Lecture 12

10 MAR 2014

Last week: quantum Fourier transform & its implementation

In Simon's algorithm, we learned how a quantum computer can be used to find a hidden string s such that $f(x) = f(y)$ iff $y = x \oplus s$. In some sense, this is finding the period of this function. Shor's main observation was to generalize this algorithm to integers mod N & then use it to factor integers.

Today, we will discuss a quantum algorithm to solve the related problem of order finding & there is a well known reduction of factoring to order finding.

Mini review of modular arithmetic & basic number theory

A number is prime if it is an integer > 1 such that its only factors are itself and 1.

Fundamental theorem of arithmetic:

Let $a \in \mathbb{Z}$ & $a > 1$.

Then a has a prime factorization as

$$a = p_1^{a_1} p_2^{a_2} \cdots p_n^{a_n}$$

where $p_1 \cdots p_n$ are distinct primes & $a_1, \ldots, a_n \in \mathbb{Z}^+$.

It is unique.

Modular arithmetic is the arithmetic of remainders.

Given any positive integers x & n, can write

$$x = kn + r$$

where $k \in \mathbb{Z}^+$ & r is the remainder when dividing by n with $r \in \{0, \ldots, n-1\}$

Addition, subtraction, & multiplication are straightforward, but division is not obvious (how to define it).

We need the notion of greatest common divisor (GCD)

GCD(a,b) – largest integer which is a divisor of both a & b.

Euclid's algorithm can compute this efficiently.

Theorem: Let $n \in \mathbb{Z}^+$ & $n > 1$. $a \in \mathbb{Z}^+$ has a multiplicative inverse mod n (denoted by a^{-1}) iff GCD(a,n) = 1. (a & n are said to be co-prime.)

Quantum order finding

Let x & $N \in \mathbb{Z}^+$ w/ $x < N$ & such that they have no common factors.

Let $L = \lceil \log N \rceil$ (# of bits to represent N

order of x mod N is the least $r \in \mathbb{Z}^+$ such that $x^r \equiv 1 \pmod{N}$

We can think of r as being like a period for $f_x(a) = x^a$

Let P be an operator that effects a cyclic shift modulo r:

$$P|y\rangle = |y+1 \pmod{r}\rangle$$

What are the eigenstates of P?

$$|u_k\rangle \equiv \frac{1}{\sqrt{r}} \sum_{y=0}^{r-1} e^{-2\pi i k y/r} |y\rangle$$

$$P|u_k\rangle = \frac{1}{\sqrt{r}} \sum_{y=0}^{r-1} e^{-2\pi i k y/r} |y+1 \pmod{r}\rangle$$

$$= \frac{1}{\sqrt{r}} \sum_{y=0}^{r-1} e^{2\pi i k/r} e^{-2\pi i k (y+1)/r} |y+1 \pmod{r}\rangle$$

$$= e^{2\pi i k/r} \frac{1}{\sqrt{r}} \sum_{y=1}^{r} e^{-2\pi i y k/r} |y \bmod r\rangle$$

$$= e^{2\pi i k/r} |u_k\rangle$$

There is a unitary operator that plays a role similar to a cyclic shift:

"Multiplication by x"

$U|y\rangle = |xy \bmod N\rangle$

How to compute U?

~~$|x,0\rangle \to |x,ax\rangle$~~

$|y,0\rangle \to |y,xy\rangle$ (reversible multiply by x)

$\to$ ~~$|xy,y\rangle$~~ $|xy,y\rangle$ (swap)

$\to |xy,0\rangle$ (uncompute reversible division circuit)

reversible division circuit does

$|y,0\rangle \to |y,x^{-1}y\rangle$

so that its action on

$|xy,0\rangle \to |xy,y\rangle$

so we can implement U efficiently

What are the eigenstates & eigenvalues of $U|y\rangle = |xy\rangle$?

$$|u_k\rangle \equiv \frac{1}{\sqrt{r}} \sum_{y=0}^{r-1} e^{-2\pi i k y/r} |x^y \bmod N\rangle$$

Proof is the same as that for the cyclic shift, i.e.,

$$U|u_k\rangle = \frac{1}{\sqrt{r}} \sum_{y=0}^{r-1} e^{-2\pi i k y/r} |x^{y+1} \bmod N\rangle$$

$$= \frac{1}{\sqrt{r}} \sum_{y=0}^{r-1} e^{\frac{2\pi i k}{r}} e^{-2\pi i k(y+1)/r} |x^{y+1} \bmod N\rangle$$

$$= e^{\frac{2\pi i k}{r}} \frac{1}{\sqrt{r}} \sum_{y=1}^{r} e^{-2\pi i k y/r} |x^y \bmod N\rangle$$

$$= e^{2\pi i k/r} |u_k\rangle$$

How can we compute the controlled-U^{2^j} unitaries needed for phase estimation?

Consider our particular case here.

We wish to compute the transformation

$$|z\rangle|y\rangle \rightarrow$$

$$|z\rangle U^{z_t 2^{t-1}} U^{z_{t-1} 2^{t-2}} \cdots U^{z_1 2^0} |y\rangle$$

$$= |z\rangle |x^{z_t 2^{t-1}} \times \cdots \times x^{z_1 2^0} y \pmod N\rangle$$

$$= |z\rangle |x^z y \pmod N\rangle$$

where z is a t-bit number.

So this sequence of gates (controlled-U) is equivalent to multiplying the contents of the 2nd register by the modular exponential

$$x^z \bmod N.$$

We can actually just do this w/ the techniques of reversible computation.

Basic idea: reversibly compute the function $x^z \bmod N$ in a third register, reversibly multiply 2nd register by $x^z \bmod N$, & then uncompute to erase third register when we're done.

In more detail, there are 2 stages:

1) repeated squaring — compute $x \bmod N$, square that to get $x^2 \bmod N$, square this to get $x^4 \bmod N$, etc. do this for all $t-1$; up to where $t = 2L+1+\lceil \log\left(2+\frac{1}{2\varepsilon}\right) \rceil$

$= O(L)$

So $O(L)$ squaring operations
at a cost of $O(L^2)$
(using the grade school algorithm
for multiplication),
total cost for 1st stage is
$O(L^3)$

2) next stage is based on the
fact that

$$x^z \bmod N = \left(x^{z_t 2^{t-1}} (\bmod N)\right) \times \left(x^{z_{t-1} 2^{t-2}} (\bmod N)\right) \times \cdots \times \left(x^{z_1 2^0} (\bmod N)\right)$$

So we need to perform
$t-1$ modular multiplications
at a cost of $O(L^2)$ each,
for a total of $O(L^3)$ gates

We can then make this into a reversible circuit, which then is promoted to a quantum circuit.

Idea is to use phase estimation of U on $|u_k\rangle$ to approximate k/r.

Some problems w/ this:

1. We don't know r, so we cannot prepare $|u_k\rangle$
2. We only get an approximation of k/r.
3. Even if we knew k/r exactly, they could have common factors.

Solutions to all of these:

1. Estimate k/r for a superposition of $|u_k\rangle$ states
2. Use the continued fraction expansion.
3. can show that $GCD(k,r) = 1$ w/ reasonable probability

To circumvent the 1st problem, consider that

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

$$\frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} \frac{1}{\sqrt{r}} \sum_{y=0}^{r-1} e^{-2\pi i k y/r} |x^y \bmod N\rangle$$

$$= \frac{1}{r} \sum_{y=0}^{r-1} \left(\sum_{k=0}^{r-1} e^{-2\pi i k y/r} \right) |x^y \bmod N\rangle$$

$$= \begin{cases} 1 & \text{if } y = 0 \\ 0 & \text{else} \end{cases}$$

So this is equal to

$$|x^0 \bmod N\rangle = |1\rangle$$

So when we feed this state into phase estimation, we get

$$|0\rangle|1\rangle = |0\rangle \left(\frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} |u_k\rangle \right)$$

$$= \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} |0\rangle |u_k\rangle \rightarrow \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} |\widetilde{k/r}\rangle |u_k\rangle$$

If we measure the first register, then we'll get an integer s such that

$\frac{s}{2^t}$ is an estimate of $\frac{k}{r}$

for some $k \in \{0, \ldots, r-1\}$ selected uniformly @ random. (the estimate will be accurate to $(1-\varepsilon)/r$ up to $2L+1$ bits)

We can then use the classical continued fractions algorithm to determine s & r.

Illustrate this algorithm by example. It decomposes a number as

$$a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \frac{1}{\ddots}}}$$

if $[a_0, a_1, \ldots, a_M]$ are integers then $[a_0, a_1, \ldots, a_m]$ where $m \leq M$ is called mth convergent.

Example: $\frac{31}{13} = 2 + \frac{5}{13}$ "split"

$= 2 + \frac{1}{\frac{13}{5}}$ "invert"

continue w/ $\frac{13}{5}$

$\frac{13}{5} = 2 + \frac{3}{5} \Rightarrow$

$\frac{31}{13} = 2 + \frac{1}{2 + \frac{3}{5}}$

continue w/ $\frac{3}{5} = \frac{1}{5/3}$

$5/3 = 1 + \frac{2}{3}$

so $\frac{31}{13} = 2 + \frac{1}{2 + \frac{1}{1 + \frac{2}{3}}}$

$\frac{2}{3} = \frac{1}{3/2}$ $\quad 3/2 = 1 + \frac{1}{2}$

so terminates

$\frac{31}{13} = 2 + \frac{1}{2 + \frac{1}{1 + \frac{1}{1 + \frac{1}{2}}}}$

this expansion can be computed w/ $O(L^3)$ operations

Suppose that s/r is a rational # such that

$$\left|\frac{s}{r} - \varphi\right| \leq \frac{1}{2r^2}$$

Then $\frac{s}{r}$ is a convergent of the continued fraction for φ & can be computed w/ $O(L^3)$ ops.

Since ℓ is an approximation of s/r accurate up to $2L+1$ bits

this means that

$$|s/r - \varphi| \leq 2^{-2L-1} \leq \frac{1}{2r^2}$$

since $r \leq N \leq 2^L$

So the continued fractions alg. efficiently produces s' & r' w/ no common factor such that $s'/r' = s/r$

Then r' is the candidate for the order & test it by $x^{r'} \bmod N$ & seeing if it equals 1.