U^t has to be applied to register B of each state $|t\rangle_A |i\rangle_B |i\rangle_C$. As an illustration we focus on operators of the type (6). It was shown in [9] that for such maps one iteration can be implemented efficiently for a fixed matrix of size a power of 2. The algorithm consists in using QFTs to shift back and forth between p and q representations, while the operators $e^{if(\hat{p})}$ and $e^{iV(\hat{q})}$ are applied in the basis where they are diagonal by multiplication of basis vectors by a phase. For $N \neq 2^m$ the simulation of the quantum map involves a QFT on vectors of size not a power of 2. Such a procedure was proposed in [43] for any fixed vector size N. The simulation of U can therefore be done efficiently. The simulation of U^t can be done sequentially, controlled by the qubits of register A (as in Fig. 1).

Step III. The state (17) is now transformed into

$$\sum_t \sum_i^{N-1} |t\rangle_A (U^t|i\rangle_B)|i\rangle_C |0\rangle_D = \sum_t \sum_{i,j=0}^{N-1} U_{j,i}^t |t\rangle_A |j\rangle_B |i\rangle_C |0\rangle_D. \tag{18}$$

By amplitude amplification on registers B, C, and D, we select vectors with $|j\rangle_B = |i\rangle_C$ and $|0\rangle_D$, leading to

$$\sum_t \sum_{i=0}^{N-1} U_{i,i}^t |t\rangle_A |i\rangle_B |i\rangle_C |0\rangle_D. \tag{19}$$

After erasing register C we perform a QFT on register B. As in Sec. III, we use amplitude amplification to select the state $|0\rangle_B$, whose amplitude is $\sum_i U_{i,i}^t / \sqrt{M} =$ tr $U^t / \sqrt{M}$. This is the slowest step in our computation. For chaotic systems the matrix elements $U_{i,j}^t$ for $N \times N$ matrices are of order $1/\sqrt{N}$ and the traces tr U^t are expected to be of order 1. Thus each amplitude amplification has a cost $O(\sqrt{N})$ and step III requires N Grover iterations in total. For integrable systems the traces are of order $\sqrt{N}$, and therefore only one of the amplitude amplifications is needed, requiring $\sqrt{N}$ Grover iterations in total for step III.

Step IV. We are now in the state

$$\sum_t \text{tr } U^t |t\rangle_A |0\rangle_B |0\rangle_C |0\rangle_D. \tag{20}$$

The relative values of the traces for different values of t are obtained by quantum state tomography. The traces themselves are then deduced from the classical calculation of tr U, requiring $O(N)$ classical operations.

The algorithm requires the calculation of N_{max} traces tr U^t, with $0 \leq N < N_{max}$. Thus the cost of the quantum algorithm is of order N_{max}^2. Classically, we need to compute all N_{max} traces. Except for tr U_N this would need $O(N^2)$ classical operations if the map is of the type (6), and up to $O(N^3)$ in the general case where diagonalization of the operator is required. Thus the classical cost is of order N_{max}^3 or N_{max}^4 operations. Thus in both cases the quantum algorithm outperforms classical computation, albeit polynomially.

We note that if one is interested in distinguishing integrable and chaotic systems via the form factor as in the algorithm proposed in [26], then one needs only to be able to distinguish traces of order $\sqrt{N}$ (integrable case) from those of order 1 (chaotic case), for a specific value of N. In this case, one can stop at step III and check that $\sqrt{N}$ Grover iterations are enough to get to the state $|0\rangle$, in which case one concludes that the system is integrable, or not enough, in which case one concludes that the system is chaotic. Our algorithm then only needs $O(\sqrt{N})$ quantum operations instead of $O(N^2)$ classical operations, an improvement from the quadratic gain in [26]. One can also compute exactly the trace (stopping at step IV), and compute the form factor for small t, with a quadratic improvement compared to classical computation.

V. EXPONENTIAL SPEEDUP BY PHASE ESTIMATION

The preceding processes can be applied to many physical systems and yield a polynomial speedup compared to classical computation. However, there exist systems where a larger (up to exponential) gain might be obtained, following a different strategy based on phase estimation. This method [8,43] enables one to obtain an eigenvalue of a given operator U by applying conditionally iterates of U to an eigenvector $|\Psi\rangle$; this gives the state $\Sigma_i |i\rangle U^i |\Psi\rangle$ which, by Fourier transform on the first register, gives $|\theta\rangle|\Psi\rangle$, where $\exp(i\theta)$ is the eigenvalue corresponding to $|\Psi\rangle$. If $|\Psi\rangle$ is not an eigenvector but some randomly chosen state, the same process leads to $\Sigma_j \alpha_j |\theta_j\rangle|\Psi_j\rangle$, where $|\Psi_j\rangle$ are eigenvectors and $|\Psi\rangle = \Sigma_j \alpha_j |\Psi_j\rangle$. To be efficient, this method critically requires not only that U can be efficiently implemented, but also that exponential iterates of U can be implemented with a polynomial number of quantum gates, a much more stringent requirement. In the case of the quantum cat map, this method is efficient and remarkably enough can lead to classical quantities with exponential efficiency.

It is known [31] that the nth iterate of the quantized cat map (4) coincides with the quantization of the classical nth iterate. In [44], it was shown that one can simulate the classical cat map efficiently on a quantum computer, while in [45], it was further shown that one can compute the classical nth iterate for exponentially large n with a polynomial number of gates. Thus if one starts from a random vector $|\Psi\rangle$, one can compute $\Sigma_i |i\rangle U^i |\Psi\rangle$ in a polynomial number of gates for exponential i's and N; a quantum Fourier transform followed by a quantum measurement leads to the value $|\theta_j\rangle$ of one eigenvalue of the quantum cat map. It is known [32] that these eigenvalues are very constrained, being of the form

$$\theta_j = \frac{2\pi j + \phi(N)}{n(N)}, \tag{21}$$

where $n(N)$ is the quantum period function, which is the smallest integer such that

$$U^{n(N)} = I e^{i\phi(N)}, \tag{22}$$

and the phase $\phi(N)$ can be calculated easily from the components of matrix L [32].

Thus using this algorithm the quantum period function can be obtained in polynomial time on a quantum computer.

This quantity is related to the classical period function, which for each matrix L is the shortest integer g such that $L^g = I \mod N$. Indeed, the quantum period $n(N)$ is also the smallest integer such that $L^{n(N)} = I \pmod{N}$ if N is odd, and such that

$$L^{n(N)} = \begin{pmatrix} 1 \pmod{N} & 0 \pmod{2N} \\ 0 \pmod{2N} & 1 \pmod{N} \end{pmatrix}$$

if N is even. The two functions in all cases differ by at most a factor of 2 [32], so knowing one of them enables one to test and easily find the other one. The classical period function describes the periodic orbits of the classical cat map. It has been shown in [45] that finding it is as complex as factorization of integers, and can nevertheless be realized on a quantum computer polynomially fast using a variant of order finding. The use of the quantum cat map enables this classical quantity to be obtained by an equally efficient alternate quantum algorithm, showing that in this specific case classical quantities can be obtained through quantum mechanics with exponential efficiency compared to classical algorithms.

VI. CONCLUSION

In the studies above, we have shown that it is possible to implement semiclassical formulas on quantum computers, with a gain in efficiency over implementation on a classical computer. The gain is in general polynomial, but in specific instances an exponential gain can be obtained for related problems. We mention again that the algorithms of Sec. IV can also be used to study quantum systems without reference to the semiclassical approximation, in the manner of [26], with actually a larger gain.

The algorithms of Secs. III and IV can be generalized to a large class of systems. Indeed, to generalize Sec. III one can use the tool of the Poincaré surface of section to transform systems with continuous time to discrete maps. For example, a popular system to study quantum chaos corresponds to billiards, i.e., classically a particle bouncing between walls, and quantum mechanically a wave function obeying Helmholtz equation with boundary conditions. In this case, a simple surface of section is represented by the boundary, the phase-space coordinates being the curvilinear abscissa along the boundary and the angle that the outgoing trajectory makes with the vector normal to the boundary. An alternative possibility would be to stick with the continuous time dynamics and use the semiclassical formulas appropriate for this case. In both cases, it is important to be able to enumerate the classical trajectories used in the semiclassical sums, which requires that a reasonably good symbolic dynamics can be constructed (e.g., with finite Markov partition). This is already the case for classical implementations of these formulas, which have all been performed in such cases. Additionally, the method exposed in Sec. III is more efficient when the Lyapunov exponent of orbits is uniform. In the case where the stability of different orbits varies widely in different phase-space regions, the quantum algorithm will become less efficient. Thus although strongly chaotic systems are the most difficult to treat by semiclassical formulas, they are probably the ones where the algorithms above will be the

most efficient compared to classical algorithms.

To generalize Sec. IV to systems with continuous time is probably possible, but would necessitate to first build an algorithm to simulate such systems on quantum computers. We think that, once this is done, the main ideas of our algorithm in Sec. IV should then be applicable.

The quantum algorithms presented here can be applied to a wide variety of systems. They show that in a domain where extensive numerical simulations have been used in the past decades, a quantum computer could significantly improve the speed of the calculations.

[1] R. P. Feynman, Found. Phys. **16**, 507 (1986).

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

[3] A. Steane, Rep. Prog. Phys. **61**, 117 (1998).

[4] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, U.K., 2000).

[5] 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.

[6] L. K. Grover, Phys. Rev. Lett. **79**, 325 (1997).

[7] S. Lloyd, Science **273**, 1073 (1996).

[8] D. S. Abrams and S. Lloyd, Phys. Rev. Lett. **79**, 2586 (1997).

[9] B. Georgeot and D. L. Shepelyansky, Phys. Rev. Lett. **86**, 2890 (2001).

[10] G. Benenti, G. Casati, S. Montangero, and D. L. Shepelyansky, Phys. Rev. Lett. **87**, 227901 (2001).

[11] O. Giraud and B. Georgeot, Phys. Rev. A **72**, 042312 (2005).

[12] A. Einstein, Verh. Dtsch. Phys. Ges. **19**, 82 (1917).

[13] M. C. Gutzwiller, J. Math. Phys. **12**, 343 (1971); R. Balian and C. Bloch, Ann. Phys. (N.Y.) **85**, 514 (1974).

[14] J. H. Van Vleck, Proc. Natl. Acad. Sci. U.S.A. **14**, 178 (1928).

[15] W. H. Miller, Adv. Chem. Phys. **25**, 69 (1974).

[16] R. Aurich, M. Sieber, and F. Steiner, Phys. Rev. Lett. **61**, 483 (1988).

[17] M. V. Berry, Nonlinearity **1**, 399 (1988).

[18] G. Tanner, P. Scherer, E. B. Bogomonly, B. Eckhardt, and D. Wintgen, Phys. Rev. Lett. **67**, 2410 (1991).

[19] E. Bogomolny and C. Schmit, Nonlinearity **6**, 523 (1993).

[20] H. Primack and U. Smilansky, J. Phys. A **31**, 6253 (1998).

[21] D. Braun, P. A. Braun, and F. Haake, Physica D **131**, 265 (1999); D. Braun, Chaos **9**, 730 (1999).

[22] P. Cvitanovic and B. Eckhardt, Phys. Rev. Lett. **63**, 823 (1989).

[23] M. V. Berry and J. P. Keating, Proc. R. Soc. London, Ser. A **437**, 151 (1992).

[24] J. Main, V. A. Mandelshtam, and H. S. Taylor, Phys. Rev. Lett. **79**, 825 (1997).

[25] J. Emerson, Y. S. Weinstein, S. Lloyd, and D. G. Cory, Phys. Rev. Lett. **89**, 284102 (2002).

[26] D. Poulin, R. Laflamme, G. J. Milburn, and J. P. Paz, Phys. Rev. A **68**, 022302 (2003).

[27] B. Levi, B. Georgeot, and D. L. Shepelyansky, Phys. Rev. E **67**, 046220 (2003); M. Terraneo, B. Georgeot, and D. L. Shepelyansky, *ibid.* **71**, 066215 (2005).

[28] A. Lichtenberg and M. Lieberman, *Regular and Chaotic Dynamics* (Springer, New York, 1992).

[29] E. B. Bogomolny, Nonlinearity **5**, 805 (1992).

[30] V. I. Arnold and A. Avez, *Ergodic Problems of Classical Mechanics* (Benjamin, New York, 1968).

[31] J. H. Hannay and M. V. Berry, Physica D **1**, 267 (1980).

[32] J. Keating, Nonlinearity **4**, 277 (1991); **4**, 309 (1991).

[33] B. V. Chirikov, Phys. Rep. **52**, 263 (1979).

[34] N. L. Balazs and A. Voros, Ann. Phys. (N.Y.) **190**, 1 (1989).

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

[36] M. Tabor, Physica D **6**, 195 (1983).

[37] V. I. Arnold, *Geometrical Methods in the Theory of Differential Equations* (Springer, Berlin, 1988).

[38] M. Basilio de Matos and A. M. Ozorio de Almeida, Ann. Phys. (N.Y.) **237**, 46 (1995).

[39] V. M. Alekseev and M. V. Yakobson, Phys. Rep. **75**, 290 (1981).

[40] R. L. Devaney, *An Introduction to Chaotic Dynamical Systems* (Benjamin, Menlo Park, CA, 1986).

[41] P. Cvitanovic, B. Eckhardt, P. E. Rosenqvist, G. Russberg, and P. Scherer, in *Quantum Chaos*, edited by G. Casati and B. Chirikov (Cambridge University Press, Cambridge, U.K., 1994).

[42] C. Series, J. Lond. Math. Soc. **s2-31**, 69 (1985).

[43] A. Kitaev, e-print arXiv:quant-ph/9511026.

[44] B. Georgeot and D. L. Shepelyansky, Phys. Rev. Lett. **86**, 5393 (2001).

[45] B. Georgeot, Phys. Rev. A **69**, 032301 (2004).