1 Shor's Algorithm

1.1 Period Finding

We define a classical oracle as some black box that we can query with $x \in \mathbb{Z}_M$ that returns $f(x)$. Classically, we can do this in $\mathcal{O}(\sqrt{M})$ to find r. We similarly define a quantum oracle as a black box that takes some $x \in \mathbb{Z}_M$ in an input register $|x\rangle$ and outputs $|f(x)\rangle$ in a way that is not necessarily reversible or unitary. We can commonly encapsulate this oracle into some unitary U_f such that:

$$|x\rangle |z\rangle \xrightarrow{U_f} |x\rangle |z + f(x)\rangle \tag{1}$$

where the + is carried out in mod N such that our input register is in $\mathbb{Z}_M$ as before. Note that our output register is in $\mathbb{Z}_N$. We note that our quantum query complexity becomes the number of times our quantum algorithm uses U_f.

Problem 1.1 (Period Finding)
Input: Given a function, $f : \mathbb{Z}_M \to \mathbb{Z}_N$ with the promise that f is periodic with period $r < M$ such that $M \bmod r \equiv 0$:

- $\forall x \in \mathbb{Z}_M, f(x + r) = f(x)$ where addition is carried out over mod M
- f is one-to-one in each period: $\forall 0 \leq x_1 < x_2 \leq r, f(x_1) \neq f(x_2)$
- We can find and are given a quantum oracle as a unitary $U_f : \mathcal{H}_M \otimes \mathcal{H}_N \to \mathcal{H}_M \otimes \mathcal{H}_N$ that encapsulates the action of $f(x)$.

Task: Find r in time $\mathcal{O}(\text{poly}(m))$ with an input size of $m = \mathcal{O}(logM)$.

To solve our period finding problem, we must invoke the quantum Fourier transform, a fundamental technique in almost every quantum algorithm with a speed-up over classical ones. The quantum transform closely follows the classical version. When formulated using qubits, we choose it to follow the discrete classical Fourier transform except with a basis of qubits going from the computational basis to a dual basis as shown in the following theorem:

Theorem 1.2 (Quantum Fourier Transform (QFT))
The quantum Fourier transform is a unitary transformation between dual variables such that

it acts as a unitary and we assume that it has an efficient implementation with $\mathcal{O}(m^2)$ gates.

$$|x\rangle \to \frac{1}{\sqrt{M}} \sum_{y=0}^{M-1} \omega^{xy} |y\rangle \tag{2}$$

where $\omega = e^{2\pi i/M}$.

We can employ our quantum Fourier transform to devise an algorithm for period finding:

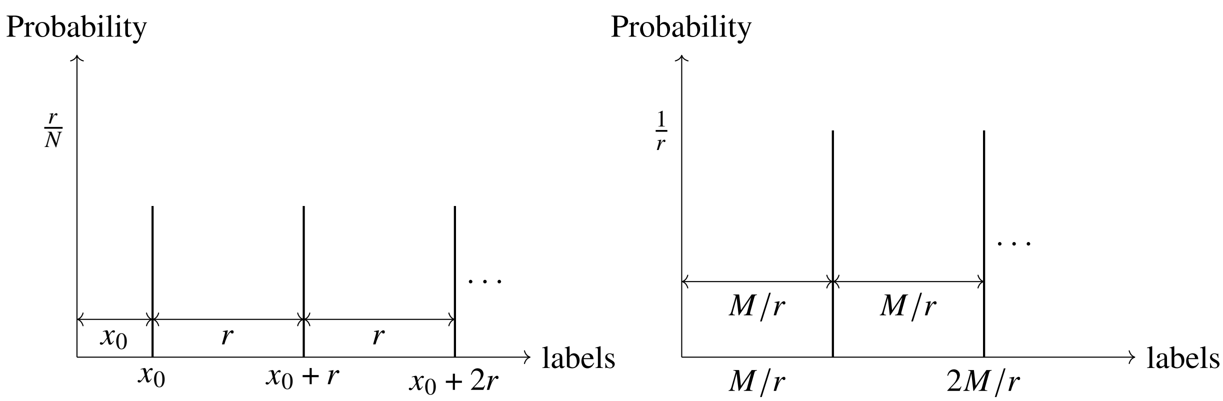
Algorithm 1.3 (Period Finding)

1. Make the superposition state $\frac{1}{\sqrt{M}} \sum_{i=0}^{M-1} |i\rangle |0\rangle$
2. Query U_f to get $\frac{1}{\sqrt{M}} \sum_{i=0}^{M-1} |i\rangle |f(i)\rangle$
3. Measure the output register and if the measurement outcome is $y \in \mathbb{Z}_N$, then by the Born rule, our state collapses such that the input register is a superposition of i such that $f(i) = y$, meaning that $i \in x_0, x_0 + r, x_0 + 2r \ldots, x_0 + (A-1)r$ where $A = M/r$ each with equal probability and some random shift x_0 so:

   $$|per\rangle = \frac{1}{-}$$

   $\sum_{j=0}^{A-1} |x_o + jr\rangle$ (3)
4. Apply the quantum Fourier transform to $|per\rangle$
5. Measure the final outcome after applying QFT

We can visualize our state outcomes of step 3 and step 4 as the following combs, (a) and (b) respectively:



(a) for $|per\rangle$ (b) for QFT $|per\rangle$

Walking through step 4 of the algorithm we are first given some $|per\rangle = \frac{1}{A}\sum_{j=0}^{A-1}|x_o + jr\rangle$ from step 3, where $A = M/r$. Applying QFT with $\omega = e^{2\pi i/A}$, we get:

$$QFT\,|per\rangle = \frac{1}{\sqrt{M}}\sum_{y=0}^{M-1}\frac{1}{\sqrt{A}}\sum_{j=0}^{A-1}\omega^{(x_0+jr)y}\,|y\rangle = \frac{1}{\sqrt{MA}}\sum_{y=0}^{M-1}\omega^{x_0 y}\left(\sum_{j=0}^{A-1}\omega^{jry}\right)|y\rangle \tag{4}$$

Note 1.4
The sum of a geometric series with ratio α starting at 1 is $\frac{1-\alpha^N}{1-\alpha}$ if $\alpha \neq 1$ and N if $\alpha = 1$.

If we let $\alpha = \omega^{ry} = e^{2\pi iry/A}$ the only nonzero terms will be if $\omega^{ry} = 1$ since if $\omega \neq 1$, $\frac{1-\alpha^A}{1-\alpha} = 0$ and we can rewrite our result as:

$$QFT\,|per\rangle = \sqrt{\frac{A}{M}}\sum_{k=0}^{r-1}\omega^{x_0 kM/r}\,|kM/r\rangle \tag{5}$$

where our dual variable y can be expressed as: $y \to kM/r$ for integer $k \leq M$.

Now, since the measurement outcomes and probability are independent of x_0, our results carry useful information about r! Going through our 5th step, we measure QFT$|per\rangle$ to get an outcome of $c = \frac{k_0 M}{r}$ for some $0 \leq k_0 \leq r-1$. Note that $\frac{c}{M} = \frac{k_0}{r}$, so if k_0 was co-prime to r, we can cancel c/M down to its lowest terms and read off r.

Theorem 1.5 (Co-primality)
The number of integers less than r that are coprime to r scales as $\mathcal{O}(\frac{r}{\log\log r})$ for large r.

Thus, the probability that step 5 gives us some k_0 that is coprime to r is $\mathcal{O}(\frac{r}{\log\log r}$. To check the computed value of $\tilde{r}$, we can query U_f to see if $f(0) = f(\tilde{r})$, verifying the periodicity. Finally, we can continue to repeat the algorithm multiple times to boost our success probability ($\mathcal{O}(log\log M) = \mathcal{O}(\log m)$ is shown to be optimal in the next section).

1.2 Shor's Factoring Algorithm

Given a classic factoring problem, we can frame it as such:

Problem 1.6 (Factoring)
Input: A positive integer N representing the number we would like to factorize.
Task: Find a nontrivial factor of N in polynomial time $\mathcal{O}(\text{poly}n)$ in $n = \log(N)$.

1. We note that the length of the input/memory space in bits required to store the input is given by $n = \log(N)$.
2. Classically, the best-known algorithm runs in $\exp\left\{\mathcal{O}(n^{\frac{1}{3}}(\log n)^{\frac{2}{3}})\right\}$ time.
3. The quantum algorithm, using Shor's, runs in $\mathcal{O}(n^3)$.

We can convert our factoring problem conveniently into a period-determination problem as follows!

If we are given some N, we can choose some a that is co-prime to N such that $a < N$. Following Euler's theorem, we define a function:

$$f : \mathbb{Z} \to \mathbb{Z}_N, f(x) = a^x \bmod N \tag{6}$$

From Euler's theorem, we know that $f(x)$ is periodic since $\exists x$ such that $1 \equiv a^x \mod N$ and $f(x_1 + x_2) = f(x_1)f(x_2)$ by nature of the exponential function. Thus, the period, r (the smallest such x), is of order N. As we have an efficient quantum algorithm (period finding) and can find r, we note that:

$$a^r - 1 = (a^{r/2} - 1)(a^{r/2} + 1) \equiv 0 \bmod N$$

So, N can always be factored into the product $(a^{r/2}-1)(a^{r/2}+1)$ and given some r, we can calculate each of these terms in $\mathcal{O}(\text{poly}(n))$ time.

Note: This is only true given that r is even and $a^{r/2} - 1 \not\equiv -1 \bmod N$ so we must determine how likely this is [1]. To do so, we invoke the following theorem whose proof is given in Nielsen and Chuang [1, 2]:

Theorem 1.7
Suppose $N \in \mathbb{Z}_+$ is odd and not a power of a prime. If $a < N$ is chosen uniformly at random with $\gcd(a,N) = 1$, then $Prob\left(r \text{ is even and } a^{r/2} \equiv -1 \pmod N\right) \geq \frac{1}{2}$.

Noting this, for any N not odd nor a prime power, we will obtain a valid factor with a probability of at least $\frac{1}{2}$ and we can easily check it in $\mathcal{O}(\text{poly}(n))$ time by simply dividing it into N. Thus, if we repeat the process, we will be able to successfully factorize N with high probability. Formalizing the bound, we can state the following theorem:

Theorem 1.8
$Prob$ (at least 1 success in M trials) $\geq 1 - \epsilon$ if $M = \frac{-\log \epsilon}{p}$, where p is the success probability for 1 trial.

Proof (Theorem 1.4)
The probability of at least 1 success in M trials is given by $1 - (1-p)^M$. Note that if we require $(1-p)^M < \epsilon$, then taking log of both sides, we get:

$$M \log(1-p) < \log \epsilon$$

Since $0 \le p \le 1$, note that $-p \le \log(1-p)$, and thus we can rewrite our inequality as:

$$-Mp \le \log \epsilon$$

which gives us our required bound of $M = \frac{-\log \epsilon}{p}$. □

Thus, using our knowledge, we can now formulate Shor's algorithm as a period-finding problem and by finding r, repeating multiple times ($M = \frac{-\log \epsilon}{p}$) for $1 - \epsilon$ success probability.

2 Hidden Subgroup Algorithm

2.1 Why does QFT help? (Shift Operators)

Given $R = \{0, r, 2, r, \ldots, (A-1)r\} \in \mathbb{Z}_M$, consider:

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

Looking at our periodic state after applying U_f:

$$|per\rangle = |x_0 + R\rangle = \frac{1}{\sqrt{A}} \sum_{k=0}^{A-1} |x_0 + kr\rangle \tag{7}$$

The problem for us previously was that x_0 was some random shift so thus we needed the QFT to exploit the periodic structure. For each $x_i \in \mathbb{Z}_M$ we can define the shift operator mapping $x_i \to x_i + k$ with an associated linear map shift operator $T(x_0) : \mathcal{H}_M \to \mathcal{H}_M$. Because $(\mathbb{Z}_M, +)$ is abelian, the $U(x_i)$ commute with each other and thus have a shift-invariant basis set:

Definition 2.1 (Shift-invariant States)
A simultaneous basis of eigenvectors, $\{|\chi_k\rangle\}, k \in \mathbb{Z}_M$, such that for any $k, x_i \in \mathbb{Z}_M$ and given some linear map shift operator $T(x_i)$ such that $T(x_i)|\chi_k\rangle = \omega(x_i, k)|\chi\rangle_k$.

- We restrict $|\omega(x_i, k)| = 1$
- This forms an orthonormal basis for $\mathcal{H}_M$.

- $\omega(x_i,k)$ can be thought of the character of Z_M.

Now, we can write R in the shift-invariant basis:

$$|R\rangle = \sum_{k=0}^{M-1} a_k |\chi_k\rangle \tag{8}$$

$$|per\rangle = T(x_0)|R\rangle = \sum_{k=0}^{M-1} a_k \omega(x_0,k) |\chi_k\rangle \tag{9}$$

where the a_k's only depend on r. A measurement in the $|\chi_k\rangle$ basis gives an outcome k with $Prob(k) = |a_k \omega(x_0,k)|^2 = |a_k|^2$.

Suppose that the unitary U maps from the shift-invariant basis to the computational basis (not to be confused with the linear map shift operator):

$$|\chi_k\rangle \xrightarrow{U} |k\rangle \tag{10}$$

We can thus let:

$$|\chi_k\rangle = \frac{1}{M} \sum_{l=0}^{M-1} e^{-2\pi ikl/M} |l\rangle \tag{11}$$

$$\implies T(x_0)|\chi_k\rangle = \frac{1}{\sqrt{M}} \sum_{l=0}^{M-1} e^{-2\pi ikl/M|x_0+l\rangle} = \frac{1}{\sqrt{M}} \sum_{l=0}^{M-1} e^{-2\pi ik(\tilde{l}-x_0)/M|\tilde{l}\rangle} \tag{12}$$

$$= e^{2\pi ikx_0/M} |\chi_k\rangle \tag{13}$$

where $e^{2\pi ikx_0/M}$ is simply our character $\omega(x_0,k) = \omega^{kx_0}$.
Looking at $U^{-1} : |k\rangle \to |\chi_k\rangle$, we note that $[U^{-1}]_{jk} = \langle j| U^{-1} |k\rangle = \frac{1}{\sqrt{M}} e^{-2\pi jk/M}$, so thus:

$$[U]_{jk} = \frac{1}{\sqrt{M}} e^{2\pi ijk/M} \tag{14}$$

$$U|k\rangle = \frac{1}{\sqrt{M}} \sum_{j=0}^{M-1} e^{2\pi ijk/M} |j\rangle \tag{15}$$

This is just our quantum Fourier transform! Thus, we can see how going from our shift-invariant basis to our computational basis and vice versa is done using QFT.

2.2 Hidden Subgroup Problem

Problem 2.2

Input: Some finite group G of size $|G|$ and an oracle for a function $f : G \to X$.

Promise: There is a subgroup $K < G$ such that:

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

Task: Determine the hidden subgroup K in time/queries $\mathcal{O}(poly(\log |G|))$ with high probability $1 - \epsilon$.

Definition 2.3 (Coset)

The set of cosets is given by $gK = \{gk|k \in K\}$ for all $g \in G$.

2.2.1 Period Finding as an HSP

$f : \mathbb{Z}_M \to \mathbb{Z}_N$ with periodic r and 1-1 in each period. $G = \mathbb{Z}_M, K =< r >= \{0,r,2r,\dots,(A-1)r\}$. Cosets of K: $x_0 + K = \{x_0, x_0 + r, x_0 + 2r, \dots, x_0 + (A-1)r\}\,; 0 \le x_0 \le r$

2.2.2 Discrete Logarithm as an HSP

We are given prime p and the group $G = \mathbb{Z}_p^* = \{1,2,\dots,p-1\}$ under multiplication mod p. $g \in \mathbb{Z}_p^*$ is called a generator (or a primitive root) mod p if powers of g generate $\mathbb{Z}_p^* = \left\{g^0, g, g^2, \dots, g^{p-2}\right\}$ and $g^{p-1} \equiv 1 \bmod p$. The discrete log problem is to find $y = \log_g x$.

References

[1] Richard Jozsa. Quantum information and computation part ii lecture notes. Lecture Notes, University of Cambridge, 2024. Available at `https://www.qi.damtp.cam.ac.uk/files/PartIIIQC/Part%202%20QIC%20lecturenotes.pdf`.

[2] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010.