Quantum Computation

Lectures by Richard Jozsa

Contents

1 Review of Shor's Algorithm

This result is powered by the **quantum period finding algorithm**, and will lead us to the **hidden subgroup problem** (henceforce HSP).

1.1 Factoring Problem

Given an integer N, with $n = O(\log N)$ digits, we want to find a non-trivial factor in time complexity $O(\text{poly}(n))$.

The important concept here is that of **polynomial time complexity**: any computation has an input, from which we obtain an input *size* n. Then by polynomial time complexity, we mean that the number of steps/gates (either classical or quantum) grows only polynomially with n (*i.e.* is $O(\text{poly}(n))$).

When we refer to **efficient** computation, we are always referring to polynomial time complexity.

The best known *classical* factoring algorithm has complexity $e^{O\left(n^{\frac{1}{3}}(\log n)^{\frac{2}{3}}\right)}$. However, the best known quantum algorithm (due to Shor) runs in $O(n^3)$, a considerable improvement.

1.2 Quantum Factoring Algorithm Summary

First, we convert factoring into period determination:

Given N, choose $a < N$ with $(a, N) = 1$ and consider $f : \mathbb{Z} \to \mathbb{Z}_N,\ x \mapsto a^x \mod N$. Euler's Theorem tells us that f is periodic, and the period r is the order of a modulo N, *i.e.* the least $m > 1$ such that $a^m \equiv 1 \mod N$ - this exists if and only if a, N are coprime. Through knowledge of r we are able to compute a factor of N.

While the process of determining r is *mathematically* very simple, it is in fact as difficult to compute from a classical perspective as factoring N itself. Instead we use the **Quantum algorithm for periodicity determination**.

The task: Given an oracle/black box for $f : \mathbb{Z}_M \to \mathbb{Z}_N$ with promises:

- f is periodic, with (unknown) period $r \in \mathbb{Z}_M$, *i.e.* f(x+r) = f(x) for all $x \in \mathbb{Z}_M$.
- f is $1-1$ in each period, *i.e.* $f(x_1) \neq f(x_2)$ for any $0 \leq x_1 < x_2 < r$.

We want to find r in time $O(\text{poly}(m)),\ m = \log M$ (with any prescribed success probability $1 - \varepsilon,\ \varepsilon > 0$).

Remark: Queries to the oracle count as 1 step. In the quantum context we assume the oracle is a unitary gate $\mathcal{U}_f$ on $\mathcal{U}_M \otimes \mathcal{U}_N$, where $\mathcal{U}_M$ is the state space with dimension M, basis $\{|i\rangle\}_{i \in \mathbb{Z}_M}$. U_f acts on basis states as

$$U_f \underbrace{|i\rangle}_{\text{input}} \underbrace{|j\rangle}_{\text{output}} = |i\rangle|j + f(i)\rangle, \qquad i \in \mathbb{Z}_M,\ j \in \mathbb{Z}_M$$

The **Query complexity** of an algorithm is the number of times the oracle is queried, which is also required to be $O(\text{poly}(m))$.

To solve the periodicity problem classically, it can be shown that it is both necessary and sufficient to query the oracle $O(\sqrt{N})$ times, so there is no polynomial algorithm. However, there *is* a quantum algorithm.

1.3 Quantum Algorithm for Periodicity Determination

For further details *c.f.* Part II notes pp.60-64.

Write $A = M/r = \#\text{periods}$. We work in the state space $\mathcal{U}_M \otimes \mathcal{U}_N$ with basis $\{|i\rangle|k\rangle : \ i \in \mathbb{Z}_M, \ k \in \mathbb{Z}_N\}$.

Step 1: obtain the state

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

Step 2: apply U_f to obtain

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

Step 3: measure the output register, obtaining result y. By the **Born rule**, the input register collapses to all those i such that $f(i) = y$, *i.e.* $i = x_0, \ x_0 + r, \cdots, \ x_0 + (A-1)r$ where $0 \leq x_0 < r$ in the first period has $f(x_0) = y$.

We discard the output reigister to obtain

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

Note that each $0 \leq x_0 < r$ occurs with probability $1/r$.

If we naively measure $|\text{per}\rangle$, the Born rule implies we get $x_0 + jr$ with $j = 0, \cdots, A-1$ chosen uniformly with probability $1/A$, *i.e.* a random element of a random period; this is a uniformly random integer in $\mathbb{Z}_M$. This is useless to us. Instead...

Step 4: apply **Quantum Fourier Transform** (QFT). Recall that Lecture 2

$$\text{QFT}|x\rangle = \frac{1}{\sqrt{M}} \sum_{y=0}^{M-1} \omega^{xy} |y\rangle$$

Fact: QFT modulo M is unitary, and can be implemented in $O(m^2)$ time, $m = \log M$. See Part II QIC notes for circuit details of implementation.

Then

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

Note that $[\cdots]$ is a geometric series, with ratio $\omega^{ry} = \mathrm{e}^{2\pi i ry/M} = \left(\mathrm{e}^{e\pi i/A}\right)^y$. So the sum equals zero unless y is a multiple of $A = M/r$, in which case it every term in the sum is 1 so the sum equals A. So the non-multiples of A get sifted out by QFT.

Hence, we have

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

Then measuring $\mathrm{QFT}|\mathrm{per}\rangle$ we get a value $c = k_0 M/r$, with $0 \leq k_0 \leq r-1$ chosen uniformly at random. Thus we have $k_0/r = c/M$, where the values c, M are known and k_0 has been chosen at random; we want r. Note that if we are fortunate enough to have $(k_0, r) = 1$, then we can (efficiently) cancel c/M down to its lowest terms, and read off r as the denominator. But in general this will not be the case:

Theorem: (Coprimality Theorem) *The number of positive integers $< r$ that are coprime to r grows as $O(r/\log\log r)$ for large r.*

Hence the above $\mathbb{P}(k_0$ coprime to $r = O(1/\log\log r)$. So if we do it enough times, we will almost surely be successful:

Probability Lemma: If a single trial has success probability p, then we repeat k times, and for any $0 < 1-\varepsilon < 1$, we have that

$$\begin{aligned} \text{if} \quad & k = -\frac{\log \varepsilon}{p} \\ \text{then} \quad & \mathbb{P}(\geq 1 \text{ success in } k \text{ trials}) > 1 - \varepsilon \end{aligned}$$

So after finding c, cancel c/M down to its lowest terms a/b (classically, in polynomial time using Euclid's algorithm). We get r as denominator b if $(k_0, r) = 1$, which happens with probability $O(1/\log\log r)$, otherwise c, M have more common factors, so $b < r$.

We don't know immediately whether that has happened or not, but we can check the b value by making two more queries to the oracle, $f(0)$ and $f(b)$; these are equal iff $b = r$.

So if we repeat this $K = O(\log\log r)$ times, then we will obtain r with any high probability we desire - and this runs in polynomial time.

Origin and utility of QFT here

Write $R = \{0, r, 2r, \ldots, (A-1)r\} \subset \mathbb{Z}_M$, and

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

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

The problem is that the $|x_0 + kr\rangle$ terms are distributed randomly.

For each $x_0 \in \mathbb{Z}_M$, consider the map $k \mapsto k + x_0$ on $\mathbb{Z}_M$; this is the 1-1 reversible map "shift by x_0".

This gives rise to a linear map $U(x_0)$ on $\mathcal{U}_M$, and $U(x_0) : |k\rangle \to |k + x_0\rangle$ is unitary, and $|x_0 + R\rangle = U(x_0)|R\rangle$.

Since $(\mathbb{Z}_M, +)$ is an *abelian* group, these shift operators all commute, *i.e.* $U(x_0)U(x_1) = U(x_0 + x_1) = U(x_1)U(x_0)$. So they have an orthonormal basis of common eigenvectors $\{|\chi_k\rangle\}_{k \in \mathbb{Z}_M}$, called the *shift-invariant* states. Note that they are not left entirely unchanged by the $U(x_0)$ operators, but they

are shifted only by a constant phase factor, i.e. $U(x_0)|\chi_k\rangle = \omega(x_0,k)|\chi_k\rangle$ for all $x_0, k \in \mathbb{Z}_M$, and $|\omega(x_0,k)| = 1$.

Now consider $|R\rangle$ written in the χ-basis

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

where the amplitudes a_k depend only on r, and not on x_0 (obviously). Then $|\text{per}\rangle = U(x_0)|R\rangle = \sum a_k \omega(x_0,k)|\chi_k\rangle$, and measurement in the χ-basis has $\mathbb{P}(k) = |a_k\omega(x_0,k)|^2 = |a_k|^2$, independent of x_0, depending only on r. So we want to measure in this basis, but aren't allowed to do that (computationally) since the basis is too complicated.

So we introduce QFT as the unitary mapping that rotates the χ_k-basis onto the standard basis $|k\rangle$, and follow this up by a standard basis measurement.

But what does this mapping look like, and where does it come from? We need the explicit form of the shift-invariant eigenstates:

$$\begin{aligned}
|\chi_k\rangle &= \frac{1}{\sqrt{M}} \sum_{\ell=0}^{M-1} \mathrm{e}^{-2\pi i k\ell/M} |\ell\rangle \\
\implies \mathcal{U}(x_0)|\chi_k\rangle &= \frac{1}{\sqrt{M}} \sum_{\ell=0}^{M-1} \mathrm{e}^{-2\pi i k\ell/M} |\ell + x_0\rangle \\
&= \frac{1}{\sqrt{M}} \sum_{\tilde{\ell}=0}^{M-1} \mathrm{e}^{-2\pi i k(\tilde{\ell}-x_0)} |\tilde{\ell}\rangle \\
&= \mathrm{e}^{2\pi i k x_0/M} |\chi_k\rangle
\end{aligned}$$

So $\omega(x_0,k) = \mathrm{e}^{2\pi i k x_0/M}$.

The matrix of QFT^{-1} (mapping $|k\rangle$ to $|\chi_k\rangle$) has components of $|\chi_k\rangle$ as the k^{th} column, so $[\mathrm{QFT}^{-1}]_{\ell k} = \frac{1}{\sqrt{M}} \mathrm{e}^{-2\pi i \ell k/M}$. Since QFT is unitary, to find the inverse we need only take the conjugate transpose.

Hence $[QFT]_{k\ell} = \frac{1}{\sqrt{M}} \mathrm{e}^{2\pi i k\ell/M}$, as previously defined.

This notion of QFT in fact occurs very naturally in group theory as the *discrete Fourier transform*. The fact that this QFT is unitary means that all the group theoretic results it relies on slot in perfectly, allowing us to make as much use of this as we want in a way that we are not able classically. Lecture 3

The following algorithm was inspired by a desire to generalise the successful technique of Shor's celebrated algorithm.

2 The Hidden Subgroup Problem (HSP)

Let G be a finite group, of size $|G|$.

We are given an oracle $f : G \to X$ (where X is some set), and a promise that there is a subgroup $K < G$ such that

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

Problem: Determine the "hidden subgroup" K.

For instance, this might entail outputting a set of generators, or we might be happy enough with just sampling uniformly from the elements of K.

$*$ we want to solve this in time $O(\text{poly}\log|G|)$, with any constant probability $1-\varepsilon$.

Examples of problems that can be seen as HSPs:

(a) Periodicity. $f:\mathbb{Z}_M \to X$ periodic, period r, bijective within periods.

Then let $G=\mathbb{Z}_M$, $K=\{0,r,2r,\dots\}<G$, and the cosets are $x_0+K=\{x_0,x_0+r,x_0+2r\dots\}$. It is then clear that f is constant/distinct on the cosets in the desired way.

(b) Discrete Logarithms: this was also solved by Shor in his original paper.

Take p prime, and consider $\mathbb{Z}_p^*$ the group of units modulo $p=\{1,2,\dots,p-1\}$. We say that $g\in\mathbb{Z}_p^*$ is a *generator*, or a *primitive root modulo p*, if the powers of g generate all of $\mathbb{Z}_p^*$.

Fact: generators always exist, *i.e.* $\mathbb{Z}_p^*$ is always cyclic. For instance, $2,3$ both generate $\mathbb{Z}_5^*$ - though $1,4$ do not.

So any $x\in\mathbb{Z}_p^*$ can be written as $x=g^y$ for $y\in\mathbb{Z}_{p-1}$. We thus write $y=\log_g x$ for the ***discrete logarithm*** of x, to base g.

The discrete log problem is then: given a generator g and $x\in\mathbb{Z}^*p$, compute $y=\log_g x$. This is very difficult classically, and it underpins public key cryptography.

To express this as a hidden subgroup problem, consider $f:\mathbb{Z}_{p-1}\times\mathbb{Z}_{p-1}\to\mathbb{Z}_p^*$ by $f(a,b)=g^a x^{-b}=g^{a-yb} \mod p$. Then (*check*) $f(a_1,b_1)=f(a_2,b_2)$ iff $(a_2,b_2)=(a_1,b_1)+\lambda(y,1),\ \lambda\in\mathbb{Z}_{p-1}$.

So we let $G=\mathbb{Z}_{p-1}\times\mathbb{Z}_{p-1}$, and $K=\{\lambda(y,1):\lambda\in\mathbb{Z}_{p-1}\}<G$. Then f is constant/distinct as appropriate on the cosets of K, and the generator $(y,1)$ of K gives $y=\log_g x$.

(c) Graph Problems.

Consider a graph $A=(V,E)$, $|V|=n$. We stipulate that these are undirected, that there is at most one edge between any pair of vertices, and that the vertices are labelled by $[n]=\{1,2,\dots,n\}$. We also might be interested in the adjacency matrix M_A, the $n\times n$ matrix given by $[M_A]_{ij}=\mathbb{I}[\{i,j\}\in E]$, which is always symmetric for an undirected graph.

The group that will be of interest to us is $P_n \coloneqq$ the permutation group of $[n]$. So $|P_n|=n!\sim\sqrt{2\pi n}(n/e)^n$ has $|P_n|\sim O(n\log n)<O(n^2)$, which is polynomial in the number of vertices. This is what we want for the running time of a graph algorithm.

The subgroup of interest is $\text{Aut}(A)$, the ***automorphism group*** of $A<P_n$ the set of permutations $\pi\in P_n$ such that for all i,j, $\{i,j\}\in E$ iff $\{\pi(i),\pi(j)\}\in E$. What this means is that after permuting the labels of the graph, we are left with the same labelled graph.

An associated HSP (non-abelian G):

Take $G=P_n$, and $X=$ the set of all labelled graphs on n vertices (equivalent to the set of all symmetric $n\times n$ 0/1-matrices).

For any $A\in X$, we consider $f_A:G\to X$, with $f_A(\pi)=\pi(A)$, *i.e.* the graph A with its vertex labels permuted by π. A little bit of thought shows that $K=\text{Aut}(A)$ is the hidden subgroup for this problem; f_A is constant on the automorphisms of A, for instance.

An important application of this is that if we can sample uniformly from K, then we can solve the ***Graph Isomorphism Problem*** (GI), which has received a lot of attention in complexity theory in recent years.

Two labelled graphs A, B each on n vertices are ***isomorphic*** if there is a bijective map (permutation) on the labels $\pi : [n] \to [n]$ such that for all $i, j \in [n]$, $\{i, j\} \in A$ iff $\{\pi(i), \pi(j)\} \in B$. In other words, A,B are the same underlying graph (*i.e.* ignoring their labels they are indistinguishable). We write $A \cong B$.

The GI Problem is then: given graphs A and B, determine whether or not they are isomorphic. This has many useful applications; *e.g.* if you can see some proteins and their structure, you may want to be able to tell which proteins are actually the same.

This can again be expressed as a non-abelian HSP, *c.f.* Sheet 1.

There is no known polynomial time classical algorithm, and in fact there is no known polynomial time quantum algorithm either. The problem is in NP, but is *not* believed to be NP-complete. A problem is NP if it is, roughly speaking, difficult to solve but easy to validate that you have the right answer once you've solved it. A problem is NP-complete if you can rephrase any other NP problem as this one, and then solve it that way - so solving an NP-complete problem solves all NP problems; it is the hardest problem in NP.

We currently do not believe that even quantum algorithms are able to solve NP-complete problems efficiently, so it is in some sense hopeless to try and work on these problems even from a quantum perspective. However, they can do NP-incomplete problems, so factoring and GI *etc*... are good candidates to attempt.

Laslo Babai (2017) found a *quasi*-polynomial time *classical* algorithm for GI; it has runtime $n^{O((\log n)^2)}$. This is slower than polynomial time, but faster than exponential time. We have the following hierarchy:

$$\begin{aligned} \text{poly}(n) < n^{O((\log n)^2)} < \exp \\ 2^{O(\log n)} < 2^O((\log n)^3) < 2^{O(n)} \end{aligned}$$

So in terms of exponents, these are linear/polynomial/exponential in $\log n$.

(d) Another non-abelian example is the ***dihedral group***; there is a connection to the HSP 'shortest vector in a lattice'. We are given n linearly independent vectors in $\mathbb{R}^n$, and consider their lattice; the problem is to find the lattice point closest to the origin.

Lecture 4

What we *d*o have is:

Quantum Algorithm for Finite Abelian HSP

We write the group $(G, +)$, additively. We need a couple of components:

Construction of shift invariant states & Fourier transform for G

Definition : (Representation of G) A group homomorphism $\chi : G \to \mathbb{C}^* = (\mathbb{C}\backslash\{0\}, \cdot)$ is called a ***representation*** of G.

For abelian groups, any such map is called an ***irreducible representation*** (henceforth *i*rrep) of G.

These have the following properties:

Theorem: (Theorem A)

(i) any value $\chi(g)$ is a $|G|^{th}$ root of unity (so $\chi : G \to S^1$, the unit circle in $\mathbb{C}$)

(ii) (Schur's Lemma/Orthogonality) If χ_i and χ_j are representations, then

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

(iii) There are exactly $|G|$ different representations of G.

By (iii) we can label the χ's as $\chi_g : g \in G$.

Example: $\chi(g) = 1$ for all $g \in G$ is always an irrep, called the ***trivial*** irrep; label it as χ_0 for $0 \in G$.

Then for any other irrep $\chi \neq \chi_0$, orthogonality to χ_0 gives

$$\sum_{g\in G}\chi(g) = 0$$

We now introduce:

Definition : (Shift-Invariant States, Shift Operators) Consider the state sapce $\mathcal{H}_G$, dimension $|G|$, basis $|g\rangle : g \in G$.

The ***shift operators*** are

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

These all commute, so we have a simultaneous eigenbasis:

For each χ_k, $k \in G$, consider the state

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

By Theorem A (ii), these form an orthonormal basis, and it follow from the group homomorphism property of irreps that these are in fact the simultaneous eigenbasis, $U(g)|\chi_k\rangle = \chi_k(g)|\chi_k\rangle$. These are the ***shift-invariant states***.

Proof. We have that

$$\begin{aligned}
U(g)|\chi_k\rangle &= \frac{1}{\sqrt{|G|}}\sum_{h\in G}\overline{\chi_k}(h)|g + h\rangle \\
&= \frac{1}{\sqrt{|G|}}\sum_{h'\in G}\overline{\chi_k}(h' - g)|h'\rangle \\
&= \frac{1}{\sqrt{|G|}}\chi_k(g)\sum_{h'\in G}\overline{\chi_k}(h')|h'\rangle \\
&= \chi_k(g)|\chi_k\rangle
\end{aligned}$$

Which follows since $\chi_k(g^{-1}) = \overline{\chi_k}(g)$. □

Definition : (Fourier Transform QFT) The unitary gate on $\mathcal{H}_G$ mapping $|\chi_g\rangle$ basis to the $|g\rangle$ basis is known as the ***Quantum Fourier Transform (QFT)*** *i*.e.

$$\begin{aligned}\mathrm{QFT}|\chi_g\rangle &= |g\rangle,\ \forall g \in G\\ \mathrm{QFT}^{-1}|g\rangle &= |\chi_g\rangle\end{aligned}$$

In particular, the k^{th} column of QFT^{-1} in $|g\rangle$-basis is given by components of $|\chi_k\rangle$, *i*.e.

$$\left[\mathrm{QFT}^{-1}\right]_{gk} = \frac{1}{\sqrt{|G|}}\overline{\chi_k}(g)\ \ \forall g,k \in G$$

so QFT (as the conjugate transpose) has the matrix

$$\begin{aligned}[\mathrm{QFT}]_{kg} &= \frac{1}{\sqrt{|G|}}\chi_k(g)\\ \implies \mathrm{QFT}|g\rangle &= \frac{1}{\sqrt{|G|}}\sum_{k\in G}\chi_k(g)|k\rangle\end{aligned}$$

Examples: $G = \mathbb{Z}_M$ Check $\chi_a(b) = \mathrm{e}^{2\pi iab/M}$ for all $a, b \in \mathbb{Z}_M$ satisfies (Hom), so we have irreps naturally labelled by $a \in \mathbb{Z}_M$, $\chi_0(b) = 1$ for all b, giving the 'usual' QFT_M for $\mathbb{Z}_M$.

Similarly for $G = \mathbb{Z}_{M_1} \times \cdots \times \mathbb{Z}_{M_r}$, where we have

$$\left.\begin{matrix}(a_1,\ldots,a_r) = g_1\\ (b_1,\ldots,b_r) = g_2\end{matrix}\right\} \in G$$
$$\chi_{g_1}(g_2) \coloneqq \mathrm{e}^{2\pi i\left(\frac{a_1b_1}{M_1}+\cdots+\frac{a_rb_r}{M_r}\right)}$$

which satisfies (Hom), and we get

$$\mathrm{QFT}_G = \mathrm{QFT}_{M_1} \otimes \cdots \otimes \mathrm{QFT}_{M_r}$$

Importantly, the above is exhaustive due to the classification of all finite abelian groups, which states that they are all isomorphic to a direct product $G \cong \mathbb{Z}_{M_1} \otimes \cdots \otimes \mathbb{Z}_{M_r}$, and furthermore we can take all the M_i's to be (not necessarily distinct) prime powers.

Quantum Algorithm for Finite Abelian HSP

We have $f : G \to X$, hidden subgroup K, cosets $K = 0 + K, g_2 + K, \ldots, g_m + K$, where $m = |G|/|K|$ is the number of cosets.

This works on the state space $\mathcal{H}_{|G|} \otimes \mathcal{H}_{|X|}$, with basis $|g\rangle|x\rangle$ for $g \in G$, $x \in X$.

- Make the initial state

$$\frac{1}{\sqrt{|G|}}\sum_{g\in G}|g\rangle|0\rangle$$

- Apply $\mathcal{U}_f$, to obtain

$$\frac{1}{\sqrt{|G|}}\sum_{g\in G}|g\rangle|f(g_0)\rangle$$

- Measure the second register to see a value $f(g_0)$. Then the first register will give a coset state:

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

  Where we are writing $|K\rangle = \frac{1}{\sqrt{|K|}} \sum_{k \in K} |k\rangle$.

  Here the coset has been chosen uniformly at random from all $|G|/|K|$ cosets.

- Apply QFT and measure, to obtain a result $g \in G$.

  How does the output g relate to K? Observe that

  – The output distribution of g is *i*ndependent of g_0, so it is the same as that obtained from $\mathrm{QFT}|K\rangle$, (*i*.e. $g_0 = 0$).

    This is since if we write $|K\rangle$ in the shift-invariant basis basis $|\chi_g\rangle$, $|K\rangle = \sum_{g \in G} a_g |\chi_g\rangle$, then $|g_0 + K\rangle = U(g_0)|K\rangle = \sum a_g \chi_g(g_0)|\chi_g\rangle$. But $\mathrm{QFT}|\chi_g\rangle = |g\rangle$, so after QFT we have that $\mathbb{P}[g] = |a_g \chi_g(g_0)|^2 = |a_g|^2$ as $|\chi_g(g_0)| = 1$. Hence we have independence of g_0.

    But what is the distribution?

  – How does it relate to K? Lecture 5

    Recall that

$$\mathrm{QFT}|k\rangle = \frac{1}{\sqrt{|G|}} \sum_{\ell \in G} \chi_\ell(k)|\ell\rangle$$

    In particular

$$\mathrm{QFT}|K\rangle = \frac{1}{\sqrt{|G|}} \frac{1}{\sqrt{K}} \sum_{\ell \in G} \left[\sum_{k \in K} \chi_\ell(k) \right] |\ell\rangle$$

    where $[\cdots]$ invovles irreps χ_ℓ of G restricted to $K < G$, which are irreps of K. Hence

$$[\cdots] = \begin{cases} |K| & \text{if } \chi_\ell \text{ restricts to trivial irrep on K} \\ 0 & \text{otherwise} \end{cases}$$

    So the measurement gives a uniformly random choice of ℓ, such that $\chi_\ell(k) = 1$ for all $k \in K$, giving information about K.

    For instance, if K has generators $k_1, \ldots, k_M$, where $M = O(\log |K|) = O(\log |G|)$ (this is true for all finite groups), then the output has $\chi_\ell(k_i) = 1$ for all i.

    It can be shown that if $O(\log |G|)$ such ℓs are chosen uniformly at random, then with probability $> 2/3$ they suffice to determine the generating set via the equation $\chi_\ell(k) = 1$.

    See Sheet 1 # 2 for an example of this problem and calculation.

A trivial but nonetheless useful fact about irreps of G is that they restrict to irreps of K, and in fact we could have a trivial irrep of G that restricts to the trivial irrep of K - as we saw above.

Example: If $G = \mathbb{Z}_{M_1} \times \cdots \times \mathbb{Z}_{M_q}$, we had for $\ell = (\ell_1, \ldots, \ell_q), g \in (b_1, \ldots, b_q) \in G$ irreps

$$\chi_\ell(g) = \mathrm{e}^{2\pi i \sum_{j=1}^{q} \left(\frac{\ell_j b_j}{m_j} \right)}$$

So for $k = (k_1, \ldots, k_q) \in K$, the equation $\chi_\ell(k) = 1$ becomes

$$\frac{\ell_1 k_1}{m_1} + \cdots + \frac{\ell_q k_q}{m_q} \equiv 0 \mod 1$$

- that is to say, it is an integer. This is a homogeneous, linear equation in $k = (k_1, \ldots, k_q)$ and $O(\log |K|)$ independent such equations determine K as the null space of the linear system.

Remarks on HSP for *n*on-abelian groups G: Disclaimer: there is no known efficient algorithm for this problem.

As before, we can easily generate coset states

$$|g_0 K\rangle = \frac{1}{\sqrt{|K|}} \sum_{k \in K} |g_0 k\rangle$$

where the g_0 are chosen randomly. But there arise new problems with the QFT construction. In particular, there is no basis of shift invariant states because the shift operators do not commute.

So if we apply the Fourier transform to the above state, it no longer works. We can make partial progress though.

Construction of non-abelian FT

Suppose we have d-dimensional representations of G, which are group homomorphisms

$$\chi : G \to U(d)$$

where the $U(d)$ are $d \times d$ unitary matrices. So $\chi(g_1 g_2) = \chi(g_1) \cdot \chi(g_2)$, where $\cdot$ denotes matrix multiplication.

We then say χ is ***irreducible*** (so an irrep) if no subspace of $\mathbb{C}^d$ is left invariant by all matrices $\chi(g) : g \in G$. In other words, we cannot simultaneously block diagonalise all the $\chi(g)$s by a basis change.

Looking back at the abelian case, we can consider the irreps as matrices also, but since all the matrices commute with each other they can all be simultaneously block diagonalised, and reduce to 1-dimensional representations.

A ***complete set*** of irreps is a set $\chi_1, \ldots, \chi_M$ such that any irrep is unitarily equivalent to one of them, by which we mean $\chi \equiv \chi'$ if $\chi' = V \chi V^{-1}$ for some $V \in U(d)$.

Theorem: (Non-Abelian Generalisation of Theorem A) *If $d_1, \ldots, d_M$ are the dimensions of a complete set of irreps $\chi_1, \ldots, \chi_M$, then*

(i) $d_1^2 + \cdots + d_m^2 = |G|$

(ii) Write $\chi_{i,jk}(g)$ for the $(j,k)^{th}$ entry of matrix $\chi_i(g)$. Then we have (Schur's orthogonality)

$$\sum_{g \in G} \chi_{i,jk}(g) \overline{\chi}_{i',j'k'}(g) = |G| \delta_{ii'} \delta_{jj'} \delta k k'$$

Hence the states

$$|\chi_{i,jk}\rangle \equiv \frac{1}{\sqrt{|G|}} \sum_{g \in G} \overline{\chi}_{i,jk}(g) |g\rangle$$

form an orthonormal basis.

QFT on G is then defined as the unitary map that rotates the $\{|\chi_{i,jk}\rangle\}$ basis into the standard basis $\{|g\rangle\}$.

But the $|\chi_{i,jk}\rangle$ are ***not*** shift invariant for all $U(g_0)$s, so measurement of the coset state $|g_0 K\rangle$ in the $|\chi\rangle$ basis has output distribution that is ***not*** independent of g_0.

However, a 'partial' shift invariance survives. Instead of performing a complete measurement, we can do an incomplete measurement on only the i labels and not the j.

We call this measurement M_{rep} on $|g_0K\rangle$ that distinguishes only irrep labels (i values) and ***not*** all (i,j,k)s, *i*.e. the measurement outcome i is associated to the d_i^2-dimensional subspace spanned by $\{|\chi_{i,jk}\rangle\}_{j,k=1,\ldots,d_i}$.

Then it can be shown that $\chi_i(g_1g_2) = \chi_i(g_1)\chi_i(g_2)$ implies that the output distribution of the i-values is independent of g_0. This gives direct, but incomplete information about K.

For instance, conjugate subgroups K and $L = g_0Kg_0^{-1}$ for some $g_0 \in G$ give the same output distribution of is.

For efficient HSP algorithm (if we use QFT), we need QFT to be efficiently implementable, *i*.e. in poly$(\log|G|)$ times - this is unusual, and a very special thing to ask for.

This is true for abelian groups, and some non-abelian groups, such as P_n - but even for this group there is still no known efficient hidden subgroup algorithm.

Consider $\mathbb{Z}_{2^n}$; we have a tower of subgroups $\{0\} < \mathbb{Z}_2 < \mathbb{Z}_4 < \cdots < \mathbb{Z}_{2^n}$, and the fast Fourier transform on this group works recursively by determining the transform on the next subgroup using the previous one. There is a similar construction $\{e\} < P_2 < \cdots < P_n$. However, there is still no efficient HSP algorithm known.

Known partial results:

For normal subgroups (*i*.e. $gK = Kg$ for all $g \in G$), we have

Theorem: (Hallgren, Russell, Tashina SIAMJ. Comp_ vol32 p916-934 (2003))

Suppose G has QFT efficiently implementable. Then if the hidden subgroup K is a normal subgroup, then there is an efficient quantum HSP algorithm.

In particular, we find $\{|g_0K\rangle\}$ and do M_{rep} on it, and repeat $O(\log|G|)$ times. Then K normal implies the outputs suffice to efficiently determine K.

Theorem: (Effinger, Heyer, Krull, 2004) *For general non-abelian HSP, $M = O(poly\log|G|)$ random coset states $|g_1K\rangle, \ldots, |g_MK\rangle$ suffice to determine K.*

In particular, there is always efficient $q$uery complexity. Unfortunately, there is no known method to efficiently determine K from the M coset states (see Sheet 1 # 7 for a proof of this theorem).

The way we do this is to use a measurement procedure on $|g_1K\rangle \otimes \cdots \otimes |g_MK\rangle$ that takes exponential time in $\log|G|$ to complete.

Lecture 6

3 Quantum Phase Estimation (PE) Algorithm

This is a unifying principle for quantum algorithms, and again uses QFT. It has many applications, *e*.g. an alternative factoring algorithm to Shor's algorithm, due to A. Kitaev (Sheet 2 #2).

Scenario: we are given a unitary operator in d dimensions, and an eigenstate $|v_\varphi\rangle : U|v_\varphi\rangle = \mathrm{e}^{2\pi i\varphi}|v_\varphi\rangle$. We want to estimate φ, with $0 < \varphi < 1$, to n binary bits of precision. In particular we have $\varphi = 0.i_1i_2\ldots i_n\ldots$ in binary, and we want to determine $i_1, \ldots, i_n$ for any given n.

We will need the ***controlled-U^k*** for integers k:

$$CU^k|0\rangle|\xi\rangle = |0\rangle|\xi\rangle$$
$$CU^k|1\rangle|\xi\rangle = |1\rangle(u|\xi\rangle)$$

Note also that $U^k|v_\varphi\rangle = \mathrm{e}^{2\pi i k\varphi}|v_\varphi\rangle$, and $C(U^k) = (CU)^k$.

Remark: Given U as a formula or circuit description, we can readily implement CU *e*.g. by controlling each gate of the circuit.

But if U is given as a black box (*i*.e. physical operation) then we need further information in order to implement CU.

We can see this since U as a black box is equivalent to $\mathrm{e}^{i\theta}U$ as a black box, but the controlled versions of these gates are different, since the C-($\mathrm{e}^{i\theta}U$) gate will switch global phase on $|0\rangle + |1\rangle$ to a local phase.

It in fact suffices to have an eigenstate $|\alpha\rangle$ with known/specified eigenvalue $\mathrm{e}^{i\alpha} : U|\alpha\rangle = \mathrm{e}^{i\alpha}|\alpha\rangle$. Then $\mathrm{e}^{i\theta}U$ has $\alpha \mapsto \alpha + \theta$.

We will actually want a "generalised controlled-U", given for $x \in \mathbb{Z}_n$:

$$|x\rangle|\chi\rangle \mapsto U^x|\chi\rangle$$

We can make it from cU^k as follows. For $x = x_{n-1}\ldots x_1x_0$ in binary... [diagram].

If input $|\chi\rangle = |v_\varphi\rangle$, then we get $\mathrm{e}^{2\pi i\varphi x}|x\rangle|v_\varphi\rangle$.

Theorem: (PE) *If the measurements give output $\theta = 0.y_0y_1\ldots y_{n-1}$, with $\varphi = 0.z_0, \ldots, z_{n-1}z_n\ldots$ then*

(a) $\mathbb{P}(\theta$ is closest n binary digit approx to $\varphi) \geq 4/\pi^2 \approx 0.4$

(b) $\mathbb{P}(|\theta - \varphi| \geq \varepsilon)$ is at most $O(1/2^n\varepsilon)$, and we will show it is in fact $\leq 1/2^{n+1}\varepsilon$.

Remark: In Theorem PE (a), we have probability $4/\pi^2$ that *a*ll n lines of n-line PE process give correct bits.

But: if we want φ accurate to m bits with probability $1-\eta$, then use Theorem PE (b) with $\varepsilon = 1/2^m$. Then we'll need $n > m$ lines with $\eta = 1/2^{n+1}\varepsilon$ and $\varepsilon = 1/2^m$, *i*.e. $n = m + \log(1/\eta) - 1$, where n is the number of lines in the PE algorithm, m is the number of bits we want with high probability, and $\log(1/\eta) - 1$ is the number of additional bits we require to ensure this accuracy.