6.841/18.405J: Advanced Complexity Theory May 13, 2002

Lecture 23

Lecturer: Madhu Sudan *Scribe: Alice Chan*

1 Today

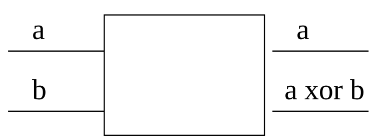
Recap of quantum computing model, Simon's algorithm, Shor's algorithm for factoring.

1.1 Quantum circuits

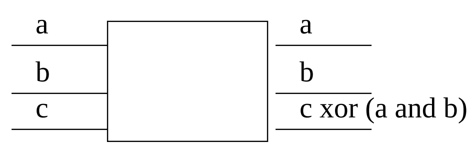
These are circuits with n different wires combined using quantum gates. A quantum gate is a map from $2^k \to 2^k$. The Hadamard transform,"negation" and "and" form a sufficient collection of gates.

The Hadamard transform, H_2: $\begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}$,

"negation"



"and"



The transition function for a quantum TM is a transition matrix and the tape consists of q-bits.

"Quantum polynomial-time"
BQP is, in some sense, an extension of BPP and EQP an extension of ZPP. BQP is the class of languages that can be solved with a polynomial number of steps on a quantum TM, with completeness and soundness as defined before. An equivalent definition is that BQP is the class of languages that can be solved by a polynomial-sized quantum circuit which is constructible in classical polynomial-time.

1.2 Simon's algorithm

This algorithm is for a promise problem where we are trying to decide whether a function f is 1-1.
The oracle is the function $f : \{0,1\}^n \to \{0,1\}^n$.
A YES instance is the case when f is not 1-1, ie. $\exists s \in \{0,1\}^n - \{0\}^n$ s.t. $\forall x f(x+s) = f(x \oplus s) = f(x)$. If such an s exists, f is at approximately 2-1.
A NO instance is the case that f is 1-1.
Suppose $x = |010111>$ and apply H_2 to each of the bits. The outcome is $\frac{1}{2^{\frac{n}{2}}} \sum_{y \in \{0,1\}^n} (-1)^{<x,y>} |y>$. Each of the 2^6 possible outcomes has equal probability of occuring.

Simon's algorithm:
Initialize the quantum circuit to $|0^n, 0^n>$.
Apply H_2 to the first n bits and get $\frac{1}{2^{\frac{n}{2}}} \sum_{x \in \{0,1\}^n} |x, 0^n>$.
Set the machine to $\frac{1}{2^{\frac{n}{2}}} \sum_{x \in \{0,1\}^n} |x, f(x)>$
(as classical computation can be simulated in the quantum world).
Undo the Hadamard computation.
(If the Hadamard computation was undone at the stage with $\frac{1}{2^{\frac{n}{2}}} \sum_{x \in \{0,1\}^n} |x, 0^n>$ we get $|0^n, 0^n>$.
But with $\frac{1}{2^{\frac{n}{2}}} \sum_{x \in \{0,1\}^n} |x, f(x)>$, the result is $\frac{1}{2^{\frac{n}{2}}} \sum_{x \in \{0,1\}^n} \frac{1}{2^{\frac{n}{2}}} \sum_y (-1)^{<x,y>} |y, f(x)>$.)
Observe the tape.

In the NO instance, every string $< y, z >$ is observed in $|y, f(x) >$ since f is 1-1. So the state, $\frac{1}{2^n}\sum_{y,z}(\pm 1)|y, z >$, is a uniformly distributed random sample.
In the YES case, f is approximately 2-1, so $f(x)$ will only take on 2^{n-1} values.
If $< y, s >= 1$, then

$$(-1)^{<x,y>}|y, f(x) > +(-1)^{<x+s,y>}|y, f(x+s) >= (-1)^{<x,y>}(|y, f(x) > +(-1)|y, f(x) >)$$

as $f(x+s) = f(x)$.
If $< y, s >= 0$, then all possible 2^{2n-2} vectors $|y, f(x) >$ are seen with equal amplitude. (There are 2^{2n-2} possibilities because half of the vectors are ruled out since f is 2-1 and half of the remaining are ruled out because $< y, s >= 0$.)
Sampling from this circuit $2n$ times and writing the results $y_1, \ldots, y_{2n}$ as a matrix, either we get $y_1, \ldots, y_{2n}$ of rank n in the NO case or we get a rank of $n-1$ for the YES case.

1.3 Shor's algorithm

Intuition: Given n, pick a random $a \in \mathbb{Z}_n^*$. Then factoring n reduces to computing the order of $a \bmod n$ (finding r such that $a^r - 1 \equiv 0 \bmod n$). Simon's algorithm seems to compute periods of functions so perhaps it can be used to compute the period of the order function $f(i) = a^i$, ie. it can find r such that $f(i+r) = f(i)$.
Fix a, n and some q. Let $j \in \mathbb{Z}_q$ and define a unitary operator $|j >\mapsto \frac{1}{\sqrt{q}}\sum_{k=0}^{q-1} e^{\frac{2\pi i}{q} j*k}|k >$, similar to a complex Fourier transform.

Shor's algorithm:
Initialize the state to $|0, 0 >$.
Apply the unitary operator above to the first half and get $\frac{1}{\sqrt{q}}\sum_j |j, 0 >$.
Set the machine state to $\frac{1}{\sqrt{q}}\sum_j |j, f(j) >$, where f is the order function.
Apply the unitary operator to get $\frac{1}{q}\sum_j\sum_k e^{\frac{2\pi i}{q} j*k}|k, f(j) >$.
Observe state.

Claim: k is very close to a multiple of $[\frac{q}{r}]$.
(Proof omitted.)

Assume $q = mr$ for some m.
Writing out $\frac{1}{q}\sum_j\sum_k e^{\frac{2\pi i}{q} j*k}|k, f(j) >$ as

$$\frac{1}{q}\sum_{j_1=0}^{\frac{q}{r}-1}\sum_{j_2=0}^{r-1}\sum_{k} e^{\frac{2\pi i}{q}(rj_1+j_2)*k}|k, f(j_2) >= \sum_k\sum_{j_2}|k, f(j_2) > e^{2\pi*i*j_2*k}(\sum_{j_1=0}^{\frac{q}{r}-1} e^{\frac{2\pi i}{q} r*j_1*k})$$

$$= \sum_{j_1=0}^{m-1}(e^{\frac{2\pi i}{m}k})^{j_1} = \begin{cases} m & \text{if } k \text{ is a multiple of } m, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

Major issues:
1) q is not a multiple of r:
Get k such that $[kr]_q$ is very small contribute (handled by extending analysis and applying integer programming in $O(1)$ variables).
2) q-ary Fourier transform is not always local:
In the case where q is a power of 2, can construct a small quantum circuit implementing any q-ary FT.