CMPE 598 - Lecture Notes

Mert Kalaylıoğlu

April 24, 2018

1. Shor's Algorithm for Factorization

Given a positive integer, find its factors.

There exists a fast classical algorithm for detecting whether the number is prime. If so, problem solved.

There exists a fast classical algorithm for detecting whether the number is a power (i.e. of the form $a^b \ for \ b > 1$).

$x = a^b$

$logx = b * loga$ so b cannot be bigger than logx.

Binary Search: If you can find a factor, you can find all other factors as well using the same method repeatedly.

Assume you are given a number pq where p and q are primes.

We want to factor a large integer N.

- FACTORING is reduced to finding **a non-trivial square root** of 1 modulo N.

$y^2 \equiv 1 \ (modN) \ (y \in 1, 2, ..., N-1)$

if N = 15

trivial square roots of 1 (mod 15)

$1^2 \equiv 1 \ (mod15)$ not exciting

$(-1)^2 \equiv 1 \ (mod15)$ not exciting

$14^2 \equiv 1 \ (mod15)$ not exciting

$-1 \equiv 14 \ (mod15)$

non-trivial square root of 1 (mod 15)

$4^2 \equiv 1 \ (mod15)$

If y is a non-trivial square root of 1 mod N,

Then N divides $y^2 - 1 = (y+1)(y-1)$, but N does **not** divide neither $y-1$

nor $y+1$. So this means that $gcd(y-1, N) > 1$ because if $y-1$ and N were relatively prime, then since N divides $(y-1)(y+1)$, it would have to divide $(y+1)$.

- Finding such a root is reduced to computing **the order of a random integer modulo N**.

  Pick a random number X (mod N).

  The order of X (mod N) is the smallest number r such that $X^r \equiv 1\ (modN)$.

  Example: The order of 2 mod 15 is 4. $2^1 = 2, 2^2 = 4, 2^3 = 8, 2^4 = 1, 2^5 = 2, 2^6 = 4, 2^7 = 8, 2^8 = 1, ...$
- The order of an integer is precisely the period of a particular periodic superposition.

  "Period = Order"
- And, periods of superpositions can be found by the quantum FFT.

Classical FFT's input is an M-dimensional, complex valued vector α (where M is a power of 2, say 2^m) and its output is an M-dimensional, complex valued vector β;

$$\begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_{M-1} \end{bmatrix} = \frac{1}{\sqrt{M}} \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & W & W^2 & \dots & W^{M-1} \\ 1 & W^2 & W^4 & \dots & W^{2(M-1)} \\ 1 & W^3 & W^6 & \dots & W^{3(M-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W^{(M-1)} & W^{2(M-1)} & \dots & W^{(M-1)(M-1)} \end{bmatrix} \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_{M-1} \end{bmatrix} \tag{1}$$

where W is a complex M^{th} root of unity.

FFT runs in $O(M * logM)$ steps.

Input: A superposition of m = log M qubits $|\alpha\!>= \sum_{j=0}^{M-1} \alpha_j\ |j\!>$.

$$\begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_{M-1} \end{bmatrix} = \alpha_0 \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + \alpha_1 \begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + \alpha_2 \begin{bmatrix} 0 \\ 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix} + ... + \alpha_{M-1} \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \tag{2}$$

Method: We'll use $O(m^2)$ quantum operations to obtain the superposition. $|\beta\!>= \sum_{j=0}^{M-1} \beta_j\ |j\!>$

Output: A random m-bit number j (i.e. $0 \leq j \leq M-1$), from the probability distribution $Pr[j] = |\beta_j|^2$.

Suppose the input to quantum Fourier sampling is periodic with period k, for some k that divides M. Then the output will be a multiple of $\frac{M}{k}$, and it is equally likely to be any of the k multiples of $\frac{M}{k}$.

$$|\alpha\!> = \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_{M-1} \end{pmatrix} \tag{3}$$

$|\alpha\!>$ is such that $\alpha_j = \alpha_i$ whenever i = j mod k where k is a particular integer that divides M. So there are $\frac{M}{k}$ repetitions of same sequence $(\alpha_0, \alpha_1, ..., \alpha_{k-1})$ of length k.

And suppose exactly one of these k numbers is non-zero, say α_j.

Suppose the vector $|\alpha\!>= (\alpha_0, \alpha_1, ..., \alpha_{M-1})^T$ is periodic with period k with no offset (that is, the non-zero terms are $\alpha_0, \alpha_k, \alpha_{2k}, ...$). Thus $|\alpha\!>= \sum_{j=0}^{\frac{M}{k}-1} \sqrt{\frac{k}{M}}\ |jk\!>$.

Claim:

$$|\beta\!>= \frac{1}{\sqrt{k}} \sum_{j=0}^{k-1} |\frac{jM}{k}\!> \tag{4}$$

In the input vector, the coefficient of α_l is $\sqrt{\frac{k}{M}}$ if k divides l, and zero otherwise. The j^{th} coefficient $|\beta>$ is

$$\beta_j = \frac{1}{\sqrt{M}} \sum_{l=0}^{M-1} w^{jl} \alpha_l = \frac{\sqrt{k}}{M} \sum_{i=0}^{\frac{M}{k}-1} w^{jik} \tag{5}$$

So this sum is the geometric series $1+w^{jk}+w^{2jk}+...$ containing $\frac{M}{k}$ terms and with ratio w^{jk}. There are two cases. If the ratio is exactly 1, which happens if $jk \equiv 0 \; mod \; M$, then the sum of the series is just the number of terms. If the ratio isn't 1, apply the usual formula for geometric series to find that the sum is

$$\frac{1 - w^{jk(\frac{M}{k})}}{1 - w^{jk}} = \frac{1 - w^{jM}}{1 - w^{jk}} = 0 \tag{6}$$

So $\beta_j = \frac{1}{\sqrt{k}}$ is M divides jk, and is zero otherwise. Also works (with little modification) for the case where the offset is non-zero.

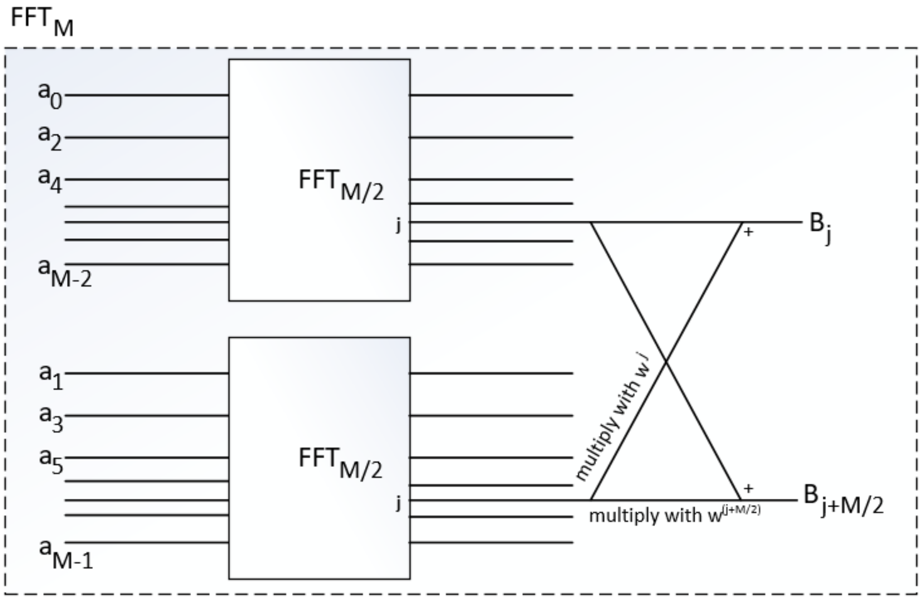
Suppose s independent samples are drawn uniformly from $\{0, \frac{M}{k}, \frac{2M}{k}, ..., \frac{(k-1)M}{k}\}$. Then, with probability at least $1 - \frac{k}{2^s}$, the greatest common divisor of these samples is $\frac{M}{k}$.

Proof: The only way this can fail is if all samples are multiples of $j\frac{M}{k}$, for some $j > 1$. So, fix any integer $j \geq 2$. The chance that a particular sample is a multiple of $j\frac{M}{k}$ is at most $\frac{1}{j} \leq \frac{1}{2}$, so the chance that **all samples** are multiples of $j\frac{M}{k}$ is at most $\frac{1}{2^s}$. The probability that this bad thing will happen for some $j \leq k$ is at most $k\frac{1}{2^s}$, since these are k candidates for the number j.

How does the classical FFT work?

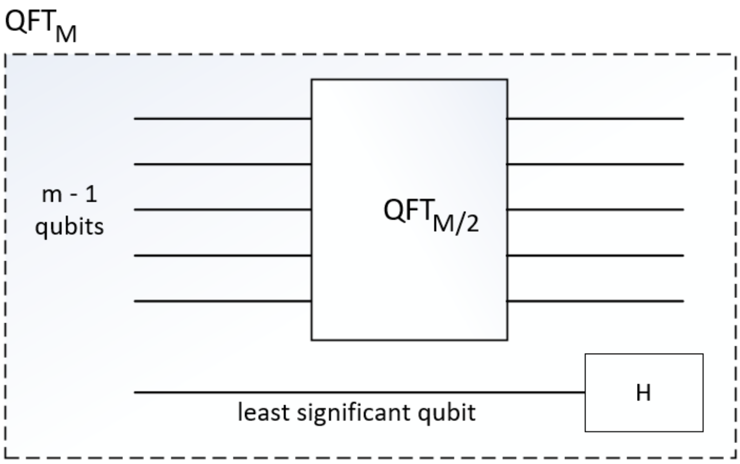
"Divide & Conquer"

from input $(\alpha_0, \alpha_1, ..., \alpha_{M-1})^T$ to output $(\beta_0, \beta_1, ..., \beta_{M-1})^T$



$w^{\frac{M}{2}} = -1$.

In the quantum version, the input is now encoded in the 2^m amplitudes of $m = logM$ qubits. So the decomposition of the inputs to evens and odds is determined by the least significant qubit. We will design a quantum circuit (subroutine) QFT_M. $QFT_{\frac{M}{2}}$ will be applied to the remaining m-1 qubits.



$$H = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \tag{7}$$

$$Other\ lines = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{8}$$

$$A = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots \\ 0 & 0 & 0 & \dots & 1 \end{pmatrix} \otimes \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & 0 & \dots & 0 & 0 \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \dots & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & \dots & 0 & 0 \\ \vdots \\ 0 & 0 & 0 & 0 & \dots & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 0 & 0 & 0 & 0 & \dots & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \tag{9}$$

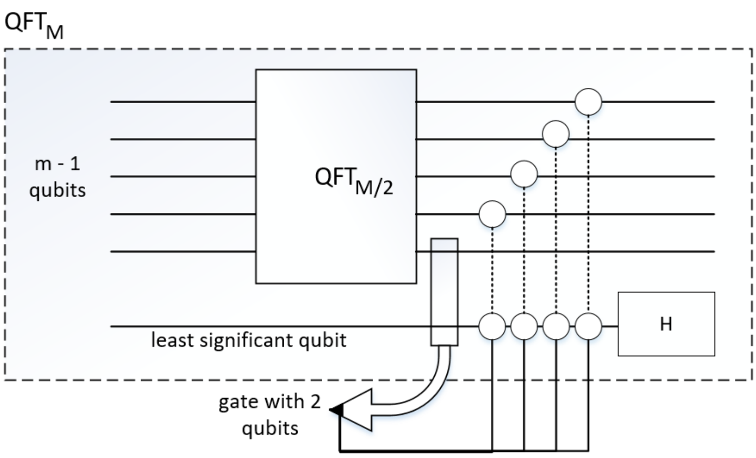
01100 0 $\rightarrow$ even

01100 1 $\rightarrow$ odd

$$\begin{pmatrix} \alpha_{y_0} : \text{even} \\ \alpha_{y_1} : \text{odd} \end{pmatrix} \tag{10}$$

$$A \begin{pmatrix} \alpha_{y_0} \\ \alpha_{y_1} \end{pmatrix} = \begin{pmatrix} \frac{\alpha_{y_0}+\alpha_{y_1}}{\sqrt{2}} \\ \frac{\alpha_{y_0}-\alpha_{y_1}}{\sqrt{2}} \end{pmatrix} \tag{11}$$

For each j, an operation is done in the classical FFT on the $(\frac{M}{2}+j)^{th}$ wire. If j is represented by the m-1 bits $j_1, j_2, ..., j_{m-1}$, then $w_j = \prod_{l=1}^{m-1} w^{2^{j_l}}$. Ex: m = 3, m - 1 = 2, $j_2 j_1$.



$$Gate\ with\ 2\ qubits = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & W^{2^j} \end{bmatrix} \tag{12}$$