Quantum Computation

oliverobrien111

July 2021

1 Lecture 1

1.1 Review of Shor's algoirthm/quantum period finding algorithm

Polynomial time hierarchy:// Computation with input of size n, and we are interested in the number of steps/gates (classical or quantum). When we say $O(poly(n))$ steps we regard this as an "efficent computation".

Shor's algorithm solves the factoring problem:
Given an integer N needing $O(logN)$ bits, we want to find a non-trival factor in $O(polyn)$ time.

The best known classical algorithm (number sieve): $e^{O(n^{\frac{1}{3}}(\log n)^{\frac{1}{3}})}$
Shor's alogrithm takes $O(n^3)$

1.1.1 Quantum factoring algorithm (summary)

1. First, convert factoring into periodicity determination. Given N, choose $a < N$ s.t. a is coprime (this is easy classically can be seen in part II lecture notes). Consider $f : \mathbb{Z} \to \mathbb{Z}_N$ $f(x) = a^x \mod N$. **Euler's Theorem**: if f is periodic with period r, then it is called 'order of $a \mod N$'.

2. In order to find r we need a quantum implementation of f. We are always workingon finite size registers so restricting $x \in \mathbb{Z}$ to $x \in \mathbb{Z}_M$ (for some large enough M): $f : \mathbb{Z}_M \to \mathbb{Z}_N$. f will no longer be exactly preriodic but this would have neglible effect if M is sufficently large e.g. $M = O(N^2)$

3. Using the classical theory of continued fractions. Define Hilbert spaces $\mathcal{H}_M \to \{|i\rangle\}_{i\in\mathbb{Z}_M}, \mathcal{H}_N \to \{|i\rangle\}_{i\in\mathbb{Z}_N}$.

4. $|x\rangle \to |f(x)\rangle$ is not generally a valid quantum operator, so we make it a unitary operation which can be implemented:

$$U_f : \mathcal{H}_M \otimes \mathcal{H}_N \to \mathbb{H}_M \otimes \mathbb{H}_N$$

$$U_f : |i\rangle |k\rangle \to |i\rangle |k + f(i)\rangle$$

5. if $x \to f(x)$ can be classically computed in $O(poly(m))$ time ($m = \log M$)), then U_f can be implemented in poly(m) time quantumly too

6. We will sometimes view U_f as a black box/oracle and we will count the number of times the algorithm invokes the oracle.

7. Back to facotoring to get r we'll use the quantum algorithm for periodicity determination:

8. Given an oracle U_f with the promise that f is periodic of some unknown period $r \in \mathbb{Z}_N$ so that $f(x + r) = f(x)$ and f is one-to-one in this period (for all $0 \leq x_1 < x_2 < rf(x_1) \neq f(x_2)$)

9. To find r in $O(polyn)$ with any persecribed success probability $1 - \epsilon$ we use the following alogirthm:

   - Step 1: Create the state

   $$\frac{1}{\sqrt{M}} \sum_{i=0}^{M-1} |i\rangle |0\rangle$$

   - Step 2: Apply U_f to get

   $$\frac{1}{\sqrt{M}} \sum_{i=0}^{M-1} |i\rangle |f(i)\rangle$$

   - Step 3: Measure the 2nd register to get y. By the born rule the first register collapses to all those i: $f(i) = y$ i.e. $i = x_0, x_0 + r, x_o + 2r, ..., x_0 + (A-1)r$, $0 \leq x_0 < r$.

     Discard the second register to get the following state:

     $$|per\rangle = \frac{1}{\sqrt{A}} \sum_{j=0}^{A-1} |x_0 + jr\rangle$$

     If we measure $|per\rangle$ in computation basis we will get a value of one of these states $x_0 + jr$ for uniformly random j. This only gives us a random element of $\mathbb{Z}_M$ with no information about r.

   - Step 4: Apply quantum fourier transform mod M (QFT). Lets recap what QFT does:

   $$|x\rangle \to \frac{1}{\sqrt{M}} \sum_{y=0}^{M-1} \omega^{xy} |y\rangle, \forall x \in \mathbb{Z}_M, \omega = e^{2\pi i/M}$$

This can be implement in $O(m^2)$ time and gives state:

$$QFT\,|per\rangle = \frac{1}{\sqrt{MA}} \sum_{j=0}^{A-1} \sum_{y=0}^{M-1} \omega^{(x_0+jr)y}\,|y\rangle = \frac{1}{\sqrt{MA}} \sum_{y=0}^{M-1} \omega^{x_0 y} \left[\sum_{j=0}^{A-1} \omega^{jry}\,|y\rangle\right]$$

The square brackets will be:

$$\begin{cases} A & \text{if } y = KA = k\frac{M}{r}, x = 0, 1, ..., r-1 \\ 0 & \text{otherwise} \end{cases}$$

So gives final state:

$$QFT\,|per\rangle = \sqrt{\frac{A}{M}} \sum_{k=0}^{A-1} \omega^{x_0 k \frac{N}{r}}\,|k\frac{M}{r}\rangle$$

Now the random shift x_0 only appears in the phase not in the ket labels. So now the measurement probabilities will be indepedant of x_0. When we measure this we get some value $c = \frac{k_0 M}{r}$ with k_0 uniformly random in range $0 \leq k_0 < r$

$$\frac{k_0}{r} = \frac{c}{M}$$

As c and M are known, and k_0 is unknown but random in the given range. We want to find r and so we recall several classical facts.

Co-primality Theorem: The number of integers less than r that are coprime to r grows with $O(\frac{r}{\log\log r})$

Therefore, the probability of k_0 being coprime to r is $O(\frac{1}{\log\log r})$.

Lemma: If a single trial has success probability p then if one repeats it M^* times, for any $0 < 1-\epsilon < 1$. We get probability of at least one success in M^* trails is greater than $1-\epsilon$ if $M^* = \frac{-\log\epsilon}{p}$. i.e. roughly $O(1/p)$ trials suffice to achieve probability of success $> 1-\epsilon$

- After step 4 cancel $\frac{c}{M}$ down to an irredicible algorithm $\frac{a}{b}$ there is an efficent algorithm ($O(polyn)$) for this. This will give us r as denominator b if k_0 is coprime to r with probablity $O(\frac{1}{\log\log r})$. So check b value by computing $f(0)$ and $f(b)$ and $b = r \iff f(0) = f(b)$.

  By repeating this process $M^* = O(\log\log r)$ times this will give us r with any desired probability $1-\epsilon$. Since $r < M$ the whole algorithm takes $O(polym)$ time!

10. From learning the period r we can use number theory to find a factor of N

1.1.2 Further insights to QFT

Now lets think about the implications of QFT. What does applying quantum fourier transform really achieve?

Lets consider a function: $f : \mathbb{Z}_M \to \mathbb{Z}_N$ with period $r \in \mathbb{Z}_M$, $A = \frac{M}{r}$. Define:

$$R = \{0, r, 2r, 3r, ..., (A-1)r\} < \mathbb{Z}_M$$

$$|R\rangle = \frac{1}{\sqrt{A}} \sum_{k=0}^{A-1} |kr\rangle$$

$$|per\rangle = |x_0 + R\rangle = \frac{1}{\sqrt{A}} \sum_{k=0}^{A-1} |x_0 + rk\rangle$$

The problem was this random shift x_0 when measuring $|per\rangle$. For each $x_0 < \mathbb{Z}_M$ consider a mapping $k \to k + x_0$. "Shift by x_0". It is a 1-1 invertible map, and can define a unitary version $U(x_0)$ on $\mathcal{H}_M$: $U(x_0)\,|k\rangle = |k + x_0\rangle$.

$$|x_0 + R\rangle = U(x_0)\,|R\rangle$$

Since $(\mathbb{Z}_M, +)$ is an abelian group $U(x_0)U(x_1) = U(x_0 + x_1) = U(x_1)U(x_0)$. So all $U(x_i)$ commute as operators on $\mathcal{H}_M$. Therefore they have an orthonomal basis of common eigenvectors $\{|\chi_k\rangle\}_{k \in \mathbb{Z}_M}$. These are called shift invariant states as $U(x_0)\,|\chi_k\rangle = \omega(x_0, k)\,|\chi_k\rangle$ for all $x_0, k \in \mathbb{Z}_M$ with the important caveat that $|\omega(x_0, k)| = 1$.

Consider $|R\rangle$ written in $\{|\chi_r\rangle\}$ basis:

$$|R\rangle = \sum_{k=0}^{M-1} a_k\,|\chi_k\rangle$$

a_k only depend on r not on x_0. Then:

$$|per\rangle = U(x_0)\,|R\rangle = \sum_{k=0}^{M-1} a_k \omega(x_0, k)\,|\chi_k\rangle$$

Here it can be seen that the probability of measuring k is

$$prob(k) = |a_k \omega(x_0, k)|^2 = |a_k|^2$$

So this is all indepedant of x_0 and depends only on r. So measuring in this basis gives us some information about r. So one can think of QFT as the unitary mapping that rotates χ basis into the standard computational basis. So can define QFT as:

$$QFT\,|\chi_k\rangle = |k\rangle$$

How do these mysterious shift invariant states look?

1.1.3 Explicit form of shift invariant shapes

$$|\chi_k\rangle = \frac{1}{\sqrt{M}} \sum_{l=0}^{M-1} e^{-2\pi i l \frac{k}{M}} |l\rangle$$

$$U(x_0)|\chi_k\rangle = 1\sqrt{M} \sum_{l=0}^{M-1} e^{-2\pi i l \frac{k}{M}} |l + x_0\rangle = 1\sqrt{M} \sum_{\tilde{l}=0}^{M-1} e^{-2\pi i (\tilde{l}-x_0) \frac{k}{M}} |\tilde{l}\rangle = e^{2\pi i k \frac{x_0}{M}} |\chi_k\rangle$$

giving eigenvalue: $\omega(x_0, k) = e^{2\pi i k \frac{x_0}{M}}$. From this we could reconstruct the basis of QFT:

$$[QFT]_{kl} = \frac{1}{\sqrt{M}} e^{2\pi i \frac{kl}{M}}$$

2 Lecture 3

2.1 Hidden Subgroup Problem

Let G be a finite group of size $|G|$. We are given an oracle $f : G \to X$ with X just some set. We are promised there is a subgroup $K < G$ s.t.

f is constant on (left) cosets of K in G

f is distinct on distinct cosets

Problem: 'Determine' the 'hidden subgroup' K (e.g. output a set of generators or sample uniformly from elements of K)

We want to solve in time $O(poly(\log |G|))$ (efficent algorithm) with anuy consitent probability $1 - \epsilon$. **Examples of problems that can be cast as HSP** Periodicity finding $f : \mathbb{Z}_M \to X$ periodic, period r 1-1 in period

$$G = \mathbb{Z}_M, K = \{0, r, 2r, ..., (A-1)r\} < G$$

Discrete Logarithm Problem: p - prime number, $\mathbb{Z}_p^*$ group of integers with multiplication mod p, $g \in \mathbb{Z}_p^*$ to be a generator (or primitive root mod p). If $\mathbb{Z}_p^* = \{g^0, g^1, ..., g^{r-2}\}$ and we have $g^{p-1} = 1 \pmod p$. Fact: These always exist for p is prime. Any $x \in \mathbb{Z}_p^*$ can be written as $x = g^y$ for some $y \in \mathbb{Z}_{p-1}$, $y = \log_g x$ is called the discrete log of x to base g. Discrete log problem is given a generator g, $x \in \mathbb{Z}_p^*$ we want to compute $y = \log_g x$. To express this as the HSP:

$$f : \mathbb{Z}_{p-1} \times \mathbb{Z}_{p-1} \to \mathbb{Z}_p^*$$

$$f(a, b) = g^a x^{-b} \mod p = g^{a-yb} \mod p$$

Can check if $f(a_1, b_1) = f(a_2, b_2) \iff (a_1, b_1) = (a_2, b_2) + \lambda(y, 1), \in \mathbb{Z}_{p-1}$:

$$G = \mathbb{Z}_{p-1} \times \mathbb{Z}_{p-1}$$

$$K = \{\lambda(y, 1) : \lambda \in \mathbb{Z}_{p-1}\} < G$$

Then f is constant and distinct on cosets of K and generator $(y, 1)$ of K gives $y = \log_g x$

Graph Problems:
So we can solve problems like those above where G is abelian, but we can also solve graph problems.

Consdier graph $A = (V, E)$, $|V| = n$ lets say that the graph is undirected and there is at most one edge between any two vertices. Vertices here are labelled by numbers from 1 to n.

Lets define an adjacency matrix M_A: $[M_a]_{ij} = \begin{cases} 1 & \iff (i,j) \\ 0 & \text{otherwise} \end{cases}$. The permuation group of $[n]$, $|P_n| = n!, \log|P_n| \sim O(n \log n)$. Define a group of automorphisms of group A which is a set of permuations with the following property: $\pi \in P_n$ s.t. $\forall i, j (i, j)$ is an edge in $A \iff (\pi(i), \pi(j))$ is also an edge in A.

An associated HSP (the case of non-abelian G):

$$G = P_n, X = \text{set of all labelled graphs on } n\text{vertices}$$

For any $A \in X$, define $f_A : G \to X$, $f_A(\pi) =$"A with vertex labels permuted by π"

$$K = Aut(A)$$

(Check $f(K)$ is constant and disctint on cosets of $Aut(A)$)

Applications:
If we can sample uniformly from K, then we can solve Graph Isomorphism problem (GI).This has a number of different applications in areas of computer science. Two labelled graphs A and B with n vertices are isomorphic if there is a 1-1 map (i.e. permutation) $\pi[n] \to [n]$ s.t. $\forall i, j \in [n](i, j)$ is an edge in $A \iff (\pi(i), \pi(j))$ is an edge in B. The GI problem is given to graphs A and B and deciding if they are isomorphic. This can be represented as a non-abealian HSP. There is no known poly(m) time classical algorithm to solve this problem, so GI is clearly in NP but not believed to be NP-complete (a class of problems such that every problem in NP can be reduced to an NP-complete problem these are the hardest NP problems). In 2017, L Babai presented a quasi-polynomial algorithm for GI runtime $n^{O}(\log n)^2)$. This ranks in between polynomial runtime and exponential algorithms.

3 Lecture 4

Quantum algorithm for finite abelian HSPs - Generalisation of period-finding algorithm
Write our abelian group $(G, +)$ additively

Construction of shift-invariant states and Fourier transform for G.
Representations of abelian G:
Consdier the mapping $\chi : G \to \mathbb{C}^* = \mathbb{C} - \{0\}$ with multiplication that satisfies:

$$\chi(g_1 + g_2) = \chi(g_1)\chi(g_2), \forall g_1, g_2 \in G$$

χ is a group homomorphism from G to $\mathbb{C}^*$. Such χ's are called irreducible representations of G. They have the following properties: **Theorem 1**:
1) any value $\chi(g)$ is a $|G|$-th root of unity ($\chi \in \to S^1$ the unit circle)
2) Schur's lemma (orthogonality): If χ_i, χ_j satisfy (HOM) then

$$\frac{1}{|G|} \sum_{g \in G} \chi_i(g)\bar{\chi_j}(g) = \delta_{ij}$$

3) There are always exactly $|G|$ different functions χ satisfying (HOM).

Examples: $\chi(g) = 1, \forall g \in G$ is an irrep/ called a trival irrep
Label the trivial irrep as χ_0, $0 \in G$. Then for any other irrep $\chi \neq \chi_0$ orthonality to χ_0 gives:

$$\sum_{g \in G} \chi(g) = 0 \text{ if } \chi \neq \chi_0$$

Going back to constructing shift-invariant states

3.0.1 Shift-invariant states

Consider a state space $\mathcal{H}_G, dim\mathcal{H}_G = |G|$ wtih basis $\{|g\rangle\}_{g \in G}$. Now introduce shift operators $U(k)$ for $k \in G$ defined as follows:

$$U(k) : |g\rangle \to |g + k\rangle \, , g, k \in G$$

All shift operators commute so there exists a simultaneous eigenbasis.
For each $\chi_k, k \in G$:

$$|\chi_k\rangle = \frac{1}{\sqrt{|G|}} \sum_{g \in G} \bar{\chi_k}(g) \, |g\rangle$$

By thereom 1 $\{\chi_k\}$ form an orthonormal basis.

$$U(g) \, |\chi_k\rangle = \chi_k(g) \, |\chi_k\rangle$$

Proof:

$$U(g) \, |\chi_k\rangle = \frac{1}{\sqrt{|G|}} \sum_{h \in G} \bar{\chi_k}(h) \, |h + g\rangle$$

$$\{h' = h + g\} = \frac{1}{\sqrt{|G|}} \sum_{h' \in G} \bar{\chi_k}(h' - g) \, |h'\rangle$$

using HOM $\chi_k(-g) = \chi_k(g)^{-1} = \bar{\chi}_k(g) \implies \chi_k(\bar{h'-g}) = \bar{\chi}_k(h')\bar{\chi}_k(-g) = \bar{\chi}_k(h')\chi_k(g)$. Therefore,

$$U(g)\left|\chi_k\right\rangle = \frac{1}{\sqrt{|G|}}\sum_{h'\in G}\chi_k(g)\bar{\chi}_k(h')\left|h'\right\rangle = \chi_k(g)\left|\chi_k\right\rangle$$

So $|\chi_k\rangle$'s form a common eigenbasis

Introduce Fourier transform QFT for a group G
- consider a unitary mapping on $\mathcal{H}_G$ mapping $|\chi_k\rangle$ basis to $|g\rangle$ basis

$$QFT\left|\chi_g\right\rangle = \left|g\right\rangle, \forall g \in G$$

$$QFT^{-1}\left|g\right\rangle = \left|\chi_g\right\rangle$$

k-th column of QFT^{-1} in $|g\rangle$ basis is mode of components of $|\chi_k\rangle$:

$$[QFT^{-1}]_{gk} = \frac{1}{\sqrt{|G|}}\bar{\chi_k}(g)$$

Example: $G = \mathbb{Z}_M$L
Check $\chi_a(b) = e^{\frac{2\pi iab}{M}}, a, b \in \mathbb{Z}_M$ satsifies HOM and has its irreps labelled by $a \in \mathbb{Z}_M$ with $\chi_0(b) = 1 \forall b \in \mathbb{Z}_m$.

$$G = \mathbb{Z}_{M_1} \times ... \times \mathbb{Z}_{M_r}$$

$$(a_1, ..., a_r) = g_1, (b_1, ..., b_r) = g_2$$

$$\chi_{g_1}(g_2) = e^{2\pi i(\frac{a_1b_1}{M_1}+...+\frac{a_rb_r}{M_r}}$$

This satifies HOM and our $QFT_G = QFT_{M_1} \otimes ... \otimes QFT_{M_r}$ on $\mathcal{H}_G = \mathcal{H}_{M_1} \otimes ... \otimes \mathcal{H}_{M_r}$.

This second example is exhaustive since we have a classification theroem:
Classification theorem: Any fintie abelian group G is isomorphic to a direct product of the form $G = \mathbb{Z}_{M_1} \otimes ... \otimes \mathbb{Z}_{M_r}$. So M_1 can be taken in a form $p_1^{s_1}, ...p_r^{s_r}$.

3.0.2 Quantum algorithm

$$f : G \to X$$

with hidden subgroup K and cosets $k = 0 + k, g_2 + k, ...g_m + k$, $m = \frac{|G|}{|K|}$. we will work on $\mathcal{H}_{|G|} \otimes \mathcal{H}_{|X|}$,$\{\left|g\right\rangle \left|x\right\rangle\}_{g\in G, x\in X}$.

Create a state $\frac{1}{\sqrt{|G|}}\sum_{g\in G}\left|g\right\rangle\left|0\right\rangle$
*Apply*U$_f$ and $\frac{1}{\sqrt{|G|}}\sum_{g\in G}\left|g\right\rangle\left|f(g)\right\rangle$

Measure the second register to get $f(g_0)$. The first register will not give the coset state:

$$|g_0 + k\rangle = \frac{1}{\sqrt{|k|}} \sum_{k \in K} |g_0 + k\rangle = U(g_0) |K\rangle$$

apply QFT and measure to get a result $g \in G$

4 Lecture 6

We can write $|K\rangle$ in the shift-invariant basis $\{\chi_g\}_{g \in G}$

$$|K\rangle = \sum_g a_g |\chi_g\rangle$$

$$|g_0 + K\rangle = U(g_0) |K\rangle = \sum_g a_g \chi_g(g_0) |\chi_g\rangle$$

as $QFT |\chi_g\rangle = |g\rangle$ so after we apply QFT

$$prob(g) = |a_g \chi_g(g_0)|^2 = |a_g|^2, |\chi_g(g_0)| = 1$$

$$QFT |K\rangle = \frac{1}{\sqrt{|G|}} \frac{1}{\sqrt{|K|}} \sum_{l \in G} (\sum_{k \in K} \chi_l(k) |l\rangle)$$

$\sum_{k \in K} \chi_l(k) |l\rangle$ involves irreps χ_l of G restricted to subgroup $K < G$, and each such object is itself an irrep in K. Hence we have the following relation:

$$\sum_{k \in K} \chi_l(k) = \begin{cases} |k| & if \chi_l \text{ restricts to the trival irrep of K} \\ 0 & \text{otherwise} \end{cases}$$

$$QFT |K\rangle = \sqrt{\frac{|K|}{|G|}} \sum_{l \in G} |l\rangle$$

Then a measurement gives a uniformly random choice of l s.t. $\chi_l(k) = 1$.
If k has generators $k_1, ..., k_n$ where $M = O(\log(K)) = O(\log |G|)$. Then the output of a measurement gives us $\chi_l(k) = 1 \forall i$.

It can be shown that if $O(\log(|G|)$ values of l chosen uniformly at random then with probability $> \frac{2}{3}$ they will suffice to determine a generating set for k via the equations $\chi_l(k) = 1$.

Example: $G = \mathbb{Z}_{M_1} \times ... \times \mathbb{Z}_{M_l}$
$l = (l_1, ..., l_q) \in G$, $g = (b_1, ..., b_q) \in G$ gives $\chi_l(g) = e^{2\pi i (\frac{l_1 b_1}{M_1} + ... + \frac{l_q b_q}{M_q}}$
For $k = (k_1, ..., k_q) \in K$ with $\chi_l(k) = 1 \implies \frac{l_1 k_1}{M_1} + ... + \frac{l_q k_q}{M_q} = 0 mod 1$. This is a homogenous linear equation on k and $O(\log(k))$ such equations determine k as null space.

4.0.1 Remarks on HSP for non-abelian groups G

Now we will consider multiplicative shifts. As before we can generate a bunch of coset states but it is curious to investigate what breaks down.

$$|g_0 K\rangle = \frac{1}{\sqrt{|K|}} \sum_{k \in K} |g_0 k\rangle \, , g_0 \in G \text{ is chosen randomly}$$

The real problem with QFT construction is that there is no good basis of shift invariant states. This is because $U(g_0)$ don't commute.

Construction of non-abelian QFT
Consider a d-dimensional representation of G and a group homomorphism $\chi : G \to U(d)$
χ is irreducible if no subset of $\mathbb{C}^d$ is left invariant by all matrices $\chi(g), g \in G$. (i.e. we cannot simulatenoulsy block-diagonalize all of $\chi(g)$'s by a simple basis change)
Let's define a complete set of irreps. It is a set $\chi_1, ..., \chi_m$ s.t. that any irrep is unitarily equivalent to one of them. e.g. $\chi \sim \chi' = V \chi C V^{-1}, V \in U(d)$
Example: G is abelian , all irreps have $d = 1$, since all $\chi(g)$ commute. Theorem(non-abelian analogue of Theorem 1) (consullt Fulton and Hardes "Representation Theory" fo more information)
If $d_1, ..., d_m$ are the dimensions of a complete set of irreps $\chi_1, ..., \chi_M$ then:
1) $d_1^2 + ... + d_m^2 = |G|$
2) $\chi_{i,jk}(g)$ is (j,k) th matrix entry of $\chi_i(g)$ then by Schur orthogonality:

$$\sum_g \chi_{i,jk}(g) \bar{\chi}_{i',j'k'}(g) = |G| \delta i i' \delta_{jj'} \delta k k'$$

Now if we look at the states that correspond to these irreps $\chi_{i,jk} = \sum_g \in G \bar{\chi}_{i,jk}(g) \, |g\rangle$ they form an orthonormal basis.
QFT on G is defined to be a unitary rotation between two basis of $\{\chi_{i,jk}\}$ basis $\to \{|g\rangle\}_{g \in G}$.
These takes $|\chi_{i,jk}\rangle$ are not shift-invariant for all $U(g_0)$ so this implies that measuring coset state $|g_0 k\rangle$ in the $\{|\chi\rangle\}$ basis results in an output distribution that is not independant of g_0.
A "partial" shift-invariance survives. Consider a measuremnt M_{rep} on $|g_0 k\rangle$ this measurement will only distinguish the irreps (i values) and not all (i,j,k)'s. the outcome i will be associated with d_i^2 dimensional orthogonal subspaces that are spanned by $\{\chi_{i,jk}\}_{j,k=1}^{d_i}$.
Then $\chi_i(g_1 g_2) = \chi_i(g_1)\chi_i(g_2) \implies$ the output distribution of i values is indeed independant of g_0.
So this gives us direct (but incomplete) information about k. For instance, conjugate subgroups k and $L = g_0 K g_0', g_0 \in G$. This measurement will give us the same statistics.
M_{rep} will result in the same output statistics
Not everything is lost there are some cases when this information is enough.

The reason HSP is good in the abelian case is we have an efficent QFT transform. In other words QFT can be implemented in $poly(\log(|G|)$ times. This is true for abelian groups and some non-abelian groups (e.g. P_n).
Some partial results:
For normal subgroups $gk = kgg \in G$ we have a theorem proven by Hall green, russel Ta shma in 2003 SIAM J Comp 32, p 916-934:
Suppose G has QFT that is efficently implementable. Then if a hideen subgroup k is a normal subgroup, then there is an efficent quantum HSP

Theorem(Edinsguin, Hoyer, Knill, 2004)
For general non-abelian HSP, $M = O(poly \log(|G|))$ then random coset states $|g_1 k\rangle$,..., $|g_n k\rangle$ suffice to determine k . But it is not known how to efficently determine k from the M coset states.

5 Lecture 6

5.1 Phase estimation Algorithm

- Unifying principle for quantum algorithms based on QFT
- also gives an alternative way of factoring (orginally discovered by Kitaev)
The fact the phase estimation algorithm became so wide spread and you can cast every algorithm that is tangenetially ralted to QFt in QPE
Givem a unitary operator U. and eigenstates $|v_\phi\rangle = U|v_\phi\rangle = e^{2\pi i\phi}|v_\phi\rangle$
Want to estiamte the phase ϕ $0 < \phi < 1$ (up to n bits of precision $\phi = 0.i_1 i_2, ...i_n = i_1/2 + i_2/4 + ...$ for any given n)
We will have to implement controlled unitary opearotrs and in particular we will need Controlled-U^k for integers k

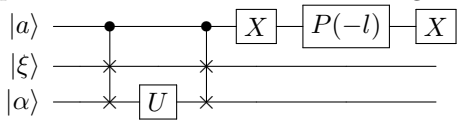
$$C - U^k |0\rangle |\xi\rangle = |0\rangle |\xi\rangle, C - U^k |1\rangle |\xi\rangle = |1\rangle U^k |\xi\rangle$$

$|\xi\rangle$ has a general dimension d:

$$U^k |v_\phi\rangle e^{2\pi ik\phi} |v\phi\rangle, C - U^k = (C - U)^k$$

If we are given U as a circuit description, we can easily implement $C - U$ by controlling each gate in U's circuit. However, if U is given as black box (e.g. a physical operation in the lab) we need further information as there is an inherent ambiguity as we have to account for local phase $e^{i\theta U}$ as it has no effect normally unless you use a controlled operation.

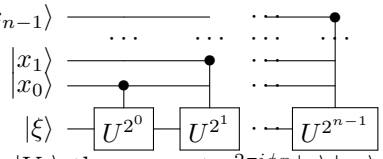
If the unitary is specified in this ambiguous way we need to figure out what to do. It suffices to have an eigenstate $|l\rangle$ with a known eigenvalue $U|l\rangle = e^{i\alpha}|l\rangle$ then $e^{i\theta}U$ will map $\alpha \to \alpha + \theta$. Consider the following circuit:



with $P(-l) = \begin{pmatrix} 1 & 0 \\ 0 & e^{-il} \end{pmatrix}$. This correctly gives $C-U$. We'll want a "generalised controlled-U" that gives:

$$|x\rangle |\xi\rangle \rightarrow |x\rangle U^x |\xi\rangle \, x \in \mathbb{Z}_{2^n}$$

For $x = x_{n-1}...x_1x_0 = 2^0x_0 + 2^1x_1 + 2^2x_2... + 2^{n-1}x_{n-1}$:



If we input $|\xi\rangle = |V_\phi\rangle$ then we get $e^{2\pi i\phi x} |x\rangle |v_\phi\rangle$. Now superpose over all $x = 0, 1, 2, ..., 2^{n-1}$ by applying hadamards to all the qubits before applying the circuit, take $|\xi\rangle = |v_\phi\rangle$:

This gives output $|A\rangle = \frac{1}{\sqrt{2^n}} \sum_x e^{2\pi i\phi x} |x\rangle$. Applying QFT^{-1} to $|A\rangle$ and measure. We get $y_0y_1...y_{n-1}$. Then output the number $0.y_1y_2...y_{n-1} = \frac{y_0}{2} + \frac{y_1}{4} + ... + \frac{y_{n-1}}{2^n}$ as an estimate of ϕ.

Now lets assume an idealised situtation where ϕ has only n binary digits:

$$\phi = 0.z_1...z_{n-1}$$

Then $\phi = \frac{z}{2^n}$ where z is an n-bit integer in $\mathbb{Z}_{2^n}$:

$$|A\rangle = \frac{1}{\sqrt{2^n}} \sum_2 e^{2\pi i 2^n z/2^n} |x\rangle$$

is a QFT of $|z\rangle$. Applying $QFT^{-1} |A\rangle = |z\rangle$ and we get ϕ exactly with certainity.

Note the algorithm up to the final measurements is a unitary operation mapping:

$$|0\rangle |0\rangle ... |0\rangle |v_\phi\rangle \rightarrow |z_0\rangle |z_1\rangle ... |z_{n-1}\rangle |v_\phi\rangle$$

If ϕ has more than n bits $\phi = 0.z_0z_1...z_{n-1}|z_nz_{n+1}$.
Theorem (PE): If measurements in the algorithm give $y_0y_1...y_n$ and the aoutput $\Theta = 0.y_0y_1...y_{n-1}$ then :
a) Prob (Θ is closeset n-binary digit approx to ϕ) $\geq \frac{4}{\pi^2} = 0.4$
b) Prob($|\Theta - \phi| \geq \epsilon$) $\leq O(\frac{1}{2^n\epsilon}$

We will show that Prob($|\Theta - \phi| \geq \epsilon$) $\leq \frac{1}{2^{n+1}\epsilon}$)

6 Lecture 8

Today we will prove the Phase estimation theorem. We need to change the defintion of distance as we need a distance on a circle.

Define $d(\theta,\phi) = \min\{|\theta-\phi|, |1+\phi-\theta|, |1+\theta-\phi|\}$ which is the distance on the circle. Lets consider the normal binary expansion 0.999999 the closest string should be 1.

Theorem (Phase Estimation): If the output of PE algorithm with n lines (initited in as zeros) is $\theta = 0.y_0y_1...y_{m-1}$, then:
a) Prob (θ is closeset n-binary digit approx to ϕ, $d(\theta,\phi) \leq \frac{1}{2^{n+1}}) \geq \frac{4}{\pi^2} = 0.4$
b) Prob($|\theta-\phi| \geq \epsilon) \leq O(\frac{1}{2^n\epsilon})$ for ϵ fixed

Recall: the output is obtained by measuring an n-qubit state $QFT^{-1}|A\rangle$ where $|A\rangle = \frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1} e^{2\pi i l x}|x\rangle$

$$QFT^{-1}|x\rangle = \frac{1}{\sqrt{2^n}}\sum_{y=0}^{2^n-1} e^{-2\pi i \frac{yx}{2^n}}|y\rangle$$

Soon we will change the notation to make sure it is not overloaded with these powers of 2^n

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

Let $\{\phi = 2^n\psi\}$:

$$QFT^{-1}|A\rangle = \frac{1}{2^n}\sum_{y=0}\sum_{x=0} e^{2\pi i\frac{\phi-y}{2^n})x}|y\rangle = \frac{1}{2^n}\sum_y \frac{1-e^{2\pi i(\phi-y)}}{1-e^{2\pi i\frac{\phi-y}{2^n}}}|y\rangle$$

In the case $\phi - y \neq 0$:

$$Prob(seey) = \frac{1}{2^n}|\sum_{x=0}^{2^n-1} e^{2\pi i\frac{(\phi-y)}{2^n}x}|^2 = \frac{1}{2^{2n}}\frac{|1-e^{2\pi i(\phi-y)}|^2}{|1-e^{2\pi i\frac{\phi-y}{2^n}}|^2} = \frac{1}{2^{2n}}\frac{|1-e^{2\pi i(\psi-\frac{y}{2^n}2^n}|^2}{|1-e^{2\pi i(\psi-\frac{y}{2^n}}|^2}$$

As $\theta_y = \frac{y}{2^n}$ and observe that $|1-e^{2\pi i(\psi-\theta_y)}|^2 = |1-e^{2\pi i d(\psi,\theta_y)}|^2$ therefore:

$$Prob(see\ y) = \frac{1}{2^{2n}}\frac{|1-e^{2\pi i 2^n d(\psi,\theta_y)}|^2}{|1-e^{2\pi i d(\psi,\theta_y}|^2}$$

As $0 < d(\psi,\theta_y) \leq \frac{1}{2}$ we will ise the following bounds:
i) $|1-e^{i\alpha}| \leq 2|$
ii) $|1-e^{i\alpha}| \leq |alpha|$
iii) For $|alpha| \leq \phi$ $|1-e^{i\alpha}| = 2|sin(\frac{\alpha}{2})| \geq \frac{2|alpha|}{\pi}$

The last of thse comes from the fact that for positive α we hae $\sin(\alpha/2) \geq \frac{\alpha/\pi}{}$

When $d(\psi,\theta) \leq \frac{1}{2^{n+1}}$ implies that y is the best approxiamtion for ψ and $2\pi d(\psi,\theta_y)2^n \leq \frac{2^{n+1}}{2^{n+1}}\pi$ so:

$$Prob(yisbestapproximation) \geq \frac{1}{2^{2n}}|\frac{2}{\pi}\frac{2\pi d(\psi,\theta_y)}{2\pi d(\psi,\theta)}|^2 = \frac{4}{\pi^2}$$

The calculations for the above will be on the moodle

Further remarks:

If $C - U^{2^n}$ is implement as $(C-U)^{2^\alpha}$, then PE algorithm needs exponential time $(1 + 2 + ... + 2^{n-1} = 2^n - 1)$. But for some some U implementing $C - U^{2^k}$ requries only polynomial time so we get a poly-time PE algorithm. Harks back to the algorithm for finding powers by repeated squaring, expressing the exponent in binary and then doing repeated squaring. The number of applications of controlled unitaries does not depend on d the dimension of the space. This can be used to rpovide an alternative factoring algorithm (due to A.Kitaev) (see example sheet).

In many applications we feed an arbitary state to the last register rather than an eigenstate. If instead of $|v_\phi\rangle$ we input general state $|\xi\rangle$, expand in eigenbasis of U:

$$|\xi\rangle = \sum_j c_j |v_{\phi_j}\rangle, U |v_{\phi_j}\rangle = e^{2\pi i \phi_j} |v_{\phi_j}\rangle$$

Then we get (before the final measuremnt) a unitary process U_{PE}:

$$|00..00\rangle |\xi\rangle \rightarrow^{U_{PE}} \sum_j c_j |\psi_j\rangle |v_{\phi_j}\rangle$$

The Born rule implies that the final measurement will give a choice of ϕ_j's (or an approximation) it can be choosen with probability $|c_j|^2$. This is not some average of the ϕ_j values.

Will be elaborated more in the notes on moodle on the following:

If you want to have n-qubits and want to get m-bits correctly probablility of success $1 - \eta$, then must have :

$$n \geq m + \log \frac{1}{\eta}$$

6.1 Amplitude amplification

Much like when we multiple up to HSP we revisited shors alogrithm, in this case we revisit grovers algorithm. This is an apotheosis of technique in Grover's algorithm.

6.1.1 Background

Reflection Operators: State $|\alpha\rangle$ in $\mathcal{H}_d$, n-dim subspace L_α with $(d-1)$ dim orthogonal l_α^α

$$I_{|\alpha\rangle} = I - 2 |\alpha\rangle \langle\alpha|$$

$$I_{|\alpha\rangle} = -|\alpha\rangle$$
$$I_{|\alpha\rangle}|\beta\rangle = |\beta\rangle$$

for any $|\beta\rangle |\alpha\rangle$

For any unitary U: $UI_{|\alpha\rangle}U^\dagger = I_{U|\alpha\rangle}$, $U|\alpha\rangle\langle\alpha|U^\dagger = |\beta\rangle\langle\beta|$ for $|\beta\rangle = U|\alpha\rangle$.

Consider a k-dimensional subspace $A < \mathcal{H}_d$, any orthonormal basis $|a_1\rangle, ..., |a_k\rangle$. Lets consider a projection operator on to this subspace:

$$P_A = \sum_{i=1}^{k} |a_i\rangle\langle a_i|$$

Define a generalised projeciton operator:

$$I_A = I - 2P_A$$

$$I_A|\xi\rangle = \begin{cases} |\xi\rangle & |\xi\rangle \in A^\dagger \\ -|\xi\rangle & |\xi\rangle \in A \end{cases}$$

Now lets recall what grover does very briefly (have a look at Part II course):

Search for a unique "goal" item in unstructured database of $N = 2^n$ items.

Write B_n = set of all n-bit strings. Given an oracle for $f : B_n \to B_n$ with the promise that there is a unique element $x_0 \in B_1$ with $f(_0) = 1$. Problem is to find x_0.

Closely related to class NP and Boolean satisfiability problem. This is explained in part II lecture notes.

7 Lecture 8

Recap of grovers algorithm We are searching for a unique "good" element in an unstructured database, $N = 2^n$ items
We are given an oracle f that maps from $B_n \to B_1$
Promise: There is a unique $x_0 \in B_n$ with $f(x_0) = 1$
Problem: Find x_0

Consider Grover iteration operator on n qubits

$$Q = -H_n I_{|0\rangle} H_n I_{|x_0\rangle} = -I_{|\psi_0\rangle} I_{|x_0\rangle}$$

where $H_n = H \otimes ... \otimes H$, $|\psi_0\rangle = H_n|00.000\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in B_n} |x\rangle$

One application of Q uses 1 query of U_f

Thereom (Grover '96): In 2-dim span of $|\psi_0\rangle$ and (unknown) $|x_0\rangle$ the action of Q is rotation by angle 2α where $\sin\alpha = \frac{1}{\sqrt{N}}$. Hence grover's algorithm to find x_0 given U_f is:
Make $|\psi_0\rangle$

Apply Q m times where $m = \frac{\arccos\frac{1}{\sqrt{N}}}{2} \arcsin\frac{1}{\sqrt{N}}$ to rotate ψ_0 very close to x_0 (within the angle $\pm\alpha$

Measure to see x_0 with high probability $1 - \frac{1}{N}$ For large N $\arccos\frac{1}{\sqrt{N}} = \frac{\pi}{2}$ and $\arcsin\frac{1}{\sqrt{N}} = \frac{1}{\sqrt{N}}$ so $m = \frac{\pi}{4}\sqrt{N}$ iteractions or queries to U_f needed.

Classically we need $O(N)$ quereies to find x_0 with any constant probability that does not depend on N, so this achieves a quadratic speed-up.

7.1 Amplitude Amplification

Let G be any subspace ('good subspace') of state space $\mathcal{H}$ G be its orthogonal complement ('bad subspace') $\mathcal{H} = G \oplus G$

Given any $|\psi\rangle \in \mathcal{H}$, we have unique decomposition with real positive coeffi-centt $|\psi\rangle = \sin\theta\,|g\rangle + \cos\theta\,|b\rangle\,, |g\rangle \in G, |b\rangle \in G$
Introduce reflection operators that flip $|\psi\rangle$ and good vectors:

$$I_{|\psi\rangle} = I - 2\,|\psi\rangle\,\langle\psi|\,, I_G = I - 2P_G$$

$\sin\theta = ||P_G\,|\psi\rangle\,||$ =length of good projection of $|\psi\rangle$
Introduce $Q = -I_{|\psi\rangle} I_G$

7.1.1 Amplitude Amplification Theorem

In the 2-dim space spanned by $|g\rangle$ and $|\psi\rangle$ Q is roation by 2θ where
$\sin\theta$ =length of good projection of $|\psi\rangle$
Proof: We have $I_G\,|g\rangle = -\,|g\rangle\,, I_G\,|b\rangle = |b\rangle$:

$$Q\,|g\rangle = I_{|\psi\rangle}\,|g\rangle\,, Q\,|b\rangle = -I_{|\psi\rangle}\,|b\rangle$$

$$I_{|\psi\rangle} = I - 2(\sin\theta\,|g\rangle + \cos\theta\,|b\rangle)(\sin\theta\,\langle g| + \cos\theta\,\langle b|)$$

using the fact that $\langle b|\,|g\rangle = 0, \langle g|\,|g\rangle = \langle b|\,|b\rangle = 1$:

$$Q\,|b\rangle = \cos 2\theta\,|b\rangle + \sin 2\theta\,|g\rangle$$

$$Q\,|G\rangle = -\sin 2\theta\,|b\rangle + \cos 2\theta\,|g\rangle$$

So

$$Q\,|g\rangle = I\,|g\rangle - 2\sin^2\theta\,|g\rangle - 2\sin\theta\cos\theta\,|b\rangle = \cos 2\theta\,|g\rangle - \sin 2\theta\,|b\rangle$$

In the $\{|b\rangle, |g\rangle\}$ basis, the Q matrix is a rotation matrix by 2θ:

$$Q = \begin{pmatrix} \cos 2\theta & \sin 2\theta \\ -\sin 2\theta & \cos 2\theta \end{pmatrix}$$

As we apply Q n $timesweget Q^n |\psi\rangle = \sin(2n+1)\theta |g\rangle + \cos(2n+1)\theta |b\rangle$. $If we measure Q^n$ in $\{|b\rangle, |g\rangle\}$ basis we have probabiliyt of seeing a good element of $\sin^2(2n+1)\theta$ so this is maximised when $(2n+1)\theta = \frac{\pi}{2}$. $So for the nearest integer to n = \pi \frac{}{4\theta - \frac{1}{2}}$ we will be within θ of the element.

Example: If we have $\theta = \frac{\pi}{6}, n = 1$ we can seee that Q^1 rotates $|\psi\rangle$ exactly onto $|g\rangle$.

Generally, for a given θ n is not an integer, so we use n = nearest integer to $\frac{4\pi}{\theta} - 1 \approx \frac{4\pi}{\theta} = O(\frac{1}{\theta}) = O(\frac{1}{\sin\theta}) = O(\frac{1}{\text{length of good projection of } |\psi\rangle}$ and then $Q|\psi\rangle$ will be witin angle θ of $|g\rangle$ so the probability of seeing a good value is: $P \geq \cos\theta = 1 - O(\theta^2)$.

All this can implement if $I_{|\psi\rangle} I_G$ can be impmeneted efficently. see ES2.

For I_G is suffices for G to be spanned by computational basis states and have an indicator funciton f.

$$f(x) = 1, x if x \text{ is good}, f(x) = 0 \text{ if} x \text{is bad}$$

For $I_{|\psi\rangle}$ we usually have $|\psi\rangle = H_n |00..000\rangle$, then $|\psi\rangle$ can be implemented in $O(n)$ time where n is the number of qubits.

In the amplitude amplification algorithm the relative amplitudes of good elements remain the same as they were in $|\psi\rangle = \sin\theta |g\rangle + \cos\theta |b\rangle$ so $|g\rangle$ remains the same just the amplitude varies. So AA amplifies overall $|g\rangle$ amplitude at the expense of reducing the amplitude of $|b\rangle$.

Second remark: Final state is generally not exactly $|g\rangle$, however, if $\sin\theta$ is known then there is a modification of this algorithm that uses a modest amount of resources t omake it exact.

The rotuine is useful for state prepation e.g.

$$\sum_{x<N, \text{x is coprime to N}} |x\rangle$$

8 Examples Class 1

Try to prove the same thing in 1(ii) in shors algorithm and it will be slightly nicer language than group theory.

For $\mathbb{Z}_2$ the irreps are $\chi_a(x) = (-1)^{ax}$ for $a, x \in \mathbb{Z}_2$ therefore $\mathbb{Z}_2^n$ has irreps $\chi_a(x) = (-1)^{a_1x_1+a_2x_2+...+a_nx_n}$ so

$$|\chi\rangle = \frac{1}{\sqrt{|G|}} \sum_g \bar{\chi}(g) |g\rangle = \frac{1}{\sqrt{|G|}} \sum_{b \in \mathbb{Z}_2^n} (-1)^{ab} |b\rangle$$

and

$$[QFT]_{ab} = \frac{1}{\sqrt{|G|}}(-1)^{ab} = \frac{1}{\sqrt{2^n}}(-1)^{a_1b_1}(-1)^{a_2b_2}...(-1)^{a_nb_n}$$

could be written using Hadmards with $QFT = H \otimes H \otimes ... \otimes H$ as $H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ with the columns being a and the rows being b so it is only -1 if they are both 1 as otherwise it will have a zero in the exponent.

First part of HSP problems is generate coset states and the second part is repeation (how many times you need to repeat the process). In standard HSP we make one query to f to make the following state:

$$|y \oplus k\rangle = \frac{1}{2^k} \sum_{x \in K} |y \oplus x\rangle, y \in \mathbb{Z}_2^n$$

Apply $QFT = H^{\otimes n}$ and measure. This gives an uniformly random output $c \in \mathbb{Z}_2^n$ which is such that the irrep χ_c of G that is restricted to K is the trivial irrep of K. In other words $\chi_c(a) = 1$ for all $a \in K$. In other word, $(-1)^{ac} = 1 \implies ca = 0 \mod 2$. We know that this k viewed as a subspace of $\mathbb{Z}_2^n$ has dimension k. So we need $(n-k)$ linearly indepdendant c_i with $c_i a = 0$ to determine K.

In order to succeed with probability $1 - \epsilon$ when given a constant probability p we run the process some M times. Then you calculate the probablity of M runs failing to determine k and show that for a constant overhead we can get any accuracy. $1 - (1-p)^M > 1 - \epsilon$.

Phase gates act with $P(\alpha) : |0\rangle \to |0\rangle, |1\rangle \to e^{i\alpha} |1\rangle$. Therefore can apply a fractional phase by first preparing a register with $i_1, ..., i_n$ s.t. $y = 0.i_1...i_n$ and then by applying phase gates $P(\frac{1}{2}...P(\frac{1}{2^n}$to get the state $e^{iy} |y\rangle$.

When inventing algorithms whilst it is useful to think about eigenstates to start with make sure to run it through with a general state as you might need to uncompute at the end. e.g. need to apply an inverse controlled unitary in question 5 after computing the correct eigenvalue.

We can't just discard additoinal registers like $\sum_j \lambda_j |u_j\rangle |c_j\rangle$ to $\sum_j \lambda_j |u_j\rangle$ we need to uncompute to remove these extra stuff.

9 Lecture 9

9.1 Amplitude amplification

We have a Hilbert space we can partition into a good part and a bad part: $\mathcal{H} = G_{good} \oplus G_{good}^{\dagger}$, for every $|\psi\rangle$ we have $|\psi\rangle = \sin\theta\,|g\rangle + \cos\theta\,|b\rangle$. We have:

$$I_{|\psi\rangle} = I - 2\,|\psi\rangle\langle\psi|\,, I_G = I - 2P_G, Q = -I_{|\psi\rangle} I_G$$

We proved AA Thereoem: That in the plane spanned by $|b\rangle$ and $|g\rangle$ Q is a rotation by 2θ.

Typically we are given some $|\psi\rangle$ and we use this property to rotate it to $|g\rangle$.

9.2 Applications of Amplitude Amplification

Grover search with one or more (k) good items in N:
Re look over Part II lecture notes and reprove for k good elemetns. Maybe try problem sheet again.

$$|\psi\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in B_n} |x\rangle = \sqrt{\frac{k}{N}}(\frac{1}{\sqrt{k}} \sum_{goodelem} |x\rangle) + \sqrt{\frac{N-k}{N}}(\frac{1}{\sqrt{N-k}} \sum_{bad} |x\rangle)$$

G is spanned by k good $|x\rangle$'s, $\sin\theta = \sqrt{\frac{k}{N}}$ so Q is rotation by 22θ, $\theta = \arcsin\sqrt{\frac{k}{N}} \implies O(\sqrt{\frac{N}{k}})$ queries.

Note that for 2-bit case $N = 4$ and $k = 1$ good element we get $\theta = \arcsin\frac{1}{2} = \pi 6$ and one application of Q rotates $|\psi_0\rangle$ exactly onto $|g\rangle$

Square-root speedy of general quantum algorithms:
Let A be a quantum algorithm/circuit(sequence of 'basic' unitary gates) on input states $|00.00\rangle$.$The final state is$A $|00...000\rangle$.
Good labels = desired computation outcomes.
$A\,|0...0\rangle = \alpha\,|a\rangle + \beta\,|b\rangle\,, \alpha = \sin\theta$ with $|a\rangle$ normalised but generally unequa superposition $\sum_{goodx} c_x\,|x\rangle$.
So probability of success in 1 run is $|\alpha|^2$ so need to do $O(\frac{1}{|\alpha|^2})$ repetitionsof A to succeed with any given const probablity $1 - \epsilon$.

Now lets try amplitude amplification instead of repeated measurement. We need to check we satissify some assumptions.
Assume we can check if the answer is good or bad.
So that we can implement $I_G\,|x\rangle \to \begin{cases} -\,|x\rangle & \text{x is good} \\ |x\rangle & \text{x is bad} \end{cases}$
Consider $|\psi\rangle = A\,|0.00\rangle$

$$Q = -I_{A|00.000\rangle} I_G = -A I_{|0..00\rangle} A^{\dagger} I_G$$

so all parts are implementable

By the amplitude amplifacation theorem Q is a rotation by 2θ with $\sin\theta = |\alpha|$. So after $n = \frac{\pi}{4\theta} = O(\frac{1}{|\theta|}) = O(\frac{1}{\sin\theta}) = O(\frac{1}{|\alpha|}$ repetitions $A|0..0\rangle$ will be rotated very near $|g\rangle$ and the final measuremnt will succeed with high probability. So we get a square root time speed up over a discrete repetition method.

Each application of Q need one A and one $A^\dagger$. and can think of $A^\dagger$ as inverse gates in reverse order so it has a similar complexity to A.

If success probability of A is known then we can do even better and we can cook up the rotation so exactly get to $|g\rangle$ (this exact method will be covered in ES2). This will convert the probabilistic algorithm A into a deterministic one.

Quantum counting:
Here we apply amplitude amplification and phase estimation.
Given $f : B_n \to B_1$ with k good x's we want to estiamte k with (instead of finding a good x).
Let's recall the grover operator: Q_G for f is a rotation by 2θ in this 2D plane spanned by $|\psi_0\rangle = \frac{1}{\sqrt{2^n}}\sum_{x\in B_n}$ and its good projeciton $|g\rangle = \frac{1}{\sqrt{k}}\sum_{goodx}|x\rangle$ with $\sin\theta = \sqrt{\frac{k}{N}} \approx \theta$
Recall that rotations in $\{|b\rangle, |g\rangle\}$ palne have eigenvaleus and eigenvectors:

$$|e_\pm\rangle = \frac{1}{\sqrt{2}}(|b\rangle \pm i|g\rangle)$$

$$\lambda_\pm = e^{\pm 2i\theta}$$

Then check that $|\psi_0\rangle = \sin\theta|g\rangle + \cos\theta|b\rangle = \frac{1}{\sqrt{2}}(e^{i\theta}|e_+\rangle + e^{-i\theta}|e_-\rangle)$
Writing $e^{\pm 2i\theta}$ as $e^{2\pi i\phi_\pm}$ with $\phi_\pm \in (0,1)$:

$$\phi_+ = \frac{2\theta}{2\pi} = \frac{\theta}{\pi}$$

$$\phi_- = \frac{-2\theta + 2\pi}{2\pi} = 1 - \frac{\theta}{\pi}$$

$\frac{\theta}{\pi} = \frac{1}{\pi}\sqrt{\frac{k}{N}}$
Run QPE algorithm with U = "Grover Q" and estimate register set to $|\psi_0\rangle$: will output $\frac{\theta}{\pi}$ or $1 - \frac{\theta}{\pi}$ with probability $\frac{1}{2}$

10 Lecture 10

$$f : B_n \to B_1$$

which has k good x's and we want to estimate k. This is harder than just finding a solution.

$$|\psi_0\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in B_n} |x\rangle \, , |g\rangle = \frac{1}{\sqrt{k}} \sum_{goodx} |x\rangle$$

$$\sin\theta = \sqrt{\frac{k}{N}}$$

for $k << N$ so

$$|\psi_0\rangle = \sin\theta \, |g\rangle + \cos\theta \, |b\rangle = \frac{1}{\sqrt{2}}(e^{i\theta} \, |e_+\rangle + e^{-i\theta} \, |e_-\rangle)$$

Run QPE -¿ $\frac{\theta}{\pi}$ or $1 - \frac{\theta}{\pi}$ with probability $\frac{1}{2}$. We can get approximately θ in either case. For any m, running QPE with m qubit lines will give an m-bit approximation to $\sqrt{\frac{k}{N}}$ this uses 2^n C-Q gates.

By Theroem PE we learn $\sqrt{\frac{k}{N}}$ to an additive error with constant probability $\frac{4}{\pi^2}$ using $O(2^m)$ queries. Write $\frac{1}{2^n}$ as $\frac{\delta}{\sqrt{N}}$ $\delta > 0$. Thus we can learn $\sqrt{k}$ to additive error $O(\delta)$ using $O(2^n) = O(\frac{\sqrt{N}\delta}{)}$ queries.

Classically the same approximation (obtained with constant probability) requires $O(\frac{N}{\delta^2})$ so requires quadratically more effort.

10.1 Hamiltonian simulation

One of the most promising applications of qunatum computers is simulating quantum systems. This is difficult for classical computers as these physical systems have so many degrees of freedom. We won't do the nitty gritty but cover some of the core fundamental ideas and talk about the complexitiy of it.

We want to use a quantum computer to simulate the evolution/dynamics of a quantum system given its Hamiltonian H.
For n qubits generally requires $O(2^n)$ time on a classical computer. We will do it in poly(n) time for a class of Hamiltonian. A lot of good hamiltonains that corrospond to real problems do admit a good representations in quantum computers. First we will focus on evolution and dynamics and then we will move on to studying the ground state properties of these hamiltonians.

10.1.1 Hamiltonain and quantum evolution

Consider a physical system in state $|\psi\rangle$, with hamiltonain H: selfadjoint (Hermitian) operator/matrix - the quantum energy observable. Time evolution given by Schrodinger equation ($\hbar = 1$)

$$\frac{d}{dt} |\psi(t)\rangle = -iH \, |\psi(t)\rangle$$

We will consider time-independant Hamiltonians $H(t) = H$

$$|\psi(t)\rangle = e^{-iHt} |\psi(0)\rangle$$

$$e^A = I + A + \frac{A^2}{2!} + ...$$

Thus given H and time t we want to simulate the action of a unitary $U(t) = e^{-iHt}$ to a suitably good approximation.
Approximation (closeness) of unitary operators Operator norm (spectral norm) $||A|| = \max_{||\psi||=1} ||A|\psi\rangle|| = |\text{maximium eigenvalue—}(if \text{A} is diagonalisable)$.

$$||A + B|| \leq ||A|| + ||B||$$

We say that U approximates $\tilde{U}$ to within error ϵ if $||U - \tilde{U}|| \leq \epsilon$ (for any $|\psi\rangle$ the action of U $\tilde{U}$ are at most ϵ apart.

In general this hamiltonain will be a huge $2^n \times 2^n$ matrix for an $n$qubit system so hard to write down so we will consider a special class of k-local hamiltonains. We will want to simulate $U = e^{-itH}$ with a circuit of poly(n,t) "basic" unitary gates (this is the defintino of efficent simulation on a quantum computer).

Not all H's can be efficently simulated. One such class of efficent hamiltonains is k-local hamiltonains: **k-local Hamiltonain**: H is k-local on n qubits if $H = \sum_{j=1}^{m} H_j$ where H_j is a hermitain matrix acting on at most k qubits. This action does not have to be contiouns a.k.a they don't need to be qubits 1-k it can be any subset of k qubits.

$$H_j = \tilde{H}_j \otimes I$$

with $\tilde{H}_j$ encoding the hamiltonain on the k qubits and the I is on the rest of the system.

$m \leq \binom{n}{k} = O(n^k) = poly(n)$ terms in H. We will see that many imporotant classes of hamiltonains fall into this category e.g.:
1. (3 qubits, 2-local)

$$H = X \otimes I \otimes I - 5Z \otimes I \otimes Y$$

2. Write $M_{(k)}$ to denotes an operator M acting on k qubits and I on the rest. This can correspond to the Ising model on $n \times n$ square lattice of qubits.

$$H = J \sum_{i,j=1}^{n-1} Z_{(i,j)} Z_{(i,j+1)} + Z_{(i,j)} Z_{(i+1,j)}$$

Heisenberg model on a line:

$$H = \sum_{i=1}^{n-1} J_x X_{(i)} X_{(i+1)} + J_y Y_{(i)} Y_{(i+1)} + J_z Z_{(i)} Z_{(i+1)}$$

with J_x, J_y, J_z are real coeff.

Remember that $e^{-\sum_i H_j t} \neq \prod_j e^{-H_j t}$ if H_j don't commute
Also remember that $e^{-H_j t}$ are local unitaries acting on k qubits.

11 Lecture 11

If we want to use some standard universal gate set, then we will invoke the theorem:

Solovay-Kitaev Theorem: Let U be a unitary operator on k (const) qubits, and S be any universal set of quantum gates. Then U can be approximated to within accuracy ϵ using a logarithm sequence of gate $O(\log^c(\frac{1}{\epsilon}))$ gates from S with $c < 4$.
We don't prove this result but it is in Nielsen and Chuang.

Lemma A about accumulation of errors: We need to prove this on ES2. Let $\{\boldsymbol{u}_i\}$, $\{\boldsymbol{v}_i\}$ be sets of m unitary operators such that $||U_i - V_i|| \leq \epsilon$. So if we want to approximate a whole sequence of U_is: $||U_m...U_1 - V_m...V_1|| \leq m\epsilon$ so errors will accumulate linearly. This is kind of the worst case error as it is the maximium error over all $|\psi\rangle$.

Proof: using induction on m.
First lets consider a little warm up with the (easy) commuting case.
Consider the case:

$$H = \sum_{j=1}^{m} H_j, \text{ any k-local Hamiltonian with commuting } H_j$$

Then for any power t, e^{-iHt} can be approximated to within ϵ by a circuit $O(mpoly(\log(\frac{m}{\epsilon})))$ from any given universal set. (Note that $m = \binom{n}{k} = O(n^k)$, this is poly$(n_1 \log \frac{1}{\epsilon})$ too and $\log \frac{1}{\epsilon}$ is the number of digits in the precision of the approximation
Proof: Using SK theorem each $e^{-iH_j t}$ can be approximated to within $\frac{\epsilon}{m}$ with $O(poly(\log \frac{m}{\epsilon}))$ gates.
Lemma A implies the full product $\prod_{i=1}^{m} e^{-iH_j t}$ is then approximated to within $m\frac{\epsilon}{m} = \epsilon$ using a total of $O(mpoly(\log \frac{m}{\epsilon}))$ gates.

The full non-commuting case
For any matrix X we write $X + O(\epsilon)$ for $X + \epsilon$, where $||E|| = O(\epsilon)$
Lemmma B(Lee-Trotter formula):
Let A, B be the following matrices with $||A|| \leq K$ and $||B|| \leq K$ with $K < 1$ (small). Then:

$$e^{-iA}e^{-iB} = e^{-i(A+B)} + O(K)^2$$

Proof: $e^{-iA} = I - iA + \sum_{k=2} \frac{(-iA)^k}{k!} = I - iA + (iA)^2 \sum_{k=0} \frac{(-iA)^k}{(k+2)!}$ we want to show that the sum in the last term has norm $< e^{-K} < 1$. Therefore, $e^{-iA} = I - iA + O(K^2)$ so therefore:

$$e^{-iA}e^{-iB} = (I - iA + O(K^2))(I - iB + O(K^2)) = I - i(A+B) + O(K^2) = e^{-i(A+B)} + O(K^2)$$

Now, apply Lee-Trotter formula repeatedly to accumulate sums of $H_1 ... H_m$ in the exponent
Note that if each term $||H_j|| < K$, then $||\sum_{i=1}^{l} H_i|| < lK$ and we'll want this expression here to be < 1 for all $l \leq m$. For now, we'll assume that $K < \frac{1}{m}$. For now we will also take $t = 1$ (deal with general t later).

$$e^{-iH_1}e^{-iH_2}...e^{-iH_m} = [e^{-i(H_1+H_2)} + O(K^2)]e^{-iH_3}...e^{-iH_m}$$

So as $||H_1 + H_2|| < 2K$ and the error that is error that is generated at each stage stays the same for subsequent unitaries U_i as $||AU|| = ||A||$ therefore:

$$e^{-iH_1}e^{-iH_2}...e^{-iH_m} = e^{-i(H_1+H_2)}e^{-iH_3}...e^{-iH_m} + O(K^2)$$

Then by Lee-Trotter for $(H_1 + H_2)$ and H_3

$$e^{-iH_1}e^{-iH_2}...e^{-iH_m} = [e^{-i(H_1+H_2+H_3)} + O((2K)^2)]e^{-iH_4}...e^{-iH_m} + O(K^2)$$

as $||H_1 + H_2|| < 2K$. We can continue in the same fashion until we get **error estimate 1**:

$$e^{-iH_1}e^{-iH_2}...e^{-iH_m} = e^{-i(H_1+...+H_m)} + O(k^2) + O((2k)^2)) + ... + O(((m-1)k)^2) = e^{-i(H_1+...+H_m)} + O(m^3k^2)$$

For general finite $||H_j||$'s and t values $||H_j t|| < Kt$ (this value can be large). So we introduce a way of breaking down the sequence by introducing N (that we will fix later) s.t. $\frac{H_j t}{N}$ gives $\tilde{K} = ||\frac{H_j t}{N}|| < \frac{Kt}{N}$. We can think about this as dividing the time T up into small $\frac{1}{N}$ intervals and then our unitary has the value:

$$U = e^{i(H_1+...+H_m)t} = (e^{i(\frac{H_1 t}{N}+...+\frac{H_2 t}{N})})^N$$

We want final error for U to be less than ϵ so by lemma A we want the error of each step to be less than $\frac{\epsilon}{N}$. So using error estimate 1 we get

$$Cm^3\tilde{K}^2 < \frac{\epsilon}{N} \implies N > \frac{Cm^3k^2t^2}{\epsilon}$$

Then —— $\mathrm{e}^{-i\frac{H_1 t}{N}}...e^{-i\frac{H_n t}{N}} - e^{-i(H_1 t+...H_n t)}|| < \frac{\epsilon}{N}$. BY Lemma A:

$$||(e^{-i\frac{H_1 t}{N}}...e^{-i\frac{H_n t}{N}})^N - e^{-i(H_1 t+...H_n)}|| < \epsilon$$

So the total circuit size is $O(m^4 \frac{(Kt)^2}{\epsilon})$ we are dealing with Hamiltonians that are k-local. So for n qubits and k-local terms $m = O(n^k)$. Therefore, the total circuit size is $O(n^{4k} \frac{(Kt)^2}{\epsilon})$. You can actually refine this method to get order $t^{1+\delta}$ but they are much more technical and would need many more lectures to explain.

We have a circuit of size $|C| = O(\frac{m^4(kT)^2}{\epsilon})$ and if we want to use gates from the universal set Lecture 12

11.1 The Local Hamiltonaian Problem and QMA

Recap: Defintion of NP. There is no known way to solve it in polynomial time but it is easy to verify the solution in polynomial time.
We will adopt the language of the part II lecture course. A language is in NP if it has an efficent (poly-time) verifier V. A verifier V for a language L is a computation with two inputs w, c s.t. if $w \in L$ then for some c $V(w,c)$ halts with "accept". such c is called a certificate/proof (of membership) for w. If $w \not\in L$, then for all C, $V(w,c)$ halts outputing "reject". V is a poly-time verifier for all pairs (w,c). V runs in poly(n) time $n = |w|$.

11.1.1 The satisficatibility problem (SAT)

Boolean forumlas $\phi(x_1...x_n) : \{0,1\}^n \to \{0,1\}$ and every $(b_1...b_n), b_i \in \{0,1\}$ s.t. $\phi(b_1...b_n) = 1$ is called a satisfying assignment. $V(\phi, c)$ evaluates $\phi(c)$. SAT is not known to be in P.

Theory of NP-completeness shows that many different problems that look very different (SAT, Travelling Salesman Problem, integer linear programming) are essentially the same problem. We can translate instances of one to another in deterministic poly time.

Now lets try to relax some of the requirements on NP. Consider a setting where the prover and verifier may use randomness and sometimes make errors (allowing for some probability of error say $\frac{1}{3}$). In other words when $w \in L$ a prover shoul dbe able to prepare a certificate/proof s.t. Prob($V(w,c)$ accepts) $\geq \frac{2}{3}$ and Prob($V(w,c)$ rejects) $\leq \frac{1}{3}$. This defines complexity class MA (Merlin-Arthur). Merlin is regarded as the omniscient prover and Arthur is a randomised poly time verifier.

Quantum Merlin Arthur class
The direct analgoue of NP.
It is a class of promise problems. A promise problem L partitions $\{0,1\}^k$ of all binary strings into L_0, L_1, L_*.
An algorithm is promised that it never recieves inputs from L_*. However, if the input is from L_0/L_1 then it has to determine 0/1.
QMA: A prmise problem $L = (L_0, L_1, L_*)$ is in the class QMA if there exists a uniform family of circuits $\{C_n\}$ with two input registers and one output qubit and a prolynomial $p()$ s.t. $\forall w \in \{0,1\}^k$
Completeness: If $w \in L_1 \cap \{0,1\}^n$ then there exists a $p(n)$- qubit state $|\psi\rangle$ (a proof/witness state) such that C_n outputs 1 with probability $\geq \frac{2}{3}$ when run on w and $|\psi\rangle$
Soundness: If $w \in L_0 \cap \{0,1\}^n$ then for every $p(n)$ qubit state $|\psi\rangle$ the circuit C_n outputs 1 with probability $\leq \frac{1}{3}$ when run on w and $|\psi\rangle$
QMA : $A prmise problem \mathrm{L} = (\mathrm{L}_0, L_1, L_*)$ is in the class QMA if there exists a uniform family of circuits $\{C_n\}$ with two input registers and one output qubit

and a prolynomial $p()$ s.t. $\forall w \in \{0,1\}^k$
Completeness: If $w \in L_1 \cap \{0,1\}^n$ then there exists a $p(n)$- qubit state $|\psi\rangle$ (a proof/witness state) such that C_n outputs 1 with probability $\geq \frac{2}{3}$ when run on w and $|\psi\rangle$
Soundness: If $w \in L_0 \cap \{0,1\}^n$ then for every $p(n)$ qubit state $|\psi\rangle$ the circuit C_n outputs 1 with probability $\leq \frac{1}{3}$ when run on w and $|\psi\rangle$
Remarks:
1) If we replace $|\psi\rangle$ with a classical bitstring we get the class QCMA
2) If we additionally to (1) force the verifier to be classical MA
3) If we additionally to (1),(2) repalce success probability ($\frac{2}{3}$) by 1 -¿ NP

$$NP \leq MA \leq QCMA \leq QMA$$

SAT is NP-complete (Cook-Levin theorem). Consider a special case: $k-SAT$ ϕ is the conjunction of clauses each of which is a disjunction of k-literals. e.g. for $k=3$:

$$(x_1 \cup \bar{x}_2 \cup x_3) \cap (\bar{x}_1 \cup x_2 \cup x_4) \cap (x_1 \cup \bar{x}_4 \cup x_5)$$

k-SAT is NP-complete for $x \geq 3$ and k-SAT is in P when $k=2$.

We will reformulate $k-SAT$ and relate it to a minimal eigenvalue of a certain hamiltonian which is diagonal in compuational basis.
Fix $k=3$. Consider a claus $C = x_1 \cup \bar{x}_2 \cup x_3$ (has one non-satsifiy assignment 010).

Lets associate a particular diagonal hamiltonian with 0 everywhere but a 1 at localation indexed by the bit string above.
H_c gives "penalty" of 1 to the bitstring of $x_1...x_n$ if x does not satisfy clause C.
We will regard H_c as part of n-qubit Hamiltoanin as $H_c = H_c \otimes I$.
When we evaluate this term: $\langle x| H |x\rangle = 0$ if clause C is satsified or 1 otherwise.

Suppose we have a 3-SAT forumla $\phi = C_1 \cap ... \cap C_n \implies H_\phi = \sum_{j=1}^m H_{c_j}$. The eigenvalues of H_ϕ lie in the interval $[0,m]$ and H_ϕ is 3-local.
Each assignment $x \in \{0,1\}^n$ generates the "energy"
$\langle x| H_\phi |x\rangle = \sum_{j=1}^n \langle x| H_{C_j} |x\rangle$ this counts the number of unsatisfied clauses under x.

We want to find the minimal energy of this hamiltonain which will correspond to the lowest eigenvalue λ_n (the smallest number of unsatsified clauses).

12 Lecture 12

The k-local Hamiltonian problem (LH): Given a classical description of a Hamiltonian of an n qubit Hamiltonian:

$$H = \sum_{j=1}^m H_j$$

where each H_j is k-local and is positive semi-definite. Also given two parameters $a, b \in [0, m]$ with $b - a \geq \frac{1}{poly(n)}$ with a promise that λ_{min}- minimal eigenvalue of H is either $\leq a$ or $\geq b$. Decide which is the case. λ_{min} is often called the ground state energy. Determining λ_{min} ($\leq a$ or $\geq b$) is tantemount to approximating the ground state energy λ_{min} up to an additive error $O(b-a)$. Further studied on ES.

We would like to prove that LH is QMA-complete

We first need to obtain that LH is in QMA: Given the corresponding witness states $|\psi\rangle$ for $x \in L_1$, we can efficently approximate the energy to check if it is $\leq a$ or $\geq b$.
Note: LH is a promise problem, which means that H with $\lambda \in (a, b)$ won't satsify the condition.
Next step, is we want to show that LH is QMA-hard (any other problem in QMA can be reduced to it)

Take any $L = (L_1, L_0, L_*)$ in QMA and fix $x \in L_1 \cup L_0$ which will be our n bit string.
Plan: Consider a circuit C_n and convert it to Hamiltonian H s.t. λ_{min} is small iff C_n has high acceptance probability on some state $|\psi\rangle$. We have $C_n = U_T ... U_1$, each U_i is a 1 or 2 qubit gate. We'll take error probability to be $\frac{1}{4T}$.
The circuit C_n take $n + s + p(n)$ qubits as inputs. n qubits encode the classical input x. s qubits is the workspace of clean qubits in state 0. $p(n)$ is a witness state $|\psi\rangle$.

Given $|\psi\rangle$, define $|\psi_0\rangle = |0\rangle^{\otimes S} |\psi\rangle$, $|\psi_t\rangle = U_t |\psi_{t-1}\rangle$ for $i = 1...T$.

We will define the following Hamiltonian which has three parts.

$$H_{init} = \sum_{i=1}^{s} |1\rangle \langle 1|_i \otimes |0\rangle \langle 0|_C$$

with $i = 1, ..., s + p(n)$

$$H_t = \frac{1}{2}[I \otimes (|t-1\rangle \langle t-1|_C + |t\rangle \langle t|_C) - U_t \otimes |t\rangle \langle t-1|_C - U_t^\dagger \otimes |t-1\rangle \langle t|_C]$$

$$H_{final} = |0\rangle \langle 0| \otimes |T\rangle \langle T|_C$$

$$H = H_{init} + \sum_{i=1}^{T} H_i + H_{final}$$

We should note that H "follows" the state $|\psi\rangle$ and percribes penalties when deviating form $|\psi_0\rangle, |\psi_1\rangle, ..., |\psi_T\rangle$
H also penalizes the zero output in the final measurement
C-is an extra "clock" register of $\log(T+1)$ qubits

Check locality of H is $k = \log(T+1) + 2 = O(\log(n))$. We will further reduce k to a constant $k = 5$.
We need to check that we can distinguish $x \in L_1$ and $x \in L_0$ by looking at λ_{min}.

Consider first case when $x \in L_1$ this means that there exists a $p(n)$ qubit witness state $|\psi\rangle$ that makes C_n accept with probability $1 - \frac{1}{4T}$. Lets consider another state $|\psi'\rangle = \frac{1}{\sqrt{T+1}} \sum_{t=0}^{T} |\psi_t\rangle |t\rangle$ this is a "History state". When we look at this state we want to evaluate the energy. State $|\psi'\rangle$ gets no contribution to penalty from H_{int} and H_t terms. Since the probability of measuring the (incorrect) outocme is $\frac{1}{4T}$ we have $\lambda_{min} = \langle\psi'| H |\psi'\rangle = \frac{1}{T+1} \langle\psi_T| \langle T| H_{final} |T\rangle |\psi_T\rangle \leq \frac{1}{T+1}\frac{1}{4T} = a$.
This will show us completeness.

Now take $x \in L_0$. We want to show that λ_{min} is at least $b = 2a$. Consider any witness state $|\psi'\rangle = \sum_{t=0}^{T} \alpha_t |\phi_t\rangle |t\rangle$ with $\alpha_t \geq 0$ and $\{|\phi_t\rangle\}$ is normalised. Then evaluate

$$\langle\psi'| H |\psi'\rangle = \frac{1}{2}(\alpha_{t-1}^2 + \alpha_t^2 - \alpha_{t-1}\alpha_t \langle\phi_t| U_t |\phi_{t-1}\rangle - \alpha_t\alpha_{t-1} \langle\phi_{t-1}| H |\phi_t\rangle)$$

$$\langle\psi'| H |\psi'\rangle = \frac{1}{2}||\alpha_t |\phi_t\rangle - \alpha_{t-1}U_t |\phi_{t-1}\rangle ||^2$$

We will be making several simplifying assumptions:
1) All α_ts are $\frac{1}{\sqrt{T-1}}$
2) $|\phi\rangle = |0\rangle^{\otimes s} |\psi\rangle$ for some $p(n)$ qubit state $|\psi\rangle$
3) $|\phi_T\rangle$ has accept probability close to 1.

Using these and the fact that $x \in L_0$, $C_n |\phi_0\rangle$ must have accept probability near 0.
From 3 we have $|\phi_T\rangle$ and $C_n |\phi_0\rangle$ must be nearly orthogonal:

$$1 \leq || |\phi_T\rangle - C_n |\phi_0\rangle || = ||sum_{t=1}^{T} U_T - U_{t-1} |\phi_t\rangle - U_T...U_t |\phi_{t-1}\rangle || \leq \sum_{t=1}^{T} ||U_T...U_t |\phi_t\rangle - U_T...U_t |\phi_{t-1}\rangle || = \sum_{t=1}^{T}$$

Using the evaulation of $\langle\psi'| H |\psi\rangle$ assumptions and setting $\alpha_t = \alpha_{t-1} = \frac{1}{\sqrt{T+1}}$:

$$\langle\psi'| H |\psi'\rangle = \sum_{t=1}^{T} \langle\psi'| H_t |\psi'\rangle = \frac{1}{2}\frac{1}{T+1} \sum_{t=1}^{T} || |\phi_t\rangle - U_t |\phi_{t-1}\rangle ||^2 \geq \frac{1}{2T(T+1)}(\sum_{t=1}^{T} || |\phi_t\rangle - U_t |\phi_{t-1}\rangle ||^2) \geq \frac{}{2T($$

13 Example Sheet 2

For Grover's search algorithm, if the indicator function is classically efficently computation then I_G is implementable.

Look at Part II course to recap coprimality conditions. They also show that to

factor N with constant probability it suffices to find an approximations ξ to $\frac{s}{r}$ for random s in range $[1, r)$ to $2m+1$ binary digits of accuracy with $m = O(\log N)$ s.t. $|\xi - \frac{s}{r}| < \frac{1}{2N^2}$

14 Lecture 13

The Hamiltonian is not k-local for k constant it is of $\log N$ we will fix this today.

14.0.1 Making Hamiltonian k-local

What is responsibile for this non-locality effect is the clock-register. So we can reduce locality from $O(\log n)$ to 5 by making the clock unary. For $t = 0$ we will write $|0\rangle^{\otimes T}$ for $t = 1$ we write $|1\rangle |0\rangle^{\otimes T-1}$ for $t = 3$ we write $|11\rangle |0\rangle^{\otimes T}$. When we had a binary format we used $\log(T)$ qubits in the unary format we will use T qubits. This construction will require an extra bit that we will see shortly. Let's crite C_t to denote the t-th qubit of the clock register. Now our hamiltonian H has the following terms:

$$H_{init} = \sum_{i=1}^{s} |1\rangle \langle 1|_i \otimes |0\rangle \langle 0|_{C_i}$$

$$H_t = \frac{1}{2}(I \otimes |100\rangle \langle 100|_{C_{t-1}C_tC_{t+1}} + I \otimes |110\rangle \langle 110|_{C_{t-1}C_tC_{t+1}} - U_t \otimes |110\rangle \langle 100|_{C_{t-1}C_tC_{t+1}} {}^{\dagger}_{U} t \otimes |100\rangle \langle 110|_{C_{t-1}C_tC}$$

$$H_{final} = |0\rangle \langle 0|_T \otimes |1\rangle \langle 1|_{C_T}$$

Need an extra term to penalise clock registers that do not confrom to the unary format:

$$H_{clock} = \sum_{t=1}^{T-1} |01\rangle \langle 01|_{C_tC_{t+1}}$$

$$H = H_{init} + \sum_{t=1}^{T} H_t + H_{final} + H_{clock}$$

So each term accesses at most 5 qubits, so this is a 5 local hamiltonian. What is the smallest k and still retain QMA completeness. k-local Hamiltonian is QMA-complete for $k = 2$.

14.0.2 QMA-complete problems

Fairly exhaustive list in a paper by A Bookatz arxiv 1292.6312. We won't really look through many of them but we will think about a few interesting problems.

Non-identity check: Given a poly(n)-sized quantum circuit U on n qubits. We want to determine whether U is not close to the identity up to some global

phase:
1) accept if $\forall \phi \in [0, 2\pi]$ $||U - e^{i\phi} I|| \geq b$ or
2) reject if there exists $\phi in [0, 2\pi]$ s.t. $||U - e^{i\phi} I|| \leq a$.
This comes with a promise that one of the above options holds and $b - a \geq \frac{1}{poly(n)}$

Excited k-local hamiltonian: For some fixed constant c and given a k-local Hamiltonian H on n qubits. We want to determine whether:
1) ACCEPT: The c-th eigenvalue of H is $\leq a$
2) REJECT: the c-th eigenvalue of H is $\geq b$
Have promise that one of these cases hold and $b - a \geq \frac{1}{(n)}$.

Many Hamiltonians that correspond to real physical systems are QMA-complete so studying the behavoiur of these systems is also hard on a quantum computer. An example of this is a 2-local Ising model $H_{zzxx} = \sum_{i=1}^{n} h_i Z_i + \sum_{i=1}^{n} \Delta_i X_i + \sum_{i,j=1}^{n} J_{ij} Z_i Z_j + \sum_{i,j=1}^{n} K_{ij} X_i X_j$ for real h_i, Δ_i, J, K. Also true for 2D Heisenberg model or the Rose-Hubbard model.

Group non-membership: Problem that is in QMA, but not known to be QMA-complete:
Given a finite group G, subgroup $H < G$, an element $g \in G$ and determine whether: 1) $g \not\in H$
2) $g \in H$

14.1 Harrow-Hassidian-Lloyd (HHL)

Quantum algorithm for solving linear systems of equations. We want to solve the linear system of equations: $A\boldsymbol{x} = \boldsymbol{b}$, $\boldsymbol{b}, \boldsymbol{x} \in \mathbb{C}^N$ where dimensions N is potentially very large $N = 2^n$ or let 2^n be the least power of 2 that is greater than N. Rather than outputting the full solution $\boldsymbol{x}$ itself whihc would take at least $O(N)$ time we want to compute suitable approximations to the value of properties of the solution x such as quadractic expression $\boldsymbol{x}^T M \boldsymbol{x}$ (e.g. the total weight of some subset of components).

Very large systems becoming increasingly important in applications:
- data mining/machine learning on datasets that are very large (petabytes...) to discover pattern properties within the data
- numerical solutions to PDEs. Use discretisation techniques (finite element methods). These techniques lead to systems that have the size for larger than the original problem description.

The best known classical techniques take poly(N) times to solve such problems - in general even to compute the properties of the solutions, no better method is known than computing solutions itself.

Important parameters/for both classical and quantum algorithms

- the system size N
- the desire approximation tolerance ϵ
- the condition number κ of the matrix A is defined as $\kappa = |\frac{max}{\lambda_{min}}|$ (this tells us how close A is to being non-invertible)

If we renormalise A to have $\lambda_{max} = 1$ then $_{min} = \frac{1}{\kappa}$. This is important as the numerical comptation of A^{-1} becomes less stable with increasing κ. This means we need more significant digits of computation.

15 Lecture 15

This algorithm has been improved a lot of times, including as recently as last year. So the exponential improvement in error scaling from $\frac{1}{\epsilon}$ to $\log \frac{1}{\epsilon}$ was shown in Childs, Kothan, Smomann in 2015.

We have an N-dimensional space of $n = \log N$ qubits. Let $\{|i\rangle : i = 0, ..., N-1\}$ be the computational basis. Let eigenvectors and corresponding eigenvalues of A be: $|u_j\rangle$ and λ_j for $j = 0, ..., N-1$. We being by implementing RHS $|b\rangle$ and consider it in the eigenbasis of A, $|b\rangle = \sum_{i=0}^{N-1} b_i |i\rangle = \sum_{i=0}^{N-1} \beta_j |u_j\rangle$. Then $|x\rangle = A^{-1} |b\rangle = \sum_{j=0}^{N-1} \beta_j \frac{1}{j} |u_i\rangle$. THis transformation of $|b\rangle$ is linear but not unitary so it cannot be implemented as a quantum operation. To achieve it probabilitistic we will use the phase estimation alogirhtm for $U = e^{iA}$ with the exponential (and all its powers need for PE) implemented by Hamiltonain simuluation. Then a post-selected measurement process will achieve the descired unitary transformation.

PE with hamiltonian simulation (both assumed to work perfectly) give

$$|b\rangle |0\rangle \to \sum_j \beta_j |u_j\rangle |\lambda_j\rangle$$

Next we will adjoin an extra ancilla qubit $|0\rangle$ and apply the controlled rotation (controlled by $|_j\rangle$ register)

C-rotation does

$$|\lambda_j\rangle |0\rangle \to \sqrt{1 - \frac{c^2}{\lambda_j^2}} |\lambda_j\rangle |0\rangle + \frac{c}{\lambda_j} |\lambda_j\rangle |1\rangle = |_j\rangle (\cos\theta_j |0\rangle + \sin\theta_j |1\rangle)$$

here c is chosen to have $c \leq \min_j |\lambda_j|$ ($c = \frac{1}{k}$ for definteness).

$\theta_i \in (-\frac{\pi}{2}, \frac{\pi}{2})$ and $\theta_i = \arcsin \frac{c}{\lambda_i}$ which is determined by the content of the $|\lambda_j\rangle$ register

Controlled roation is a fixed operation on $n+1$ qubits (independent of A or b) that can be implemented by a poly(n) circuit of 1 and 2 qubit gates.

Apply controlled rotation:

$$\sum_{j=0}^{N-1} \beta_j \sqrt{1-\frac{c^2}{\lambda_j^2}} \left|u_j\right\rangle \left|u_j\right\rangle \left|0\right\rangle + \beta_j \frac{c}{\lambda_j} \left|u_j\right\rangle \left|\lambda_j\right\rangle \left|1\right\rangle$$

We want the part associated with ancilla in state $|1\rangle$. This step is the post-selection step.

We measure the ancilla (hoping to get the result 1) with probability

$$p = || \sum_j \beta_j \frac{c}{\lambda_j} \left|u_j\right\rangle \left|_j\right\rangle ||^2 = \sum_j |\beta_j \frac{c}{j}|^2 = \frac{1}{k^2} \sum_j |\frac{\beta_j}{\lambda_j}|^2 \geq \frac{1}{k^2}$$

as $\sum_j |\beta_j|^2 = 1$ and $\frac{1}{\lambda_j} \geq 1$. In this case the post-measurment state is:

$$\left|\gamma\right\rangle = \frac{1}{\sqrt{p}} c \sum_{j=1}^{N-1} \frac{\beta_j}{j} \left|u_j\right\rangle \left|\lambda_j\right\rangle$$

To mitigate the probabilsitic nature of this process and obtain the state with any fixed proability with $1-\eta$ for $\eta > 0$ and η small. We can repeat the whole process $\frac{\log \frac{1}{\eta}}{p} = O(k^2)$ times and the outcome 1 will occur at least once with probability $1-\eta$.

Fact: If a signle trial has success probability p and we repeat the trial M times independantly then for any $0 < 1-\epsilon < 1$ Prob(at least on trial in M succeeds) $> 1-\epsilon$ if $M \geq \frac{-\log \epsilon}{p}$.

Remark (amplitude amplification): the $O(k^2)$ repetitions needed to generate $|\gamma\rangle$ can be improve to $O(k)$ by using AA instead of repreated measurements.

Having obtained $|\gamma\rangle$ we run the inverse phase estimation process to "uncompute" or "erase" the $|\lambda_j\rangle$ register resetting it to $|0\rangle$ and get $|\hat{x}\rangle = \frac{c}{\sqrt{p}} \sum_{j=0}^{N-1} \frac{\beta_j}{j} \left|u_j\right\rangle = \frac{c}{\sqrt{p}} \left|x\right\rangle$
$Finally, we will perform a measuremnt of observable M on |\hat{x}\rangle$ to estimate its mean value $\frac{c^2}{p} \left\langle x\right| M \left|x\right\rangle$.

According to Chernofl-Hopffling bound (look this up in lecture notes) $O(\frac{\log \frac{1}{\eta}}{\xi^2}$ measurements will suffice to estimate the meant to any desired accuracy ξ with any probability of success $1-\eta$. Similarly we can estiamte the prob p in the post-selection step to any accuarcy ξ by applying C-H bound to the ancilla

measurment outcome. As the ancilla measuremnt outcome is a random variable with two outomces 0 and 1, so finding the mean of this variable gives p. Using these values of $\frac{c^2}{p} \langle x| M |x\rangle$ and $p, c = \frac{1}{k}$ we get out our value $M = \langle x| M |x\rangle$.

16 Lecture 15

Let's stop and think about HHL a little bit more. We have considered the ideal situation but we now want to consider how it is effected by approxiamtion errors

16.0.1 Summary analysis of approxiamtion erros and runtime in HHL (non-examinable)

A full analysis of runtime and approximation errors can be found in Physical Review Letters vol 103.150502 (2005).
We assume we can make $|b\rangle$ correctly (the reason we don't want to focus on this is the issue of creating $|b\rangle$ is an orthongonal process to HHL and should be considered separately so introduces no errors). Similarly, the controlled rotation C-ROT is a final unitary (it can be efficently implemented using some universal gate set by Solovay-Kitaev theorem)
We will assume that C-rotations can be done exactly
Convsider the phase estimation part. PE for a unitary U and n qubit lines gives an estimate of λ up to n bits of precision. Denote this estimate λ' and assume that it is the closest n-bit approximation λ' is up to additive error $\eta = \frac{1}{2^n}$
The PE process uses Hamalitonian simulation that requires execution of controlled U, U^2, U^4,...,$U^{2^{n-1}}$ If $U = e^{iA}$ then $C-U = e^{i\tilde{A}}$ with $\tilde{A} = \begin{pmatrix} 0 & | & 0 \\ - & & - \\ 0 & | & A \end{pmatrix}$.
$\tilde{A}$ remains s-sparse and row-computbale if A was.

Thus we need to implement the following $e^{i\tilde{A}t}$ for $t = 1, 2, 4, ..., 2^{n-1}$ which is done using hamiltonian simualtion for s-spare matrices with circuit size $O(\log Nts^2)$. So the total circuit size $O(\log Nt_0 s^2)$ where $t_0 = 1 + 2 + ... + 2^{n-1} \sim 2^n = \frac{1}{\eta}$.

After the controlled rotation C_{rot} using the λ's produced and the corresponding measuremtn in post-selction step the final state is:

$$|\hat{x}'\rangle = \frac{1}{D'} \sum_j \frac{c}{\lambda'_j} |u_j\rangle, D' = \sqrt{p'} = \sqrt{\sum_j |\beta_j|^2 \frac{c^2}{j}}$$

$|\hat{x}\rangle$, D, p - will denote the expressions with λ's replaced by true λ values. Our requirement is that $|\hat{x}\rangle$ should be wihtin ϵ of $|\hat{x}\rangle$ so we need to choose a suitably large $t_0 = \frac{1}{\eta}$ (number of qubit lines). To establish dependenace of t_0 on we need the following facts:
a) If λ has additive error $\delta() = \eta$ then $\frac{1}{\lambda}$ has $\delta(\frac{1}{\lambda}) \sim \frac{1}{\lambda}\delta(\lambda)$ so $\frac{1}{\lambda}$ has relative

error $\frac{\delta(\frac{1}{\lambda})}{\frac{1}{\lambda}} = \frac{\eta}{}$

b) If A' and B' approximate A and B to relative error ξ then $\frac{A'}{B'}$ approximates $\frac{A}{B}$ to relative error ξ too $\frac{A'}{B'} = \frac{A(1+O(\epsilon))}{B(1+O(\epsilon))} = \frac{A}{B}(1 + O(\epsilon))$

By (a) $\frac{1}{\lambda_j'}$ approximates $\frac{1}{\lambda_j}$ to relative error $O(\frac{q}{\lambda_j}) = O(Kq)$. Similarly D' approximates D to relative error $O(Kq)$. Then by (b) $A' = \frac{\beta_j c}{\lambda_j'}$, $\beta' = D'$ the ampltitudes of $|\hat{x}'\rangle$ approximate those of $|\hat{x}\rangle$ up to multiplicative error $O(Kq)$.

$$|| \, |\hat{x}'\rangle - |\hat{x}\rangle \, || = ||(1 + O(k\eta) \, |\hat{x}\rangle - |\hat{x}\rangle \, || = O(kq) = \epsilon$$

for a given ϵ we choose $kq = \epsilon, t_0 = \frac{1}{\eta}$ we gget $t_0 = \frac{k}{\epsilon}$ and the PE step (run once) has runtime $O(\log N t_0 s^2) = O(\log N \frac{k}{q} s^2)$ for $k = poly \log N$ so runtime is $O(poly \log N) \frac{1}{\epsilon})$.

Further comments on erros and runtime:
In the above analysis we assumed that the PE process (with n qubits outputs the closest n-bit approximation λ' to λ whereas it outputs a superposition of all $\lambda_k = \lambda' \pm \frac{k}{2^n}$ (peaked around $k = 0$) which is another source of error. TO mitigate this we use PE(b), use more qubit lines to get clsoer approxiamtion with anu high probability.

In the HHL paper they use a slighly improved PE which uses non-uniform superposition state at the start. It may be shown that an optimal choice of this non-uniform superposition can lead to a quadratic reduciton in the variance of the k-value outcomes. In other works $\delta(\lambda)$ is more concentrated on smaller vlaues.

In the HHL paper, the authors present an algorithm that can handle all conditioned matrices A by introducing a further qubit "flag" and filtering functions to separate well and ill conditioned parts of A and processing the former.

In the HHL algorithm the PE step is the soruce of the worst accuracy depdnance $O(\frac{1}{\epsilon}$. This was ubsequently improved by Berry et all arXiv 1212.1414 from $O(\frac{1}{\epsilon} \rightarrow O(\log(\frac{1}{\epsilon})$ this technique invovles rpelacing PE process by an entirely different method for implementing A^{-1} . This new method is called "linear combinations of unitaries" (LCU). Implementing A^{-1} as a linear combination of unitaries each of which is individually implmented. Then we represent the linear sum in terms of another unitary V on a larger space with extra ancilla qubits.

17 Lecture 16

17.1 Clifford computations (and classical simualtion of QC)

Vague question, what is the key "quantum effect/resource/ingredient" giving quantum computing its benefits or advantage over classical computation.

17.1.1 Classical simulation of quantum computation

We are given classical description of the following:
1) description of a quantum circuit C which is just a collection of 1 or 2 qubit gates on n qubits with size $N = \text{poly(n)}$.
2) description of the input (e.g. $|l_1...l_n\rangle, l_k \in \{0,1\}$ or $|\alpha_1\rangle ... |\alpha_n\rangle$ a general product state)
3) cominated output lliines (e.g. 1st line to read off the answer)

We want to by classical means only:
Weak simulation: sample the output distribution (using classical computer with access to calssical randomness)
Strong simulation: calculate the output probabilities of p_0 and p_1 (probability of getting 0 or 1 on the first line).

We want to do it efficently in classical poly(n) time known as "efficent classical simulation".
Note, the quantum process itself gives only a weak simulation "self-simulation" as it only gives a sample of the output distribution.
"Direct" strong classical simulation is always possible but not efficent in general.

However, if we are promised that our state is a product state at every stage then we can give an efficent simulation. Each step we only involve 1 or 2 qubits in known state $|\alpha_i\rangle |\alpha_j\rangle$. Consider the 2-qubit case: U acting on 2 qubits with $|\psi\rangle = \sum_{i_1..i_n} C_{i_1...i_n} |l_1...l_n\rangle$ then in general you get $U|\psi\rangle = \sum_{i_1}^{n} V_{l_1 l_2}^{k_1 k_2} C_{k_1 k_2 l_3...l_n} |\beta_{k_1}\beta_{k_2} l_3'...l_n'\rangle$ (let $\tilde{C}_{i_1...i_n} = V_{l_1 l_2}^{k_1 k_2} C_{k_1 k_2 l_3...l_n}$. When $|\psi\rangle$ is a product state we have $C_{i_1...i_n} = a_{i_1} b_{i_2} c_{i_3} ... c_{i_n}$ so $\tilde{C}_{i_1..i_n} = (\sum U_{i_1 i_2}^{k_1 k_2} a_{k_1} b_{k_2}) c_{i_3} ... c_{i_n}$. so we can factorise back into the form $f_1...f_n$. Sometimes it was claimed that entanglement was the secret ingredient as it looks like if you take away entanglement then you can classically simulate the system. However, this argument shows that if we have no entanglement then we get no quantum advantage. This shows that entangelment is necessary but not sufficent for quantum advantage. "Most entangled states are too entnagled to be useful as quantum computatoinal resources" Fross, Flanimma, Eisert Phys Rev Letter 107 1905 01 '99. Universal quantum compuation with little entanglement Physical Rev Letter 110 060504 '13. So the story with entanglement is complicated. so we are going to consider something else called Clifford computations.

17.1.2 Clifford Compuation

Prelminary defintions: n-qubit Pauli group $\mathcal{P}_n = \{\pm 1, \pm iP_1 \otimes ... \otimes P_n\}$ with each $P_i \in \{I, X, Y, Z\}$.
Define a Clifford operation C on n qubits as such that conjugates Paulis to paulis. $\forall P \in \mathcal{P}_n CPC^\dagger \in \mathcal{P}_\backslash$. So the Clifford group is the normalizer of subgroup $\mathcal{P}_n$ in $U(2^n)$.
Clifford circuits found many applications:
1) the theory of quantum error correction ('stabilizer codes')
2) insights into quantum compuational power
3) verification of quantum computations

Examples: All Paulis are clifford
H, $S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$ phase $(\frac{\pi}{2})$ gate
CX 2 qubit gate
$Z = S^2$ and $X = HZH$
$SWAP_{12} = CX_{12}CX_{21}CX_{12}$
Full explict characterisation of Clifford gates:
Theorem: C on n qubits is Clifford iff C is a circuit of H, S, CX gates
Clifford computation circuits that only involve clifford gates.
Note: Generating entanglement $|0\rangle |0\rangle \rightarrow^H \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) |0\rangle \rightarrow^{CX_{12}} \frac{1}{\sqrt{2}}(|0\rangle |0\rangle + |1\rangle |1\rangle)$
Grottesman-Knull Theorem: Let C be any Clifford circuit on n qubits, $N = poly(n)$ with input any product state $|a_1\rangle ... |a_n\rangle$ and output a measurment on the 1st line. Then the output can always be classically strongly efficently simulated.

18 Lecture 17

Proof: The idea is that instead of evolving the input state $|\psi_{in}\rangle$ to a final state $|\psi_{final}\rangle$ for measurement we will back propagate the final Z measurement.

Let's write the measurement $Z_1 = Z \otimes B \otimes ... \otimes B$ and recall that $Z = |0\rangle \langle 0| - |1\rangle \langle 1| = \Pi_0 - \Pi_1$. So we can write the difference of the probabilities as $p_0 - p_1 = \langle \psi_{final}| Z_1 |\psi_{final}\rangle = \langle \psi_{in}| C^\dagger Z_1 C |\psi_{in}\rangle$. Let $C = C_N C_{N-1} ... C_1$ with $C_i \in \{H, S, CX\}$ so :

$$p_0 - p_1 = \langle \psi_{in}| C_1^\dagger ... C_N^\dagger Z_1 C_N ... C_1 |\psi_{in}\rangle$$

Each conjugation gives us a Pauli, and thus we get:

$$p_0 - p_1 = \langle \psi_{in}| \tilde{P}_n \otimes ... \otimes \tilde{P}_n |\psi_{in}\rangle$$

All update rules for Pauli products by H, S, CX conjugations are easy to compute (we have only 1 and 2 gate computations so it is a constant size computation)

Finally we make use of the fact that our input state is a product state. Therefore:

$$p_0 - p_1 = \prod_{i=1}^{n} \langle a_i | \tilde{P}_i | a_i \rangle$$

Each of these terms is a product of $n2x2$ matrix calculations so is efficently computable. We just need to remember that they are normalised so $p_0 + p_1 = 1$ so we get p_0, p_1 values via efficent classical computation. So Cliffords are easy.

Let us extend unitary Clifford circuits to allow the inclusion of intermediate measurements (1-qubit) measurements

Distinguish two scenarios, non-adaptive or adaptive (if we change future gates or measurements depending on the outcome of measurements). Non-adaptive: Gates cannot depend on earlier measurement outcomes. Adaptive: Gates can depend on earlier measurement outcomes.

Theorem Let C be any Clifford circuit with intermediate measurements with an arbitary product state input and a single line output. Then:
a) If C is non-adaptive then the output is classically efficiently strongly simulatable
b) If C is adaptive then full universal quantum computation is possible

Note: if input is restricted to the compuatational basis states then (b) remains classically weakly efficently simulatable (so b) relies on the arbitary product state input for its universality). R. Sazsen M van den Nest arXic 1305.6190 (2013).

Proof: for a) we observe that non-adaptive Clifford circuits are reducible to the full unitary case, and then use GK. To see how this reduction works consider a measurement between C_1 and C_2 on one of the qubit lines with C_2 independant of measurement outcome. This is equivalent to introducing an extra qubit ancilla which the measurement is swap for a CX on the ancilla, as then measuring the ancilla at any point would give the original circuit but you don't even need to measure it you can just discard it.

for b) we will use a Bravyi-Kitaev idea of so called magic states. Fact: Clifford gates with 1-qubit (non-Clifford) T gate with

$$T = \begin{pmatrix} 1 & 0 \\ 0 & e^{-\frac{\pi}{4}} \end{pmatrix} = \sqrt{s} = \text{phase } \frac{\pi}{4} \text{gates}$$

is universal. In fact if you add anything to the Clifford group that is non clifford you regain universal quantum computation. Let $|A\rangle$ denote 1-qubit state $|A\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{i\frac{\pi}{4}} |1\rangle)$.
We will implement a T gate using $|A\rangle$ ancilla and an adaptive Clifford circuit

(T-gadget).

If we state with any n-qubit state plus the ancilla $|A\rangle$. If we apply a CX from any line to the ancilla and then measure and discard the ancilla. If we get 1 in this measurement we apply an S gate to the line we applied the CX from. This generates a T gate (up to overall phase). Easy to check (ES3) this gives an adaptive Clifford circuit that always implements T gate (up to overall phase that depends on m) on line k and probabilities are $\frac{1}{2}, \frac{1}{2}$ regardless of output $|\psi\rangle$

Remark: To give increase in compuational power we could:
1) allow inclusion of new gates (T-gate)
2) add extra "magic" states as extra inputs

For Clifford circuits with product state inputs, single line outputs and intermediate measurements
a) non-adaptive -¿ classicaly strongly efficently simulatable
b) adaptive -¿ full universal quantum computing power

a) $C_0M(i_1, y_1)C_1M(i_2, y_2)C_2...$
b) $C_0M(i_1, y_1)C_1(y_1)M(i_2, y_2)C_2(y_1, y_2)$ ...

19 Lecture 17

19.1 Quantum Error Correction/Stabilizer formalism

Classical computers are incredibly reliable about 10^{-9} accuracy. This is not the case for quantum hardware, the gate fidelities are about 99.9% nowhere near classical computers. So if you ran a thousand gates in a row then as the errors are multiplicative the final state gets very perturbed, so we need to think about error correction. Our go to place is classical intituion, how is this done classically?

Consider a single classical bit x that we want to store. Let's assume that it gets flipped with probability p. One way to improve the probability of reading the correct value is instead of storing x we store xxx. Readout $y = y_1y_2y_3$. The wrong answer will be returned wih probability $3p^2(1-p) + p^3 = p^2(3-2p) \sim O(p^2)$ so for $p \in (0, \frac{1}{2})$ $p^2(3-2p) < p$. So this map $x \to x_1x_2x_3$ is an error correcting code.

Now let's talk about quantum errors. Not talking about a classical value but rather a state. We want to preserve a qubit $|\psi\rangle$. So the process of error correction can be described as:

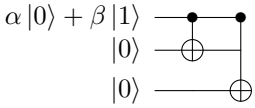
$$|\psi'\rangle = DNE\,|\psi\rangle\,|0\rangle^{\otimes n}$$

with E is the unitary encoding, N is the noise operation, and D is decoding operation. The aim is to obtain $|\psi'\rangle \approx |\psi\rangle$ for a given set of noise operations.

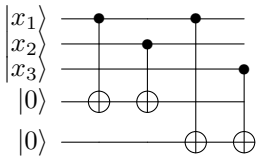
—c—c— N Syndrome
$I \otimes I \otimes I$ 00
$I \otimes I \otimes X$ 01
$I \otimes X \otimes I$ 10
$I \otimes X \otimes X$ 11
$X \otimes I \otimes I$ 11
$X \otimes I \otimes X$ 10
$X \otimes X \otimes I$ 01
$X \otimes X \otimes X$ 00

Attempt 1: Measure $|\psi\rangle$ in the computational basis to get 0 or 1. Then encode using classical repetition code. This is not a good ida as we destroy superposiiton/entanglement
Attempt 2: Engineer a map $|\psi\rangle \to |\psi\rangle\,|\psi\rangle\,|\psi\rangle$ this violates the no-cloning so isn't possible

Different approach (that works!). This is a 2-step process:
1. Encode $|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$ as $|E(\psi)\rangle = \alpha\,|000\rangle + \beta\,|111\rangle$. This is NOT the same as cloning. Encoding:



Decoding:

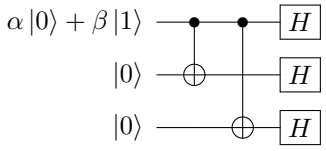


For any input superposition of the form $\alpha\,|x_1x_2x_3\rangle + \beta\,|x_1x_2x_3 \oplus 111\rangle$ the decoding circuit performs the map $(\alpha\,|x_1x_2x_3\rangle + \beta\,|x_1x_2x_3 \oplus 111\rangle)\,|0\rangle\,|0\rangle \to (\alpha\,|x_1x_2x_3\rangle + \beta(x_1x_2x_3 \oplus |11\rangle)\,|x_1 \oplus x_2\rangle\,|x_2 \oplus x_3\rangle$
If we measure two output qubits we learn $x_1 \oplus x_2$, $x_1 \oplus x_3$ without disturbing the original state. The result of measuring these output qubits is called the syndrome. If we assume only one bit has been flipped the syndrome tells us which one as they have all distinct syndromes. After we dectect a bit-flip error (on a single qubit) we apply the same bit-flip operation to that qubit and restor the original encode state $\alpha\,|000\rangle + \beta 111 \to \alpha\,|0\rangle + \beta\,|1\rangle$ by reverse the original encoding circuit.

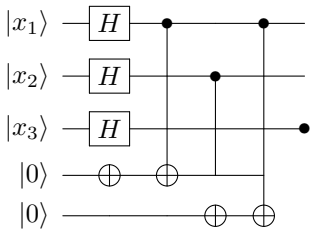
Consider the effect of a Z (or "phase") error. It maps the encoded state on a single qubit:

$$\alpha\,|000\rangle + \beta 111 \to \alpha\,|000\rangle - \beta\,|111\rangle$$

would have Syndrome measurement 00, so we cant even detect a Z error. However, they can be detected using a different code. If we notice that $Z = HXH$, so Z is effectively X but in a different basis the $|+\rangle$ $|-\rangle$ basis. We can use a different code $|\psi\rangle = \alpha|+++\rangle + \beta|---$ so what is our new encoding circuit:



Decoding:



A code of this kind can protect against Z errors. But there is a problem as this code can no longer protect against X errors.:

$$\alpha|0\rangle+\beta|1\rangle \rightarrow \frac{1}{2\sqrt{2}}\alpha(|0\rangle+\beta|1\rangle)(|0\rangle+|1\rangle))(|0\rangle+|1\rangle)+\beta(|0\rangle-|1\rangle)(|0\rangle-|1\rangle)(|0\rangle-|1\rangle) \rightarrow \frac{1}{2\sqrt{2}}\alpha(|000\rangle+|111\rangle)(|0$$

This is a 9-qubit code.

To decode, we first apply the decoding circuit for bit-flip error to each block. Assuming that at most one bit flip error has occured in each block the resulting state will be $\alpha|+++\rangle+\beta|---\rangle$ (with at most one phase flip error). We map this state to $\alpha|0\rangle + \beta|1\rangle$ using the decoding algorithm for the phase flip code.

So with a great amount of effort we managed to protect against only two types of errors. It turns out this approach works for all errors, this is because the matrices $\{I, X, Y, Z\}$ with $Y = iXZ$ form a basis for the complex vector space of $2x2$ matrices, so an arbitary error acting on a single qubit can be written as a linear combination of I, X, Z and iXZ.

Theorem: Quantum error correcting criteria