COS 533: Advanced Cryptography — Princeton University
Lecture 23 (December 11, 2017)
Lecturer: Mark Zhandry — Scribe: Fermi Ma

Notes for Lecture 23

1 Solving Hard Problems with Quantum Computation

Today we'll talk about how to use the effects of quantum mechanics to solve hard computational problems. We'll switch to a circuit model of computation. Let's briefly recap the classical circuit model (in the non-uniform setting).

1.1 Classical Circuit Model

In this model, you have a finite "basis" B of gates ex: NOT, AND, OR, XOR. It turns out that this basis B generates all possible functions on $\{0,1\}^n$. Actually, it suffices to have NOT and AND. We call basis sets that generate all possible functions "universal". In cryptography, we usually care about the complexity of various functions. Here, we'll measure complexity by the number of gates in the circuit. So we consider a circuit to be polynomial time if there is a polynomial number of gates (poly-sized circuit).

1.2 Quantum Circuit Model

In this model, we have a finite basis of gates (which will be unitary operations). We might hope that we can generate all possible unitaries on an arbitrary number of qubits with a finite basis. But this is impossible, even for one qubit. Recall that a unitary matrix looks like

$$U = \begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix} \in \mathbb{C}^{2\times 2}$$

where $U^\dagger U = I$ and the entries are real numbers. A finite basis will give us a countable number of unitary matrices. However, a unitary matrix has real number entries, and thus there are an uncountable number of them.

Doing this exactly is impossible, so instead we will approximate.

Just as with classical gates, there's an arbitrary number of ways to give the basis vectors. We'll give one such basis consisting of the following three unitiaries

- A Hadamard gate H is one that satisfies

$$H|0\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{2}|1\rangle$$
$$H|1\rangle = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{2}|1\rangle$$

  It takes a vertical polarization and moves it to 45 degrees, and a horizontal polarization and moves it to 135 degrees. We can think of it as basically a rotation. We like this gate because people have showed how to build this for various quantum systems. We can write this gate as

$$\frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}$$

  and easily confirm that it is unitary.

- The next gate we'll use is the $\pi/8$ gate, which we denote by T. This one satisfies

$$T|0\rangle = |0\rangle$$
$$T|1\rangle = e^{i\pi/4}|1\rangle$$

  Note that we can represent $e^{i\pi/4}$ as $\cos(\theta) + i\sin(\theta)$. So this gate doesn't do anything to $|0\rangle$, but it adds a phase to $|1\rangle$. It's a silly convention that this has a $\pi/4$ in the exponent but it's called a $\pi/8$ gate (probably due to the fact that usually we write 2π instead of π).

  We can write this gate as

$$\begin{pmatrix}1 & 0\\ 0 & e^{i\pi/4}\end{pmatrix}$$

  which shows that it is unitary.

- The last gate is the CNOT gate. The CNOT gate is a two qubit gate that does

$$CNOT|a,b\rangle = |a, a \oplus b\rangle$$

Informal Theorem: $\{H, T, CNOT\}$ is universal for quantum computation (if we allow arbitrarily good approximations).

1.3 Simulating Classical Computation

We claim that quantum circuits can be used to efficiently simulate any classically efficient computation.

Suppose we have $f : \{0,1\}^n \to \{0,1\}^m$ that is classically efficiently computable. We might want to build a U_f that is efficiently computable such that $U_f|x\rangle = |f(x)\rangle$. The problem with this approach is that this U_f might not be a unitary transformation. For example if $f(x) = 0$, then such a U_f won't be unitary because we can't take inverses.

Because unitary matrices have inverses, quantum computation is reversible. What we can do is simulate any reversible classical operation. We'll redefine our goal to be to compute a U_f such that

$$U_f|x, y\rangle = |x, f(x) \oplus y\rangle$$

We won't prove this, but you can build an approximate U_f from the quantum gates.

1.4 Quantum Fourier Transform

The Quantum Fourier Transform is parameterized by a q and operates on n qubits. First, let $q = 2$, $w_q = (-1)$, and $n > 0$.

Let $\overline{x}$ be a binary string of length n. We have

$$QFT_2|\overline{x}\rangle = \frac{1}{\sqrt{2^n}} \sum_{\overline{y} \in \mathbb{Z}_2^n} (w_q)^{\langle \overline{x}, \overline{y} \rangle} |\overline{y}\rangle$$

More generally we have $w_q = e^{i2\pi/q}$. The input will be $\overline{x} \in \mathbb{Z}_q^n$. The transformation is

$$QFT_q|\overline{x}\rangle = \frac{1}{\sqrt{q^n}} \sum_{\overline{y} \in \mathbb{Z}_q^n} (w_q)^{\langle \overline{x}, \overline{y} \rangle} |\overline{y}\rangle$$

It takes the basis vector $\overline{x}$ as input, and creates a uniform superposition of all the $|\overline{y}\rangle$ states, but it adds a phase depending on the inner product mod q between the input basis state $|\overline{x}\rangle$ and whatever $|\overline{y}\rangle$ we are looking at.

First, we should verify that this transform is even unitary. What does the corresponding matrix look like?

Imagine a matrix where the rows are indexed by $\overline{y}$ and the columns are indexed by $\overline{x}$. The entry in column $\overline{x}$ and row $\overline{y}$ is $\frac{w_q^{\langle \overline{x}, \overline{y} \rangle}}{\sqrt{q^n}}$. The conjugate transpose has rows indexed by $\overline{x}$, columns indexed by $\overline{y}$, and the entry in row $\overline{x}$ and column $\overline{y}$ is $w_q^{-\langle \overline{x}, \overline{y} \rangle}$.

First note that

$$QFT_q^\dagger|\overline{y}\rangle = \frac{1}{\sqrt{q^n}} \sum_{\overline{x}} w_q^{-\langle \overline{x}, \overline{y} \rangle} |\overline{x}\rangle$$

Now we compute

$$\begin{aligned} QFT_q^\dagger QFT_q |\overline{x}_0\rangle &= QFT_q^\dagger (\frac{1}{\sqrt{q^n}}) \sum_y w_q^{\langle \overline{x}_0, \overline{y} \rangle} |\overline{y}\rangle) \\ &= \frac{1}{q^n} \sum_{\overline{y}} w_q^{\langle \overline{x}_0, \overline{y} \rangle} \sum_{\overline{x}} w_q^{-\langle \overline{x}, \overline{y} \rangle} |\overline{x}\rangle \\ &= \sum_{\overline{x}} (\frac{1}{q^n} \sum_{\overline{y}} w_q^{\langle \overline{x}_0 - \overline{x}, y \rangle}) |\overline{x}\rangle \end{aligned}$$

We claim that $\sum_{y=0}^{q-1} w_q^{zy}$ is q if $z = 0 \mod q$ and 0 if $z \neq 0 \mod q$.

We can prove this by thinking of this as a geometric series, expanding out the definition of w_q, and applying the appropriate sum formula.

More generally, we'll need that

Claim. $\sum_{\overline{y} \in \mathbb{Z}_q^n} w_q^{\langle \overline{z}, \overline{y} \rangle}$ is $q^n = 1$ if $\overline{z} = \overline{0} \mod q$ and 0 otherwise.

We can do this by breaking up the sum into a product of n sums and evaluate each part.

We can conclude that

$$\sum_{\overline{x}} (\frac{1}{q^n} \sum_{\overline{y}} w_q^{\langle \overline{x}_0 - \overline{x}, y \rangle}) |\overline{x}\rangle = |x_0\rangle$$

and so it is indeed unitary.

In the case where $q = 2, n = 1$, expanding out QFT_q gives the Hadamard gate H. For $n > 1$, this is the Hadamard gate on each qubit.

Example: Suppose we have $q = ab$. Let

$$|\psi\rangle = \sum_{r=0}^{b-1} \frac{1}{\sqrt{b}} |ar\rangle$$

What is $QFT_q|\psi\rangle = \sum_{s=0}^{a-1} \frac{1}{\sqrt{a}} |bs\rangle$? Basically we get equal weight on all the multiples of b and 0 everywhere else.

Why does this happen? Applying the QFT within the sum gives

$$\sum_{r=0}^{b-1} \frac{1}{\sqrt{b}} (\frac{1}{\sqrt{q}} \sum_y w_q^{(ar)y} |y\rangle) = \sum_y (\frac{1}{b\sqrt{q}} \sum_{r=0}^{b-1} w_q^{ary}) |y\rangle$$

Note that $(w_q^a)^{ry} = w_b^{ry}$. Now we apply the claim from above. This tells us that this sum is 0 unless $y = 0 \mod b$.

Example: Consider a multi-dimensional setting. Let $A \in \mathbb{Z}_q^{n \times m}$

Suppose my state is

$$|\psi\rangle \propto \sum_{x:Ax=0} |x\rangle$$

where $x \in \mathbb{Z}_q^m$ (we use the $\propto$ symbol just so we don't have to deal with normalization).

If we apply the QFT, we get

$$QFT|\psi\rangle \propto \sum_r |A^\dagger r\rangle$$

The previous example was where $n = m = 1$ and A was just b.

Example: We can see that applying the QFT to a shifted quantum state is equivalent to applying the QFT to the original quantum state and then doing a phase shift. Suppose my state is

$$|\psi\rangle = \sum_{\overline{x}} \alpha_{\overline{x}} |\overline{x}\rangle.$$

Applying the QFT is $|\phi\rangle = \sum_{\overline{y}} \hat{\alpha}_{\overline{y}} |\overline{y}\rangle$

Suppose I also have a shifted state

$$|\psi'\rangle = \sum_{\overline{x}} \alpha_{\overline{x}} |\overline{x} + \overline{r}\rangle.$$

Then applying the QFT gives $|\phi'\rangle = \sum \hat{\alpha}_{\overline{y}} w_q^{\langle \overline{x}, \overline{r} \rangle} |\overline{y}\rangle$.

1.5 Discrete Log

We show how to use the QFT to solve the discrete log problem. Recall that we're given $g, h = g^a$ which are elements of a cyclic group G of order q. The goal is to find a.

First we construct the state that just has equal amplitude on all states by applying the QFT on the 00 state.

$$|\psi_0\rangle = \frac{1}{q} \sum_{(x,y) \in \mathbb{Z}_q^2} |x, y\rangle = QFT_q |00\rangle$$

Let $f : \mathbb{Z}_q^2 \to G$. Let $f(x, y) = g^x h^{-y}$. Now we apply U_f to $|\psi_0\rangle|0\rangle$ (note that we need this extra register to ensure the computation is reversible).

We compute

$$|\psi_1\rangle = U_f|\psi_0\rangle|0\rangle = \frac{1}{q}\sum_{x,y}|x,y,g^x h^{-y}\rangle.$$

Now we measure the last set of qubits. We get some group element $U = g^r$. Recall that when we measure, all the terms that are inconsistent with the measurement go away. We get

$$\sum_{x,y:g^x h^{-y}=U}|x,y\rangle|U\rangle$$

Once I do the measurement, the U is not entangled with the x, y anymore. So we can basically ignore the U at this point. So we define

$$|\psi_U\rangle \propto \sum_{x,y:g^x h^{-y}=U}|x,y\rangle$$

We know that $g^x h^{-y} = g^{x-ay} = g^r$. If this is true, then $x - ay = r$, so what I actually get is a state with equal weight on all x and y that satisfy this linear constraint. More precisely, we have

$$|\psi_U\rangle = \frac{1}{\sqrt{q}}\sum_{x,y:x-ay=r}|x,y\rangle$$

Let's pretend $r = 0$ for now. Let A be the matrix $\begin{pmatrix}1 & -a\end{pmatrix}$. We get

$$|\psi_U\rangle = \frac{1}{\sqrt{q}}\sum_{x,y:A\cdot\begin{pmatrix}x & y\end{pmatrix}^T=0}|x,y\rangle$$

which is a state with uniform weight on all vectors in the kernel of A.

So we apply the QFT to this and get

$$QFT|\psi_U\rangle = \frac{1}{\sqrt{q}}\sum_{z}|z,-az\rangle$$

which is a state with equal weight on everything in the rowspace of A.

So I can just measure this, get one element of the form $z, -az$, divide out the z, and recover a.

But for general r, let x_0, y_0 be some solution to $x_0 - ay_0 = r$. Then we can use the fact that given a solution to $A \cdot \begin{pmatrix}x & y\end{pmatrix}^T = 0$, we can get another solution by adding an element in the nullspace. We write

$$|\psi_u\rangle = \frac{1}{\sqrt{q}}\sum_{x,y:A\cdot\begin{pmatrix}x & y\end{pmatrix}^T=0}|x+x_0,y+y_0\rangle$$

So this state is just the shift of the state we want, which was

$$|\psi_U\rangle = \frac{1}{\sqrt{q}} \sum_{x,y:A\cdot\begin{pmatrix} x & y \end{pmatrix}^T=0} |x, y\rangle$$

But recall from our earlier example that applying the Fourier Transform to a shift means that we just pick up a phase. When I go to measure, the phase doesn't affect any of my probabilities. So we still get the same thing as before and recover a.

Next time we'll see how to use these ideas to factor integers. We'll also see how to speed up solving NP problems.