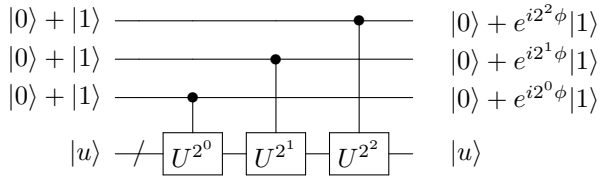
Phase estimation and Shor's algorithm

Artur Ekert and Alastair Kay

I. OPTIMAL PHASE ESTIMATION

Query models of quantum computation provided a natural setting for subsequent discoveries of "real quantum algorithms". The most notable example is Shor's quantum factoring algorithm which evolved from the the order-finding problem, which was originally formulated in the language of quantum queries. Following our "interferometric approach" we will describe this algorithm in the terms of multiparticle quantum interferometry. We start with a simple eigenvalue or phase estimation problem.

Suppose that U is any unitary transformation on m qubits and $|\,u\rangle$ is an eigenvector of U with eigenvalue $e^{i\phi}$ and consider the following scenario. We do not explicitly know U or $|\,u\rangle$ or $e^{i\phi}$, but instead we are given devices that perform controlled-U, controlled-U^{2^1}, controlled-U^{2^2} and so on until we reach controlled-$U^{2^{n-1}}$. Also, assume that we are given a single preparation of the state $|\,u\rangle$. Our goal is to obtain an n-bit estimator of ϕ. We start by constructing the following network,



The second register of m qubits is initially prepared in state $|u\rangle$ and remains in this state after the computation, whereas the first register of n qubits evolves into the state,

$$(|\,0\rangle + e^{i2^{n-1}\phi}\,|\,1\rangle)(|\,0\rangle + e^{i2^{n-2}\phi}\,|\,1\rangle)\cdots(|\,0\rangle + e^{i\phi}\,|\,1\rangle)$$

which can be written as

$$\sum_{y=0}^{2^n-1} e^{2\pi i\frac{\phi y}{2^n}}|y\rangle.$$

Consider the special case where $\phi = 2\pi x/2^n$ for$x = \sum_{i=0}^{n-1} 2^i x_i$, and recall the quantum Fourier transform (QFT). The state which gives the binary representation of x, namely, $|\,x_{n-1}\cdots x_0\rangle$ (and hence ϕ) can be obtained by applying the inverse of the QFT , that is by running the network for the QFT in the backwards direction (consult the diagram of the QFT). If x is an n-bit number this will produce the exact value ϕ.

However, ϕ does not have to be a fraction of a power of two (and may not even be a rational number). For such a ϕ, it turns out that applying the inverse of the QFT produces the best n-bit approximation of ϕ with probability at least $4/\pi^2 \approx 0.405$.

To see why this is so, let us write $\phi = 2\pi(a/2^n + \delta)$, where $a = (a_{n-1}\ldots a_0)$ is the best n-bit estimate of $\frac{\phi}{2\pi}$ and $0 < |\delta| \leq 1/2^{n+1}$. Applying the inverse QFT to the state generated by the network now yields the state

$$\frac{1}{2^n}\sum_{x=0}^{2^n-1}\sum_{y=0}^{2^n-1} e^{\frac{2\pi i}{2^n}(a-x)y}e^{2\pi i\delta y}|x\rangle$$

and the coefficient in front of $|x=a\rangle$ in the above is the geometric series

$$\frac{1}{2^n}\sum_{y=0}^{2^n-1}(e^{2\pi i\delta})^y \;=\; \frac{1}{2^n}\left(\frac{1-(e^{2\pi i\delta})^{2^n}}{1-e^{2\pi i\delta}}\right). \qquad (1)$$

Since $|\delta| \leq \frac{1}{2^{n+1}}$, it follows that $2^n|\delta| \leq 1/2$, and using the inequality $2z \leq \sin\pi z \leq \pi z$ holding for any $z \in [0, 1/2]$, we get $|1 - e^{2\pi i\delta 2^n}| = 2|\sin(\pi\delta 2^n)| \geq 4|\delta|2^n$. Also, $|1 - e^{2\pi i\delta}| = 2|\sin\pi\delta| \leq 2\pi\delta$. Therefore, the probability of observing $a_{n-1}\cdots a_0$ when measuring the state is

$$\left|\frac{1}{2^n}\left(\frac{1-(e^{2\pi i\delta})^{2^n}}{1-e^{2\pi i\delta}}\right)\right|^2 \geq \left(\frac{1}{2^n}\left(\frac{4\delta 2^n}{2\pi\delta}\right)\right)^2 = \frac{4}{\pi^2}, \qquad (2)$$

which proves our assertion.

II. PERIODICITY AND QUANTUM FACTORING

Amazingly, the application of optimal phase estimation to a very particular unitary operator will allow us to factor integers efficiently. In fact, it will allow us to solve a more general class of problems related to the periodicity of certain integer functions.

Let N be an m-bit integer, and let a be an integer smaller than N, and coprime to N. Define a unitary operator U_a acting on m qubits such that for all $y < N$

$$|y\rangle \mapsto U_a|y\rangle = |ay \bmod N\rangle. \qquad (3)$$

This unitary operation can be called multiplication by a modulo N. Since a is coprime to N, as discussed in Section 2, there exists a least strictly positive r such that $a^r = 1 \bmod N$. This r is called the *order* of a modulo N. Equivalently, r is the period of the function $f(x) = a^x \bmod N$, *i.e.* the least $r > 0$ such that $f(x) = f(x+r)$ for all x. We are after the optimal n-bit estimate of this period, given some specified precision n.

Now let the vectors $|u_k\rangle$ ($k \in \{1,\ldots,r\}$) be defined by

$$|u_k\rangle = r^{-1/2}\sum_{j=0}^{r-1} e^{-\frac{2\pi ikj}{r}}\big|a^j \bmod N\big\rangle. \qquad (4)$$

It is easy to check [**?**] that for each $k \in \{1, \ldots, r\}$, $|u_k\rangle$ is an eigenvector with eigenvalue $e^{2\pi i \frac{k}{r}}$ of the modular multiplication operator U_a defined above.

It is important to observe that one can efficiently construct a quantum network for controlled multiplication modulo some number N. Moreover, for any j, it is possible to efficiently implement a controlled-$U_a^{2^j}$ gate [**?** **?**]. Therefore, we can apply the techniques for optimal phase estimation discussed in Section 7. For any $k \in \{1, \ldots, r\}$, given the state $|u_k\rangle$ we can obtain the best n-bit approximation to $\frac{k}{r}$. This is tantamount to determining r itself. Unfortunately, there is a complication.

Our task is: given an m bit long number N and randomly chosen $a < N$ coprime with N, find the order of a modulo N. The problem with the above method is that we are aware of no straightforward efficient way to prepare any of the states $|\, u_k\rangle$. However, the state

$$|\, 1\rangle = r^{-1/2} \sum_{k=1}^{r} |\, u_k\rangle \tag{5}$$

is most definitely an easy state to prepare.

If we start with $|1\rangle$ in place of the eigenvector $|u_k\rangle$, apply the phase estimation network and measure the first register bit by bit we will obtain n binary digits of x such that, with probability exceeding $4/\pi^2$, $\frac{x}{2^n}$ is the best n-bit estimate of $\frac{k}{r}$ for a randomly chosen k from $\{1, \ldots, r\}$. The question is: given x how to compute r? Let us make few observations:

- *k/r is unique, given x.*
  Value $x/2^n$, being the n-bit estimate, differs by at most $1/2^n$ from k/r. Hence, as long as $n > 2m$, the n bit estimate x determines a unique value of $\frac{k}{r}$ since r is an m-bit number.

- *Candidate values for k/r are all convergents to $x/2^m$.*
  For any real number θ, there is a unique sequence of special rationals $(\frac{p_n}{q_n})_{n \in \mathbf{N}}$ $(\gcd(p_n, q_n) = 1)$ called the *convergents* to θ that tend to θ as n grows. A theorem [**?**] states that if p and q are integers with $\left|\theta - \frac{p}{q}\right| < \frac{1}{2q^2}$ then p/q is a convergent to θ. Since we have $\frac{1}{2^n} \leq \frac{1}{2(2^m)^2} \leq \frac{1}{2r^2}$, this implies $\left|\frac{x}{2^n} - \frac{k}{r}\right| < \frac{1}{2r^2}$ and k/r is a convergent to $x/2^n$.

- *Only one convergent is eligible.*
  It is easy to show that there is at most one fraction a/b satisfying both $b \leq r$ and $\left|\frac{x}{2^n} - \frac{a}{b}\right| < \frac{1}{2r^2}$.

Convergents can be found efficiently using the well-known *continued fraction* method [**?**]. Thus we employ continued fractions and our observations above to find a fraction a/b such that $b \leq 2^m$ and $\left|\frac{x}{2^n} - \frac{a}{b}\right| < \frac{1}{2^n}$. We get the rational k/r, and $k = a, r = b$, provided k and r are coprime. For randomly chosen k, this happens with probability greater than or equal to $1/\ln r$ [**?**].

Finally, we show how order-finding can be used to factor a composite number N. Let a be a randomly chosen positive integer smaller than N such that $\gcd(a, N) = 1$. Then the order of a modulo N is defined, and we can find it efficiently using the above algorithm. If r is even, then we have:

$$\begin{aligned} a^r &= 1 \bmod N && (6)\\ \Leftrightarrow \quad (a^{r/2})^2 - 1^2 &= 0 \bmod N && (7)\\ \Leftrightarrow \quad (a^{r/2} - 1)(a^{r/2} + 1) &= 0 \bmod N. && (8) \end{aligned}$$

The product $(a^{r/2}-1)(a^{r/2}+1)$ must be some multiple of N, so unless $a^{r/2} = \pm 1 \bmod N$ at least one of terms must have a nontrivial factor in common with N. By computing the greatest common divisor of this term and N, one gets a non-trivial factor of N.

Furthermore, if N is odd with prime factorisation

$$N = p_1^{\alpha_1} p_2^{\alpha_2} \cdots p_s^{\alpha_s}, \tag{9}$$

then it can be shown [**?**] that if $a < N$ is chosen at random such that $\gcd(a, N) = 1$ then the probability that its order modulo N is even and that $a^{r/2} \neq \pm 1 \bmod N$ is:

$$\Pr(r \text{ is even AND } a^{r/2} \neq \pm 1 \bmod N) \geq 1 - \frac{1}{2^{s-1}}. \tag{10}$$

Thus, combining our estimates of success at each step, with probability greater than or equal to

$$\frac{4}{\pi^2} \frac{1}{\ln r} \left(1 - \frac{1}{2^{s-1}}\right) \geq \frac{2}{\pi^2} \frac{1}{\ln N} \tag{11}$$

we find a factor of N [1]. (Here we have used that N is composite and $r < N$.) If N is $\log N = n$ bits long then by repeating the whole process $O(n)$ times, or by a running $O(n)$ computations in parallel by a suitable extension of a quantum factoring network, we can then guarantee that we will find a factor of N with a fixed probability greater than $\frac{1}{2}$. This, and the fact that the quantum network family for controlled multiplication modulo some number is uniform and of size $O(n^2)$, tells us that factoring is in the complexity class BQP.

But why should anybody care about efficient factorisation?