Lecture 10: Order finding

February 28, 2006

The Order Finding problem

Now that we have discussed the phase estimation technique, it is time to apply it to an interesting computational problem. The problem is called Order Finding, and as we will see in the next couple of lectures it is only one step away from integer factoring. In fact, this will be the quantum part of Shor's factoring algorithm—the rest is completely classical after this.

Some additional notation will be helpful for discussing the Order Finding problem. If a, b, and N are integers with $N \geq 1$ we write

$$a \equiv b \pmod{N}$$

to mean $N \mid (a - b)$ (shorthand for N divides $a - b$). As I mentioned before, we let $\mathbb{Z}_N$ denote the set $\mathbb{Z}_N = \{0, \ldots, N-1\}$. When we also associate with $\mathbb{Z}_N$ the operations of addition and multiplication modulo N, $\mathbb{Z}_N$ forms a ring. If in addition N is prime, then $\mathbb{Z}_N$ forms a field. We write $\mathbb{Z}_N^*$ to denote the following set:

$$\mathbb{Z}_N^* = \{a \in \mathbb{Z}_N \,:\, \gcd(a, N) = 1\}.$$

The number of elements in $\mathbb{Z}_N^*$ determines a function called the *Euler φ-function*: $\varphi(N) = |\mathbb{Z}_N|$. When we associate with the set $\mathbb{Z}_N^*$ the operation of multiplication modulo N, it forms a group. This implies that for any element $a \in \mathbb{Z}_N^*$, there exists a unique element $b \in \mathbb{Z}_N^*$ that satisfies

$$ab \equiv 1 \pmod{N}.$$

We generally write $a^{-1} \pmod{N}$ (or just a^{-1} when N is understood) to denote this element b.

Now, we can define the *order* of a given element $a \in \mathbb{Z}_N^*$, which is what the order-finding problem concerns. For $a \in \mathbb{Z}_N^*$, the *order of a in $\mathbb{Z}_N^*$* (or the *order of a modulo N*) is the smallest positive integer r such that

$$a^r \equiv 1 \pmod{N}.$$

The fact that every $a \in \mathbb{Z}_N^*$ indeed has a well-defined order follows from Euler's Theorem, which states that

$$a^{\varphi(N)} \equiv 1 \pmod{N}$$

for any integers a and $N \geq 2$ with $\gcd(a, N) = 1$. It turns out that the order of any element $a \in \mathbb{Z}_N^*$ is always a divisor of $\phi(N)$.

For example, let $a = 4$ and $N = 35$. Because $\gcd(4, 35) = 1$ we have $4 \in \mathbb{Z}_{35}^*$. Computing powers of 4 modulo 35 gives:

$$4^1 \equiv 4, \quad 4^2 \equiv 16, \quad 4^3 \equiv 29, \quad 4^4 \equiv 11, \quad 4^5 \equiv 9, \quad 4^6 \equiv 1$$

(where all congruences are of course modulo 35). Thus, the order of 4 modulo 35 is 6.

Now that we know how the order of a given element $a \in \mathbb{Z}_N^*$ is defined, we can state the order finding problem. It is as you would probably guess:

Order Finding

Input: A positive integer $N \geq 2$ and an element $a \in \mathbb{Z}_N^*$.

Output: The order of a in $\mathbb{Z}_N^*$.

Classically this problem is hard (at least as far as we know). Certainly the obvious approach of computing powers of a modulo N until 1 is obtained can take time exponential in $\lg N$.

Solving order finding using phase estimation

Our main goal for the remainder of the lecture and part of the next will be to show that the order finding problem can be solved using the method of phase estimation.

Assume $N \geq 2$ is given and let n be the number of bits needed to encode elements of $\mathbb{Z}_N$ in binary (so $n = \lfloor \log_2(N-1) \rfloor + 1$). Given any $a \in \mathbb{Z}_N^*$, define an n-qubit transformation M_a as

$$M_a |x\rangle = |ax \pmod N\rangle .$$

for every $x \in \mathbb{Z}_N$. This is not a complete specification of M_a because it does not specify $M_a |x\rangle$ for $N \leq x < 2^n$, but we will not care about its action on such states. For the sake of choosing a well-defined transformation, we may say for simplicity that $M_a |x\rangle = |x\rangle$ for $N \leq x < 2^n$. The transformation M_a is reversible, and therefore unitary. This follows from the fact that it maps classical states to classical states, along with the observation that it has an inverse: $M_{a^{-1}} M_a = I$, where a^{-1} is the inverse of a modulo N.

Let us consider subjecting the transformation M_a to phase estimation. It will turn out that in doing this we will be able to determine the order of a modulo N. One can efficiently implement a reversible circuit for performing M_a given that the functions $f(x) = ax \pmod N$ and $g(x) = a^{-1}x \pmod N$ are efficiently computable by Boolean circuits. If we wish to subject this transformation to phase estimation, however, recall that we will need to have an efficient implementation of $\Lambda_m(M_a)$ for m (roughly) corresponding to the number of bits of precision we need. In fact, this transformation, which can be expressed as

$$\Lambda_m(M_a) |k\rangle |x\rangle = |k\rangle |a^k x \pmod N\rangle ,$$

can also be implemented efficiently. Specifically, based on the modular exponentiation algorithm I mentioned several lectures ago, it can be implemented using $O(mn^2)$ gates. We will need to wait and see how precise our procedure needs to be to determine the order of a, but we may keep in the back of our minds that $m = O(n)$ will be sufficient.

What are the eigenvectors and eigenvalues of M_a? It turns out that there are lots, but we will only need to consider a subset of them. Letting r be the order of a in $\mathbb{Z}_N^*$, we see that the following vector is an eigenvector of M_a:

$$|\psi_0\rangle = \frac{1}{\sqrt{r}} \left(|1\rangle + |a\rangle + |a^2\rangle + \cdots + |a^{r-1}\rangle \right) .$$

(From now on the operations inside kets are implicitly assumed to be modulo N.) To see that it is an eigenvector, just compute:

$$\begin{aligned} M_a |\psi_0\rangle &= \frac{1}{\sqrt{r}} \left(|a\rangle + |a^2\rangle + |a^3\rangle + \cdots + |a^r\rangle \right) \\ &= \frac{1}{\sqrt{r}} \left(|a\rangle + |a^2\rangle + \cdots + |a^{r-1}\rangle + |1\rangle \right) \\ &= |\psi_0\rangle . \end{aligned}$$

So, the associated eigenvalue is 1. Here is another:

$$|\psi_1\rangle = \frac{1}{\sqrt{r}} \left(|1\rangle + \omega_r^{-1} |a\rangle + \omega_r^{-2} |a^2\rangle + \cdots + \omega_r^{-(r-1)} |a^{r-1}\rangle \right)$$

where as before we define $\omega_r = e^{2\pi i/r}$. We have

$$\begin{aligned} M_a |\psi_1\rangle &= \frac{1}{\sqrt{r}} \left(|a\rangle + \omega_r^{-1} |a^2\rangle + \omega_r^{-2} |a^3\rangle + \cdots + \omega_r^{-(r-1)} |a^r\rangle \right) \\ &= \frac{\omega_r}{\sqrt{r}} \left(\omega_r^{-1} |a\rangle + \omega_r^{-2} |a^2\rangle + \omega_r^{-3} |a^3\rangle + \cdots + \omega_r^{-r} |a^r\rangle \right) \\ &= \omega_r |\psi_1\rangle . \end{aligned}$$

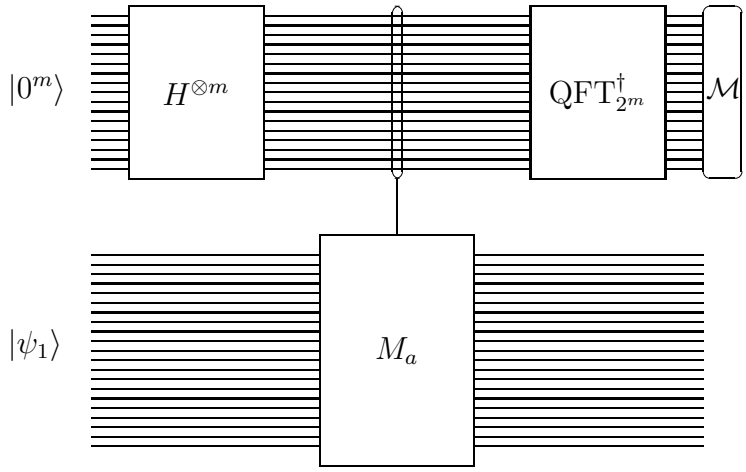
In general, for

$$|\psi_j\rangle = \frac{1}{\sqrt{r}} \left(|1\rangle + \omega_r^{-j} |a\rangle + \omega_r^{-2j} |a^2\rangle + \cdots + \omega_r^{-j(r-1)} |a^{r-1}\rangle \right)$$

we have

$$M_a |\psi_j\rangle = \omega_r^j |\psi_j\rangle .$$

At this point we don't know how to get our hands on any of these eigenvectors, but let's imagine, for the sake of understanding how phase estimation could help us with our problem, that we have a copy of the state $|\psi_1\rangle$. Consider the phase estimation procedure for this eigenvector:



The eigenvalue associated with $|\psi_1\rangle$ is $\omega_r = e^{2\pi i(1/r)}$. Assuming we were to perform the procedure several times and take the most commonly appearing result, we will have an approximation $j/2^m$ that, with very high probability, is within distance $1/2^{m+1}$ to $1/r$. I mentioned this before, but it is worth noting again that we only need one copy of the eigenvector $|\psi_1\rangle$ even if we want to run the phase estimation procedure several times, because the eigenvector is unaffected by the phase estimation procedure.

Remember that our goal is to find r. If we have some integer $j \in \{0, \ldots, 2^m - 1\}$ for which

$$\frac{j}{2^m} \approx \frac{1}{r},$$

there is an obvious strategy that (hopefully) will find r: you just type $j/2^m$ into your calculator and press the button that looks like this:

$$\boxed{1/x}$$

In other words, whether it is with a calculator or (more likely) with an ordinary classical computer, compute the reciprocal of $j/2^m$. Although r must be an integer, you probably would not get an integer when you compute $2^m/j$ because $j/2^m$ is only an approximation to r. The natural thing to do is to round off to the nearest integer, so your guess for r would be

$$\left\lfloor \frac{2^m}{j} + \frac{1}{2} \right\rfloor.$$

Now, if your approximation $j/2^m$ to $1/r$ does not have sufficiently many bits of precision, you cannot be sure your answer is correct. So how accurately do we need to approximate $1/r$ to be correct? Intuitively, you need enough precision to discriminate between estimates for $1/(r-1)$, $1/r$, $1/(r+1)$, and so on, so a good guess is that the approximation from the phase estimation procedure should be within $1/(2r(r+1))$ of the correct value—because this is half the smallest distance between $1/r$ and $1/s$ for some integer $s \neq r$. Of course we do not know what r is, but we do know that $r < N$. So let us guess that it is sufficient that our approximation satisfies

$$\frac{j}{2^m} = \frac{1}{r} - \varepsilon$$

for ε satisfying

$$|\varepsilon| \leq \frac{1}{2N^2}.$$

Indeed this accuracy is sufficient. We can be sure that rounding $2^m/j$ to the nearest integer will give r if

$$\left| \frac{2^m}{j} - r \right| < \frac{1}{2}.$$

Assuming that $|\varepsilon| \leq 1/(2N^2)$ implies

$$\left| \frac{2^m}{j} - r \right| = \left| \frac{1}{\frac{1}{r} - \varepsilon} - r \right| = \left| \frac{r^2\varepsilon}{1 + r\varepsilon} \right| \leq \frac{\frac{r^2}{2N^2}}{1 - \frac{r}{2N^2}} = \frac{r^2}{2N^2 - r} \leq \frac{(N-1)^2}{2N^2 - N} < \frac{1}{2}.$$

Therefore, if we take $m = 2n$ in the phase estimation procedure, the resulting accuracy will be sufficient to find r.

We still have a big problem to contend with, however, which is that we do not know how to get our hands on $|\psi_1\rangle$. Obtaining this vector is probably no easier than finding r, so we will have to change our approach slightly. The solution will be to run the phase estimation procedure on the state $|1\rangle$ rather than on an eigenvector. It follows from the observation that

$$\frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} |\psi_k\rangle = \frac{1}{r} \sum_{k=0}^{r-1} \sum_{l=0}^{r-1} \omega_r^{-kl} |a^l\rangle = |a^0\rangle = |1\rangle$$

that running the phase estimation procedure on the state $|1\rangle$ is equivalent to running the procedure on an eigenvector $|\psi_k\rangle$ for $k \in \{0, 1, \ldots, r-1\}$ chosen uniformly at random. It is not obvious that this is so—it depends on the fact that the phase estimation procedure leaves eigenvectors unchanged, and that these eigenvectors form an orthonormal set. Specifically, if we were to run the phase estimation procedure on the state

$$|1\rangle = \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} |\psi_k\rangle ,$$

the state immediately before the measurement would have the form

$$\frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} |\phi_k\rangle |\psi_k\rangle$$

where each $|\phi_k\rangle$ is the state of the first m qubits that you would get by running the phase estimation procedure just on $|\psi_k\rangle$. Because the states $|\psi_0\rangle, \ldots, |\psi_{r-1}\rangle$ are orthonormal, the probability to obtain some value j from the measurement is just the average over $k \in \{0, \ldots, r-1\}$ chosen uniformly to have measured that value starting with the eigenvector $|\psi_k\rangle$.

So, when we run the phase estimation procedure on $|1\rangle$, we may as well imagine that we instead ran the phase estimation procedure on an eigenvector $|\psi_k\rangle$ for $k \in \{0, \ldots, r-1\}$ chosen uniformly at random. This will be almost as good as having $|\psi_1\rangle$, as we will discuss in the next lecture.