Deutsch's algorithm

A function $f: \mathbb{B}^n \to \mathbb{B}$ is called

1) constant if $f(s_1 \cdots s_n) = b \in \mathbb{B}$

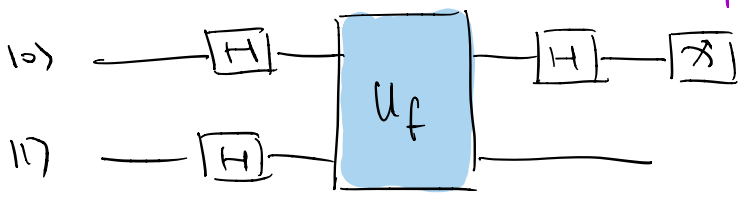
$\forall\ s_1 \cdots s_n \in \mathbb{B}^n$

2) balanced if $|f^{-1}(0)| = |f^{-1}(1)|$

Problem: Given $f: \mathbb{B} \to \mathbb{B}$ $(n=1)$ determine whether f is constant or balanced. two evaluations

Classical solution: Evaluate f at 0 & 1 and compare the results.

Quantum solution: black box (also called a quantum oracle)



where $U_f : (\mathbb{C}^2)^{\otimes 2} \to (\mathbb{C}^2)^{\otimes 2}$

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

$$H_1 U_f H \otimes H |01\rangle = \frac{1}{2} H_1 U_f (|0\rangle + |1\rangle)(|0\rangle - |1\rangle)$$

$$= \frac{1}{2} H_1 \left((-1)^{f(0)}|0\rangle + (-1)^{f(1)}|1\rangle \right)(|0\rangle - |1\rangle)$$

$(|0\rangle+|1\rangle)/\sqrt{2}$ $\qquad$ $(|0\rangle-|1\rangle)/\sqrt{2}$

$$U_f |x\rangle(|0\rangle - |1\rangle) = \begin{cases} |x\rangle(|0\rangle - |1\rangle) & f(0)=0 \\ |x\rangle(|1\rangle - |0\rangle) & f(0)=1 \end{cases} \Big\} f \text{ is constant}$$

$$\begin{cases} |x\rangle(|x\rangle - |1\oplus x\rangle) & f(0)=0 \\ |x\rangle(|1\oplus x\rangle - |x\rangle) & f(0)=1 \end{cases} \Big\} f \text{ is balanced}$$

$n=1$

$f(0) \neq f(1)$

$$= (-1)^{f(x)} |x\rangle (|0\rangle - |1\rangle)$$

$$= \frac{1}{2\sqrt{2}} \Big[\left((-1)^{f(0)} + (-1)^{f(1)} \right) |0\rangle + \left((-1)^{f(0)} - (-1)^{f(1)} \right) |1\rangle \Big] (|0\rangle - |1\rangle)$$

$$2(-1)^{f(0)} |0\rangle \quad \text{if } f \text{ is constant}$$

$$2(-1)^{f(0)} |1\rangle \quad \text{if } f \text{ is balanced}$$

Measuring first qubit in 2-bases gives

1) If f is constant then $p(0) = 1$
2) If f is balanced then $p(0) = 0$.

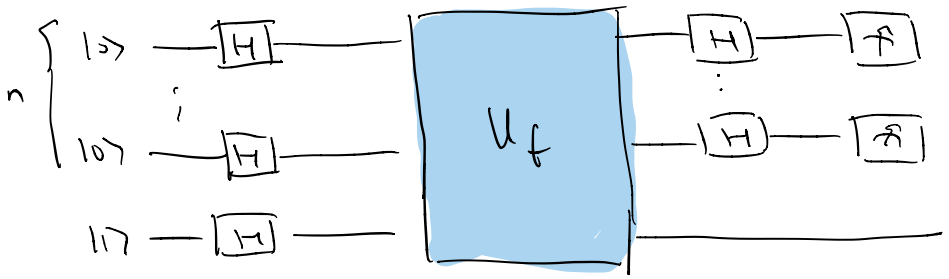
We have evaluated f <u>only</u> once (using U_f)

1) Quantum parallelism : superposition of $|0\rangle$ & $|1\rangle$
2) Quantum interference : destructive & constructive

Deutsch – Jozsa algorithm

Problem Suppose that $f: \mathbb{B}^n \to \mathbb{B}$ is either constant or balanced.

Determine whether f is constant or balanced.



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

$$H^{\otimes n}_{(1\ldots n)} \; U_f \; H^{\otimes (n+1)} \; |0\ldots0\rangle |1\rangle$$

$$= H^{\otimes n}_{(1\ldots n)} \; U_f \; \frac{1}{2^{n/2}} \sum_{x=0}^{2^n-1} |x\rangle \; \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle)$$

$b_{n-1}\cdots b_0 \in \mathbb{B}^n$

$$x = \sum_{i=0}^{n-1} b_i 2^i$$

$$= H^{\otimes n}_{(1\ldots n)} \; \frac{1}{2^{n/2}} \sum_{x=0}^{2^n-1} (-1)^{f(x)} |x\rangle \; \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle)$$

$$H^{\otimes n} |x\rangle = \frac{1}{2^{n/2}} \sum_{y=0}^{2^n-1} (-1)^{x \cdot y} |y\rangle$$

if $\langle x \rangle = b_{n-1}\cdots b_0$ then $x \cdot y = \sum_{i=0}^{n-1} b_i b'_i$

$\langle y \rangle = b'_{n-1}\cdots b'_0$

$$= \sum_y \frac{1}{2^n} \sum_x (-1)^{f(x)+x\cdot y} |y\rangle \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle)$$

↳ if $f(x)$ constant then

$$(-1)^{f(x)} \frac{1}{2^n} \sum_x (-1)^{x\cdot y} = (-1)^{f(x)} \delta_{y,0}$$

$y \neq 0 \quad \sum_{x \mid x\cdot y=0} 1 + \sum_{x \mid x\cdot y=1} -1 = 0$

e.g.

1) $y = 011$
$z = 010$

2) $y = 01$
$z = 01$

we can find z $\quad z\cdot y = 1$

$$\{x \mid x\cdot y = 0\} \underset{x+z \leftarrow x}{\overset{x \mapsto x+z}{\rightleftarrows}} \{x \mid x\cdot y = 1\}$$

if $f(x)$ is balanced then coeff of $|0\rangle$ is

$$\frac{1}{2^n} \sum_x (-1)^{f(x)} = 0$$

Meaning in the z-basis

1) If f is constant then $p(0) = 1$
2) if f is balanced then $p(0) = 0$

We solved the problem by evaluation of f once.
Classically we require $2^n/2 + 1$ evaluations

↳ probabilistic algorithm solving in polynomial time exists.

We have experienced speed up with one query to the quantum oracle.

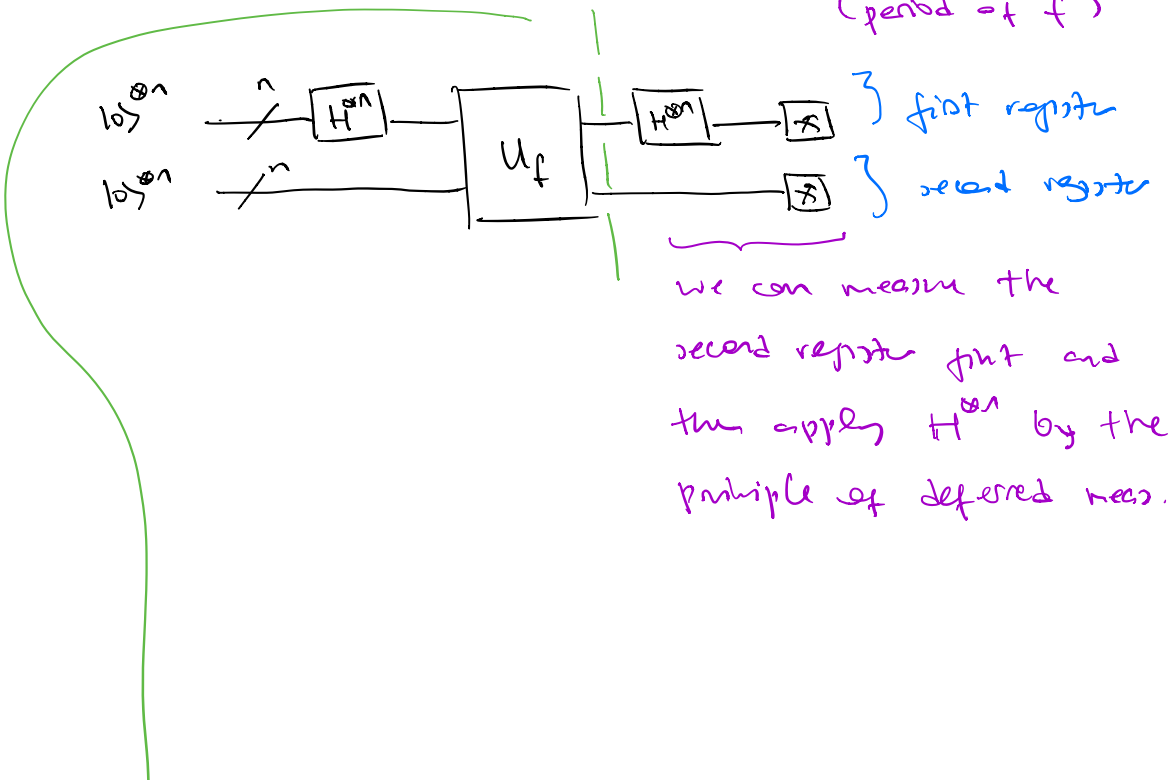
Simon's algorithm

Problem: Given $f: \mathbb{B}^n \to \mathbb{B}^n$ such that

$$f(x) = f(y) \iff y = \begin{cases} x \\ x \oplus a \end{cases} \text{ or}$$

↑ fixed $a \in \mathbb{B}^n$ (period of f)

Find the period a.



$$U_f \, H^{\otimes n}_{(1..n)} \, |0\rangle^{\otimes n} \, |0\rangle^{\otimes n} = U_f \, \frac{1}{2^{n/2}} \sum_{x=0}^{2^n-1} |x\rangle \, |0\rangle^{\otimes n}$$

$$= \frac{1}{2^{n/2}} \sum_x |x\rangle \underbrace{|0\dots0 \oplus f(x)\rangle}_{|f(x)\rangle}$$

Measure the second register

$$\frac{1}{2^{n/2}} \sum_x |x\rangle |f(x)\rangle \xrightarrow[\text{outcome is } f(x_0)]{\text{meas.}} \frac{1}{\sqrt{2}} \left(|x_0\rangle + |x_0 \oplus a\rangle \right) |f(x_0)\rangle$$

Apply $H^{\otimes n}_{(1..n)}$ we obtain

$$\frac{1}{2^{(n+1)/2}} \left(\sum_y \underbrace{(-1)^{x_0 \cdot y} + (-1)^{(x_0 \oplus a)\cdot y}}_{\text{This is } \begin{cases} 0 & \text{if } a\cdot y = 1 \\ 2(-1)^{x_0 \cdot y} & \text{if } a \cdot y = 0 \end{cases}} \right) |y\rangle \, |f(x_0)\rangle$$

$$= \frac{1}{2^{(n-1)/2}} \sum_{y : a\cdot y = 0} (-1)^{x_0 \cdot y} \, |y\rangle \, |f(x_0)\rangle$$

Measuring the first register

$$p(y) = \begin{cases} 0 & a \cdot y = 1 \\ \frac{1}{2^{n-1}} & a \cdot y = 0 \end{cases}$$

Repeat the algorithm to find

$$\{ y_1, y_2, y_3, \cdots, y_{n-1} \}$$ linearly indep. over $\mathbb{Z}_2$.

field two elements

Then solve

$$\left.\begin{aligned} a \cdot y_1 &= 0 \\ &\vdots \\ a \cdot y_{n-1} &= 0 \end{aligned}\right\}$$ Gaussian elimination mod-2 which is polynomial in n.

to find a.

↑ unique non-zero sol.

Classically this problem is hard: requires exponentially many queries to oracle.

Using a quantum oracle $O(n)$ repetitions is enough.

$$BPP \neq BQP$$

↑ relative to the oracle.

Ex $n = 2$ $\quad y_1 = (1,0)$ $\quad \{0,1\} = \mathbb{Z}_2$

$a \cdot y_1 = 0$ $\quad a = (a_1, a_2)$ $\quad a_1, a_2 \in \mathbb{Z}_2$

↳ $a_1 = 0 \Rightarrow a = (0, a_2)$

↑ 1

Quantum Fourier Transform (FT)

Motivation: $H : \mathbb{C}^2 \to \mathbb{C}^2$

encoding

$$H|x\rangle = \frac{|0\rangle + (-1)^x |1\rangle}{\sqrt{2}} \qquad x \in \mathbb{B}$$

information about x is encoded as a relative phase

Since $H^2 = I_{\mathbb{C}^2}$ we have

$$H\left(\frac{|0\rangle + (-1)^x |1\rangle}{\sqrt{2}}\right) = |x\rangle$$

decoding

$|x\rangle \mapsto U|x\rangle$

$|x\rangle\langle x| \mapsto U|x\rangle\langle x|U^\dagger$

e $\quad$ e

More generally $H^{\otimes t} : (\mathbb{C}^2)^{\otimes t} \to (\mathbb{C}^2)^{\otimes t}$

encoding

$$|x\rangle \qquad \frac{1}{2^{t/2}} \sum_k (-1)^{x \cdot k} |k\rangle$$

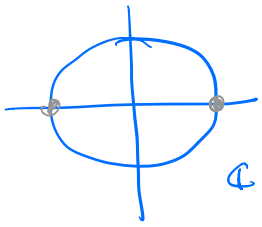
decoding

only ± 1 can occur

We would like to develop a similar procedure for arbitrary phases:

$$\frac{1}{2^{t/2}} \sum_{k=0}^{2^t - 1} e^{2\pi i \varphi k} |k\rangle$$

$0 \le \varphi < 1$



FT: Consider $\mathbb{C}^N$ with canonical orthonormal basis $\{|j\rangle\}_{j=0}^{N-1}$.

$F : \mathbb{C}^N \longrightarrow \mathbb{C}^N$ unitary operator

$$F(|j\rangle) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi i jk/N} |k\rangle$$

Ex $F : \mathbb{C}^2 \longrightarrow \mathbb{C}^2$ is the Hadamard H.

Applying F to an arbitrary vector in $\mathbb{C}^N$:

$$F\left(\sum_{j=0}^{N-1} x_j |j\rangle\right) = \sum_j x_j F(|j\rangle)$$

$$= \sum_j x_j \frac{1}{\sqrt{N}} \sum_k e^{2\pi i jk/N} |k\rangle$$

$$= \sum_k \underbrace{\frac{1}{\sqrt{N}} \sum_j x_j e^{2\pi i jk/N}}_{y_k} |k\rangle$$

$(y_0, \dots, y_{N-1})$ is called the discrete FT of $(x_0, \dots, x_{N-1})$.

Aside: Let $\mathbb{Z}_N = \{0, 1, \dots, N-1\}$

$f : \mathbb{Z}_N \to \mathbb{C} \xmapsto{FT} \hat{f} : \underbrace{\mathrm{Irr}(\mathbb{Z}_N)}_{\mathbb{Z}_N} \to \mathbb{C}$

Same machinery applies to any finite group G.

$H^{\otimes n}$ is the FT corresponding to $G = \mathbb{Z}_2^n$.

Remark $\mathbb{Z}_2^n \not\cong \mathbb{Z}_{2^n}$ different as abelian groups.

Quantum circuit for FT $\qquad \mathbb{C}^N \cong (\mathbb{C}^2)^{\otimes n}$

Let $N = 2^n$ for some $n \in \mathbb{N}$.

Binary representation

An integer $0 \le j < N$ will be represented by

$$j_1 j_2 \cdots j_n \in \mathbb{B}^n$$

that is $\quad j = \sum_{\ell=1}^{n} j_\ell \, 2^{n-\ell}$.

Also $\quad 0.j_\ell j_{\ell+1} \cdots j_m$ will represent

$$\sum_{k=\ell}^{m} j_k / 2^{k-\ell+1} = j_\ell/2 + \cdots + j_m/2^{m-\ell+1}.$$

1) $$F(|j\rangle) = \frac{1}{\sqrt{N}} \sum_{k=0}^{2^n-1} e^{2\pi i jk/N^{2^n}} |k\rangle$$

$|j\rangle = |j_1 \cdots j_n\rangle$, $\frac{1}{\sqrt{N}} = 1/2^{n/2}$, $|k\rangle = |k_1 \cdots k_n\rangle = |k_1\rangle \otimes \cdots \otimes |k_n\rangle$

use

$$= \frac{1}{2^{n/2}} \sum_{k_1} \cdots \sum_{k_n} e^{2\pi i j \left(\sum_{\ell=1}^{n} k_\ell 2^{-\ell} \right)} |k_1\rangle \otimes \cdots \otimes |k_n\rangle$$

$$\bigotimes_{\ell=1}^{n} e^{2\pi i j k_\ell 2^{-\ell}} |k_\ell\rangle$$

$$= \frac{1}{2^{n/2}} \bigotimes_{\ell=1}^{n} \left(\sum_{k_\ell=0}^{1} e^{2\pi i j k_\ell 2^{-\ell}} |k_\ell\rangle \right)$$

$$= \frac{1}{2^{n/2}} \bigotimes_{\ell=1}^{n} \left(|0\rangle + e^{2\pi i j 2^{-\ell}} |1\rangle \right)$$

$$j 2^{-\ell} = (j_1 2^{n-1} + \cdots + j_{n-\ell} 2^{\ell} + \cdots + j_n 2^{0}) 2^{-\ell}$$

$$= (j_1 2^{n-1-\ell} + \cdots + j_{n-\ell} 2^{0} + j_{n-\ell+1} 2^{-1} + \cdots + j_n 2^{-\ell})$$

applying $e^{2\pi i (-)}$ gives 1 $\quad 0.j_{n-\ell+1} \cdots j_n$

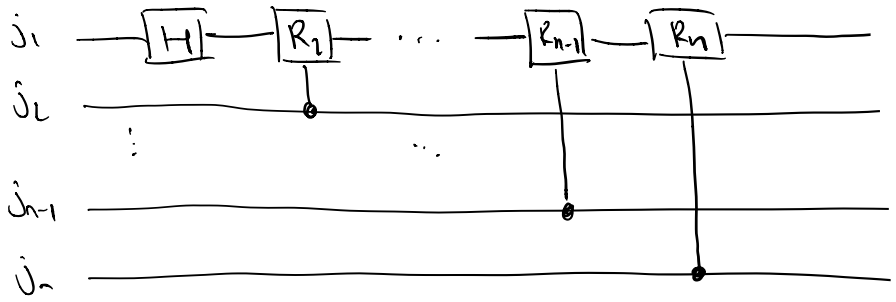
$$= \frac{1}{2^{n/2}} \bigotimes_{\ell=1}^{n} \left(|0\rangle + e^{2\pi i \, 0.j_{n-\ell+1} \cdots j_n} |1\rangle \right)$$

$$= \frac{\left(|0\rangle + e^{2\pi i \, 0.j_n} |1\rangle\right)\left(|0\rangle + e^{2\pi i \, 0.j_{n-1} j_n} |1\rangle\right) \cdots \left(|0\rangle + e^{2\pi i \, 0.j_1 \cdots j_n} |1\rangle\right)}{2^{n/2} \quad \cdots}$$

FT encoding as relative phase

product representation of FT.

2) Let us consider



$$\frac{|0\rangle + e^{2\pi i\, 0.j_1 \cdots j_n}|1\rangle}{\sqrt{2}}$$

where $R_k = \begin{pmatrix} 1 & 0 \\ 0 & e^{2\pi i/2^k} \end{pmatrix}$

$$C(R_n)_{n1} \cdots C(R_2)_{21}\, H_1\, |j_1 j_2 \cdots j_n\rangle$$

control, target

$$\frac{|0\rangle + (-1)^{j_1}|1\rangle}{\sqrt{2}}$$

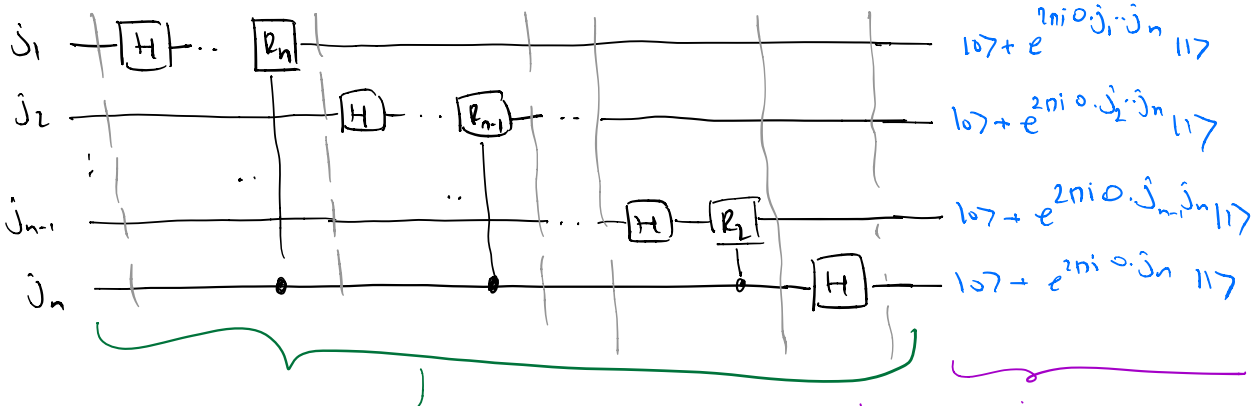
$$(-1)^{j_1} = e^{2\pi i\, j_1/2}$$

$$= e^{2\pi i\, 0.j_1}$$

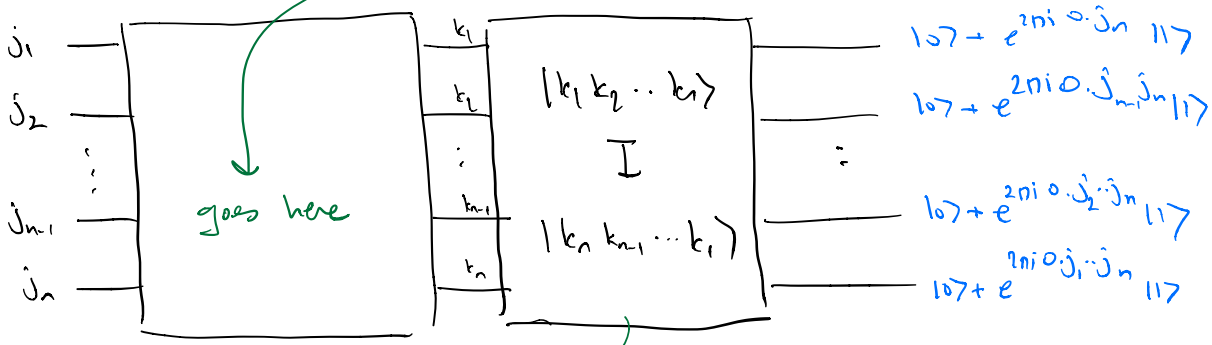
$$= C(R_n)_{n1} \cdots C(R_3)_{31} \left(\frac{|0\rangle + e^{2\pi i\, 0.j_1 j_2}|1\rangle}{\sqrt{2}} \right) |j_2 \cdots j_n\rangle$$

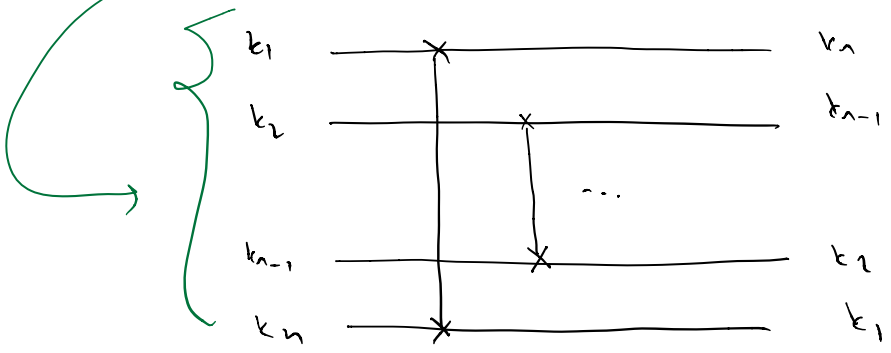
$$= \left(\frac{|0\rangle + e^{2\pi i\, 0.j_1 j_2 \cdots j_n}|1\rangle}{\sqrt{2}} \right) |j_2 \cdots j_n\rangle$$

We apply a similar circuit to $\hat{j}_2, \cdots, \hat{j}_n$:

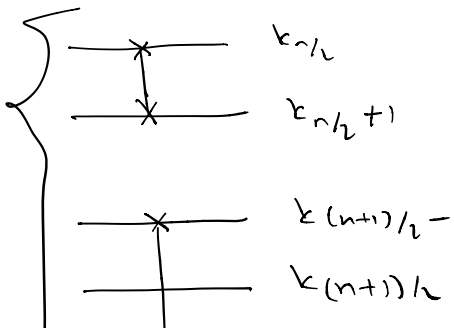


the order is wrong.



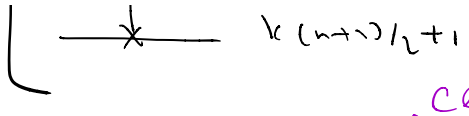


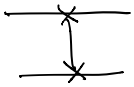
the last gate:



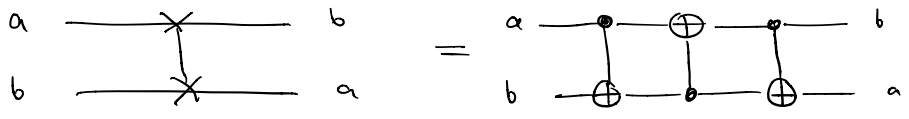
if n is even

if n is odd



Here  is the swap gate

Classical version: CROSSOVER



$$C(X)_{12}\ C(X)_{21}\ C(X)_{12}\ |ab\rangle = |ba\rangle$$

$|a\ \ a\oplus b\rangle$

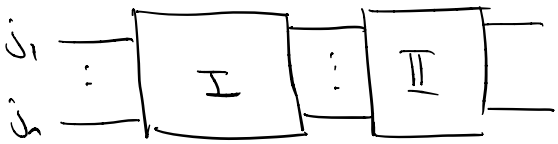
$|a\oplus a\oplus b\ \ a\oplus b\rangle$

b

$|b\ \ b\oplus a\oplus b\rangle$

a

Number of gates



FT

2 CNOTs + 4 Single Qubits

(I) consists of H & $C(R_k)$

$$\sum_{k=0}^{n-1} (2\ \text{CNOTs} + 4\ \text{Single Qubits})\ k + n\ \text{single Qubit}$$

$(2\ \text{CNOTs} + 4\ \text{Single Qubits})\ \frac{(n-1)n}{2}$

(II) consists of at most $\lceil n/2 \rceil$ SWAPs (= 3 CNOTs)

$3\lceil n/2 \rceil$ CNOTs

ceiling function: smallest integer $\geq n/2$

Thus we need $O(n^2)$ CNOTs & Single Qubits.

By Solovay-Kitaev thm we need

$O(n^2 \log(n^2/\varepsilon))$ gates in $A_Q = \{H, T, CNOT\}$

to approximate a circuits consisting of $O(n^2)$ CNOT + 1 Qubits.

polynomial

Computing FT of 2^n complex numbers classically (using Fast FT) takes $O(n 2^n)$ gates.

exponential

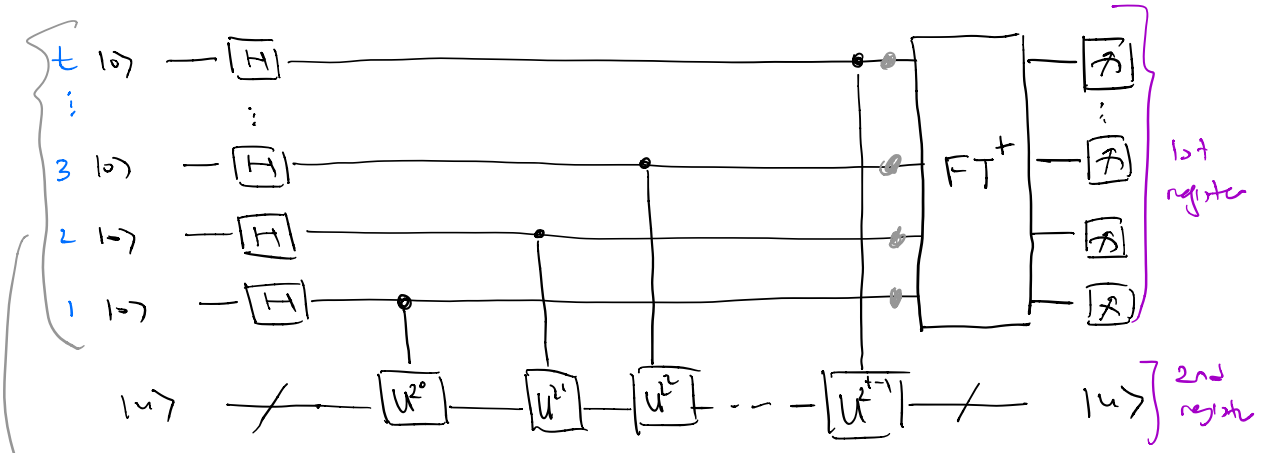
This is the source of the speedup in the algorithms that use FT. In particular Shor's alg.

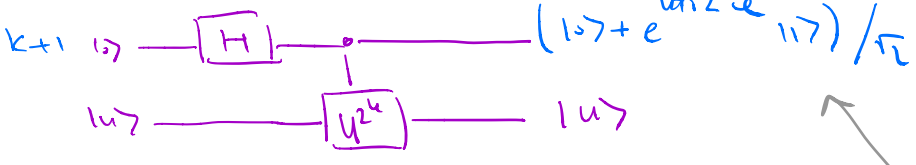
Phase estimation

Goal of the algorithm is to estimate an eigenvalue $e^{2\pi i \varphi}$ of a unitary U given an eigenvector

$$U|u\rangle = e^{2\pi i \varphi}|u\rangle.$$

The circuit:





$$C(U^{2^k})(H \otimes I_N)|0\rangle|u\rangle$$

$$= C(U^{2^k})\left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right)|u\rangle$$

$$= \left(\frac{|0\rangle + e^{2\pi i 2^k \varphi}|1\rangle}{\sqrt{2}}\right)|u\rangle$$

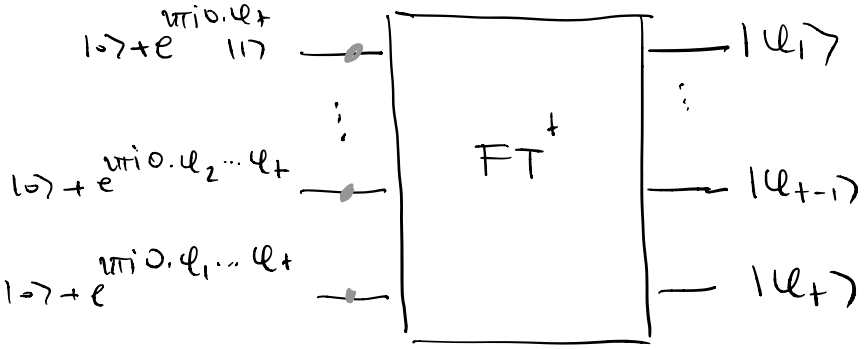
1) Let us assume that — later on we'll assume $\varphi \approx \tilde{\varphi}$

$$\boxed{\varphi = \tilde{\varphi} := \varphi_1/2 + \varphi_2/2^2 + \dots + \varphi_t/2^t}$$

Then

$$e^{2\pi i 2^k \tilde{\varphi}} = e^{2\pi i \left(\varphi_{k+1}/2 + \varphi_{k+2}/2^2 + \dots + \varphi_t/2^{t-k}\right)}$$

$$= e^{2\pi i\, 0.\varphi_{k+1}\dots\varphi_t}$$



Before the measurement the state is

$$|\varphi_1\varphi_2\dots\varphi_t\rangle\,|u\rangle$$

call this $\tilde{\varphi}$ since $0.\varphi_1\varphi_2\dots\varphi_t$ its binary rep.

Measuring the first t qubits in the Z-basis reveals the values of φ_k $1 \le k \le t$.

Therefore the output of the measurement is essentially the binary representation of $\tilde{\varphi} = \varphi$.

2) If $U \neq \tilde{U}$ then we need to do a bit of more work to show that the output of the circuit approximates U.

a) Recall that for $|u\rangle \in V$ the circuit gives

$$C(U^{2^{t-1}}) \cdots C(U^{2^0}) (H_1 \otimes \cdots \otimes H_t) |0 \cdots 0\rangle |u\rangle$$

$$= \frac{1}{2^{t/2}} \left(|0\rangle + e^{2\pi i 2^{t-1} \varphi} |1\rangle \right) \cdots \left(|0\rangle + e^{2\pi i 2^0 \varphi} |1\rangle \right) |u\rangle$$

$$= \left(\frac{1}{2^{t/2}} \sum_{k=0}^{2^t - 1} e^{2\pi i \varphi k} |k\rangle \right) |u\rangle$$

We'll apply the inverse FT

Claim: $G : \mathbb{C}^N \longrightarrow \mathbb{C}^N$

$$G(|k\rangle) = \frac{1}{\sqrt{N}} \sum_{\ell=0}^{N-1} e^{-2\pi i k \ell / N} |\ell\rangle$$

Then $G = F^\dagger$.

proof

$$G F |j\rangle = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi i j k / N} F^\dagger |k\rangle$$

$$= \frac{1}{N} \sum_{k,\ell} e^{2\pi i (j-\ell) k / N} |\ell\rangle$$

$$= \sum_{\ell} \left(\frac{1}{N} \sum_{k} e^{2\pi i (j-\ell) k / N} \right) |\ell\rangle$$

$\ell \mid N \quad k$

Let $\Delta = j - \ell$ and

$$\omega = e^{2\pi i \Delta / N}$$

Then

$$\frac{1}{N} \sum_{k=0}^{N-1} \omega^k = \begin{cases} 1 & \Delta = 0 \\ 0 & \Delta \neq 0 \end{cases}$$

when $\Delta \neq 0$ comes from

$$\sum_{k=0}^{N-1} \omega^k = \frac{1 - \omega^N}{1 - \omega} = 0 \quad \text{since} \quad \omega^N = 1.$$

$$= \sum_{\ell} \delta_{j\ell} \, |\ell\rangle = |j\rangle.$$

So $GF = I_{\mathbb{C}^N}$. Similar $FG = I_{\mathbb{C}^N}$.

Therefore $G = F^\dagger$.

Next we apply the inv FT

$$F^\dagger\left(\frac{1}{2^{t/2}}\sum_{k=0}^{2^t-1} e^{2\pi i \varphi k}|k\rangle\right)$$

$$= \frac{1}{2^{t/2}}\sum_{k=0}^{2^t-1} e^{2\pi i \varphi k} F^\dagger(|k\rangle)$$

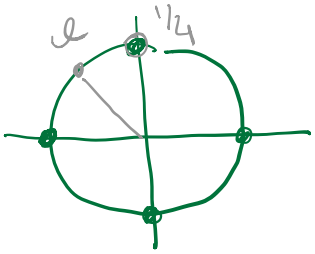
$$= \frac{1}{2^t}\sum_{k,\ell} e^{2\pi i \varphi k} e^{-2\pi i k\ell/2^t}|\ell\rangle \quad (*)$$

b) Let b be an integer in $\{0,1,\ldots,2^t-1\}$ such that

$$0 \le \underbrace{\varphi - b/2^t}_{\delta} \le 2^{-t}$$

$b/2^t$ the best approximation $\le \varphi$

e.g. $t=2$



$0 \le \varphi < 1$

$b = 1$

The coefficient of $|b+\ell\rangle$ in $(*)$ is given by

$$\alpha_\ell = \frac{1}{2^t}\sum_k \left[\underbrace{e^{2\pi i\left(2^t\varphi - (b+\ell)/2^t\right)}}_{\omega}\right]^k$$

$$\frac{1-\omega^{2^t}}{1-\omega}$$

$$= \frac{1}{2^t} \frac{1 - e^{2\pi i (2^t u - (b+\ell))}}{1 - e^{2\pi i (2^t u - (b+\ell))/2^t}}$$

$$(2^t u - (b+\ell))/2^t = [2^t (u - b/2^t) - \ell]/2^t$$

$$= s - \ell/2^t$$

$$= \frac{1}{2^t} \frac{1 - e^{2\pi i (2^t s - \ell)}}{1 - e^{2\pi i (s - \ell/2^t)}}$$

Let $\epsilon > 0$ be a real number. (error tolerance)

For $0 \le m \le 2^t - 1$ let

$$p(|m - b| > \epsilon)$$

denote the probability of obtaining m such that $|m - b| > \epsilon$.

↳ when we meas. at the end of the circuit.

This probability is given by

$$p(|m - b| > \epsilon) = p(|\ell| > \epsilon) \quad \sum_{\ell} \alpha_\ell |b+\ell\rangle$$

↑ state before the meas.

$m = b + \ell$ where $-2^{t-1} < \ell \le 2^{t-1}$

e.g. $t = 2$ $\{0, 1, 2, 3\}$ $-2 < \ell \le 2$

$b-1$ b $b+1$ $b+2$

$$= \sum_{-2^{t-1} < \ell \le -(\ell+1)} |\alpha_\ell|^2 + \sum_{\ell+1 \le \ell \le 2^{t-1}} |\alpha_\ell|^2$$

$|1 - e^{2\pi i\theta}| \le |1| + |e^{2\pi i\theta}| = 2$

We have

$$|\alpha_\ell| = \frac{|1 - e^{2\pi i (2^t\delta - 1)}|}{2^t |1 - e^{2\pi i (2^t\delta - 1)/2^t}|} \le \frac{1}{2^{t-1} |1 - e^{\pi i (2^t\delta - 1)/2^t}|}$$

θ

$\to$ $|1 - e^{i\gamma}| \ge \frac{2|\gamma|}{\pi}$ if $-\pi \le \gamma \le \pi$

$\uparrow$ calculus

$\gamma = 2\pi(\delta - \ell/2^t)$

$-\pi \le 2\pi(\delta - \ell/2^t) \le \pi$

since $0 \le \delta \le 2^{-t}$ and

$-2^{t-1} < \ell \le 2^{t-1}$

Therefore

$$|\alpha_\ell| \le \frac{1}{2^{t+1}(\delta - \ell/2^t)}$$

$$\Rightarrow |\alpha_\ell|^2 \le \frac{1}{4} \frac{1}{(2^t\delta - \ell)^2}$$

This gives us

$$P(|m-b|>\epsilon) = \sum_{-2^{t-1} < \ell \le -(\epsilon+1)} |\alpha_\ell|^2 + \sum_{\epsilon+1 \le \ell \le 2^{t-1}} |\alpha_\ell|^2$$

$$\le \frac{1}{4}\left(\sum_{\cdots} \frac{1}{(2^t\delta - \ell)^2} + \sum_{\cdots} \frac{1}{(2^t\delta - \ell)^2} \right.$$

$$\le \frac{1}{4}\left(\sum_{\cdots} \frac{1}{\ell^2} + \sum_{\cdots} \frac{1}{(\ell-1)^2} \right)$$

$0 \le 2^t\delta \le 1$ since $0 \le \delta \le 2^{-t}$

$$\sum_{(\epsilon+1) \le \ell < 2^{t-1}} \frac{1}{\ell^2} \le \sum_{\epsilon \le \ell \le 2^{t-1}-1} \frac{1}{\ell^2}$$

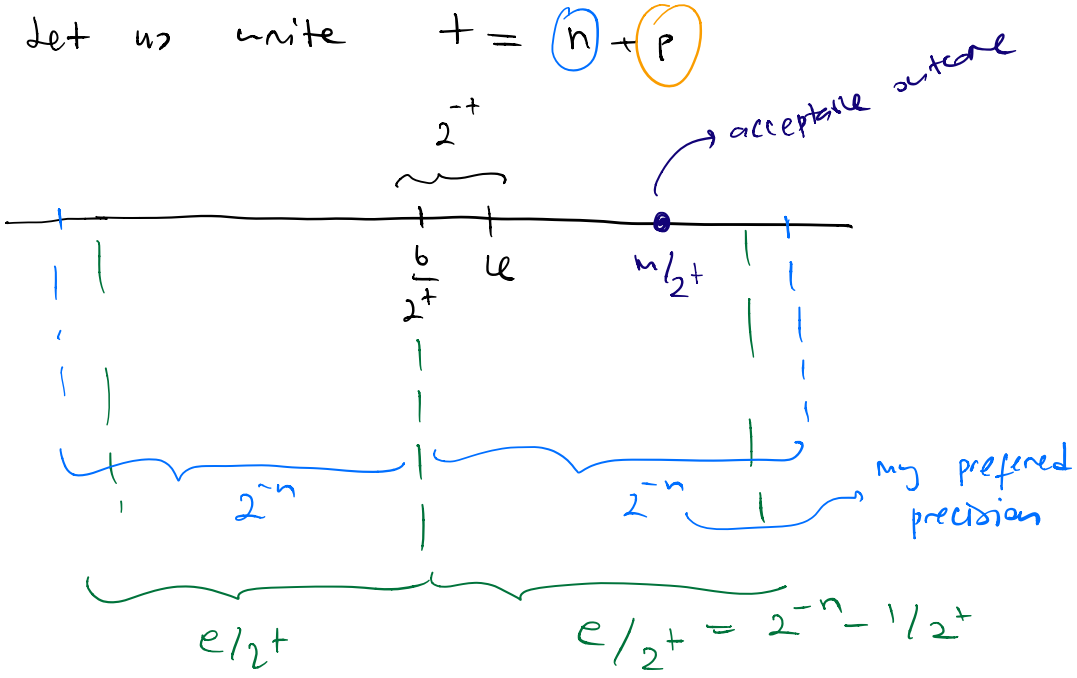
$$\le \frac{1}{2} \sum_{\epsilon \le \ell \le 2^{t-1}-1} \frac{1}{\ell^2} \le \frac{1}{2} \int_{\epsilon-1}^{\infty} \frac{1}{\ell^2}\, d\ell$$

($\epsilon > 1$)

$$= \frac{1}{2(\epsilon+1)}$$

$$P(|m-b|>\epsilon) \le \underbrace{\frac{1}{2(\epsilon+1)}}_{\text{does not depend on } t.}$$

Let us write $t = n + p$



To have accuracy 2^{-n} set $e = \underbrace{2^{t-n} - 1}_{2^p - 1}$

Given $\varepsilon > 0$ we would like to have

$$p(|m-b| \le e) = 1 - p(|m-b| > e) \geqslant 1 - \varepsilon$$

$$\geqslant \frac{1}{2(e-1)}$$

To achieve this choose p such that

$$\underbrace{e}_{2^p - 1} = 1 + \frac{1}{2\varepsilon}$$

Then

$$p = \log\left(2 + \frac{1}{2\varepsilon}\right)$$

Therefore to have (accurate to n bits)

1) approximation accuracy 2^{-n}

2) success probability $\geq 1-\varepsilon$

we need

$$t = n + \lceil \log\left(2 + \frac{1}{2\varepsilon}\right) \rceil$$

(↑ ceiling function)

many qubits in the first register.

Application to order-finding

(no common factor)

Order finding: Given $x < N$ positive integers find the **order** of x mod N i.e. the smallest positive integer r s.t.

$$x^r = 1 \mod N.$$

No efficient classical algorithm exists.

Let $L = \lceil \log N \rceil$.

We will apply the phase estimation to

$$U : (\mathbb{C}^2)^{\otimes L} \longrightarrow (\mathbb{C}^2)^{\otimes L}$$

$$U|y\rangle = \begin{cases} |xy \mod N\rangle & \text{if } 0 \leq y \leq N-1 \\ |y\rangle & \text{if } N \leq y \leq 2^L - 1. \end{cases}$$

U is unitary because $\gcd(x, N) = 1$.

Since $U^r = I_{(\mathbb{C}^2)^{\otimes L}}$ (order r) eigenvalues λ of U satisfy $\lambda^r = 1$ (r-th root of unity)

For $0 \le s \le r-1$

$$|u_s\rangle = \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} e^{-2\pi i s k / r} |x^k\rangle$$

is an eigenvector corresponding to the eigenvalue $\lambda = e^{2\pi i s/r}$.

To see this

$$U|u_s\rangle = \frac{1}{\sqrt{r}} \sum_k e^{-2\pi i s k/r} U|x^k\rangle$$

$U|x^k\rangle = |x^{k+1}\rangle$ since $x^k < N$

$e^{2\pi i s/r} \quad e^{-2\pi i s(k+1)/r}$

$$= e^{2\pi i s/r} \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} e^{-2\pi i s(k+1)/r} |x^{k+1}\rangle$$

$|u_s\rangle$

note $|x^r\rangle = |1\rangle$

$$= e^{2\pi i s/r} |u_s\rangle$$

can use phase estimation to find $\varphi = s/r$.

2) We will modify the phase estimation alg.

$$|0\ldots0\rangle|u\rangle \longmapsto |\tilde{\varphi}_u\rangle|u\rangle$$

$|u\rangle$: eig. vect of U

$\tilde{\varphi}_u$: good estimator for eigval φ_u of $|u\rangle$.

Let us initiate the alg. with

$$|\psi\rangle = \sum_u c_u |u\rangle$$

$\{|u\rangle\}$ an orthonormal set of eig. vect of U.

instead of $|u\rangle$.

Then

$$|0\ldots0\rangle|\psi\rangle \longmapsto \sum_u c_u |\tilde{\varphi}_u\rangle|u\rangle$$

linearity

With prob. $|c_u|^2$ we obtain a good estimator for φ_u, the eig. val of $|u\rangle$.

when we measure at the end of the circuit

If we take

$$t = n + \left\lceil \log\left(2 + \frac{1}{2\varepsilon}\right)\right\rceil$$

then

approximation accuracy for $\varphi_u = 2^{-n}$ and success probability $\geq |c_u|^2(1-\varepsilon)$

next HW

We will apply this to order-finding:

initial state: $\frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} |u_s\rangle$

$$= \frac{1}{\sqrt{r}} \sum_s \frac{1}{\sqrt{r}} \sum_k e^{-2\pi i s k / r} |x^k\rangle$$

$$= \sum_k \left(\frac{1}{r} \sum_{s=0}^{r-1} e^{-2\pi i s k / r} \right) |x^k\rangle$$

$\delta_{k,0}$

$$= |1\rangle$$

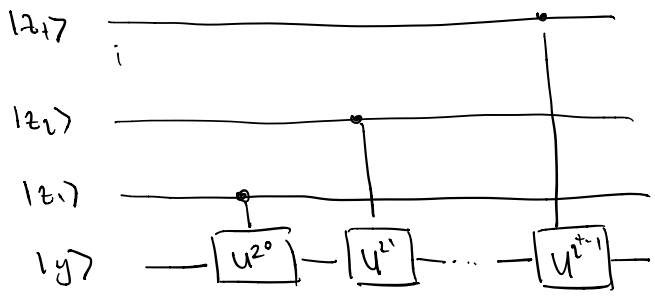
Therefore initially with $|1\rangle$ and taking

$$t = \underbrace{2L+1}_{\text{this is so that cont. frac. alg. works}} + \lceil \log\left(2 + \frac{1}{2\epsilon}\right) \rceil \qquad O(L)$$

we obtain an estimate for $\varphi = s/r$ accurate to $2L+1$ bits with success prob $\geq \frac{1}{r}(1-\epsilon)$

$|c_s|^2$

2) Modular exponentiation



$$|z_1 \dots z_t\rangle |y\rangle \longmapsto |z_1 \dots z_t\rangle U^{z_t 2^{t-1}} \dots U^{z_1 2^0} |y\rangle$$

$$= |z_1 \dots z_t\rangle | \underbrace{x^{z_t 2^{t-1}} \dots x^{z_1 2^0}}_{x^z \text{ where } z = \sum_{i=1}^{t} z_i 2^{i-1}} y \rangle$$

Side remark: Classical complexity of elementary arithmetic operations

addition / subtraction	$O(n)$
multiplication / division	$O(n^2)$
elementary arith.	$O(n^2)$

a) Computing x^{2^j} for $j = 1, \dots, t-1$

$$x \longmapsto x^2 \longmapsto x^4 \longmapsto \dots \longrightarrow x^{2^{t-1}}$$

$O(L^2)$ since $t = O(L)$

There are $\underbrace{t-1}_{O(L)}$ squaring operations.

Total cost $O(L^3)$.

b) Multiply $(x^{z_t 2^{t-1}}) \cdots (x^{z_1 2^0}) = x^z$

$$\underbrace{(t-1)}_{O(L)} \quad \underbrace{(\text{multiplication})}_{O(L^2)} = O(L^3)$$

Therefore we can construct a reversible circuit

$$|z, y\rangle \longmapsto |z, x^z y\rangle$$

using $O(L^2)$ gates.

This can be turned into a quantum circuit

$$|z\rangle|y\rangle \longmapsto |z\rangle|x^z y\rangle$$

which use $O(L^3)$ gates.

3) Recovering $\underbrace{\text{order}}_{r}$ from the $\underbrace{\text{phase}}_{s/r}$

For positive integers $a_0, \ldots, a_N$ the finite continued fraction expression is the expression

$$[a_0, \ldots, a_N] = a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{\cdots + \cfrac{1}{a_N}}}}$$

Ex

$$\frac{31}{13} = 2 + \frac{5}{13} = 2 + \frac{1}{13/5}$$

$$= 2 + \frac{1}{2 + 3/5} = 2 + \frac{1}{2 + 5/3}$$

$$= 2 + \frac{1}{2 + \frac{1}{1 + 2/3}} = 2 + \frac{1}{2 + \frac{1}{1 + \frac{1}{1 + \frac{1}{2}}}}$$

$$\frac{31}{13} = \begin{cases} [2, 2, 1, 1, 2] \\ [2, 2, 1, 1, 1, 1] \end{cases}$$

$$\frac{1}{1 + \frac{1}{1}}$$

In fact, any rational number has a finite cont. frac. expansion.

In the opposite direction given $[a_0, \dots, a_N)$ the n-th convergence is given by

$$[a_0, \dots, a_n) = \frac{p_n}{q_n} \qquad n \le N$$

where p_n & q_n are defined inductively

$$\begin{cases} n = 0: & p_0 = a_0, \quad q_0 = 1 \\ n = 1: & p_1 = 1 + a_0 a_1, \quad q_1 = a_1 \\ 2 \le n \le N: & p_n = a_n p_{n-1} + p_{n-2} \\ & q_n = a_n q_{n-1} + q_{n-2} \end{cases}$$

Ex $[a_0] = \frac{a_0}{1}$

$[a_0, a_1] = a_0 + \frac{1}{a_1} = \frac{1 + a_0 a_1}{a_1}$

Proof is not hard:

$$[a_0, \dots, a_n] = a_0 + \cfrac{1}{\dots \cfrac{1}{a_{n-2} + \cfrac{1}{\underbrace{a_{n-1} + \frac{1}{a_n}}_{= \tilde{a}_{n-1}}}}}$$

$$= [a_0, \dots, a_{n-2}, \tilde{a}_{n-1}]$$

by induction

$$[a_0, \dots, \tilde{a}_{n-1}] = \frac{\tilde{p}_{n-1}}{\tilde{q}_{n-1}}$$

$$= \frac{\tilde{a}_{n-1} p_{n-2} + p_{n-3}}{\tilde{a}_{n-1} q_{n-2} + q_{n-3}}$$

$$= \frac{a_{n-1} p_{n-2} + p_{n-3} + p_{n-2}/a_n}{a_{n-1} q_{n-2} + q_{n-3} + q_{n-2}/a_n}$$

$$= \frac{(p_{n-1} + p_{n-2}/a_n) a_n}{(q_{n-1} + q_{n-2}/a_n) a_n} = \frac{p_n}{q_n}.$$

Fact $\gcd(p_n, q_n) = 1$

This follows from

$$q_n p_{n-1} - p_n q_{n-1} = (-1)^n$$

which is proved by induction:

For $n=1$: $q_1 p_0 - p_1 q_0 = q_1 q_0 - (1 + q_0 q_1) \cdot 1$

$= -1$

In general

$$(a_n q_{n-1} + q_{n-2}) p_{n-1} - (a_n p_{n-1} + p_{n-2}) q_{n-1}$$

$$= a_n (q_{n-1} p_{n-1} - p_{n-1} q_{n-1}) - (-1)^{n-1}$$

$$= -(-1)^{n-1}$$

$$= (-1)^n.$$

Thm Let x be a rational number and p/q be a rational number such that

$$\left| p/q - x \right| \leq \frac{1}{2q^2}$$

Then p/q is a convergent of x.

Pf Assume $x \neq p/q$.

Let $p/q = [a_0, \cdots, a_n]$

and define p_i & q_i $0 \le i < n$.

Let $\delta = 2q^2 \left(x - p/q \right)$.

Then $|\delta| = 2q^2 \left| x - p/q \right| \le 1$.

If we let

$$\lambda = 2 \left(\frac{q_n p_{n-1} - p_n q_{n-1}}{\delta} \right) - \frac{q_{n-1}}{q_n}$$

($(-1)^n$ over the numerator)

then

$$x = \frac{\lambda p_n + p_{n-1}}{\lambda q_n + q_{n-1}}$$

(p_{n+1} over the numerator, q_{n+1} over the denominator)

(verify)

This means that

$$x = [a_0, \cdots, a_n, a_{n+1}] \quad \text{where } a_{n+1} = \lambda.$$

Then

$$\lambda = \frac{2(-1)^n}{\delta} - \frac{q_{n-1}}{q_n}$$

if $\delta > 0$ choose n even

if $\delta < 0$ choose n odd

$$[a_0, \dots, a_n] = \begin{cases} [a_0, \dots, a_{n-1}, \overbrace{1}^{a_{n+1}}] & a_n > 1 \\ [a_0, \dots, \underbrace{a_{n-1}+1}_{\tilde{a}_{n-1}}] & a_n = 1 \end{cases}$$

$$= \frac{2}{|\delta|} - \underbrace{\frac{q_{n-1}}{q_n}} > \frac{2}{1} - \frac{1}{1} = 1$$

$q_n = a_n q_{n-1} + q_{n-1}$

$q_n > q_{n-1}$

So λ is a rational number s.t. $\lambda > 1$.

Let $\lambda = [b_0, \dots, b_m)$. Then we have

$$x = [\underbrace{a_0, \dots a_n}, b_0, \dots, b_m)$$

p/q is the n-th convergent of x

□

Efficiency of computing cont. frac. expansion

Let p/q be a rational number with $p > q$ and $\gcd(p, q) = 1$.

In the expansion

$$p/q = [a_0, \dots, a_N)$$

we have

$$p_N q_N \geq 2^{\lfloor N/2 \rfloor}$$ ← floor function

since $q_n = a_n q_{n-1} + q_{n-2}$

$$\geq 2 q_{n-2}$$

similarly $p_n \geq 2 p_{n-2}$

Therefore if p & q are L bit integers then

$$N \text{ is } O(\log p) = O(L)$$

So to compute $[a_0, \ldots, a_N)$ we need to perform $O(L)$ split & invert steps each of which requires $O(L^2)$ gates:

$$\text{Total \# gates} = O(L^3).$$

Now approximating $t = \underbrace{n}_{2L+1} + p$

$$\varphi = s/r$$

accurate to $2L+1$ bits we have

$$|s/r - \tilde{\varphi}| \leq \underbrace{2^{-(2L+1)}}_{r \leq N \leq 2^L}$$

$$2^{-(2L+1)} = \frac{1}{2(2^L)^2}$$

$$|s/r - \tilde{\varphi}| \leq \frac{1}{2r^2}$$

Therefore we can apply the earlier thm to recover

s/r with cont. frac. exp.

Recall the thm:

Thm Let x be a rational number and p/q be a rational number such that

$$\left|\frac{p}{q} - x\right| \leq \frac{1}{2q^2}$$

Then p/q is a convergent of x.

$\varphi = s/r$ $\quad \tilde{\varphi}$ $\quad q = r$

Putting pieces together

Order finding: Given $x < N$ and $\gcd(x, N) = 1$ find the order $r > 0$ such that

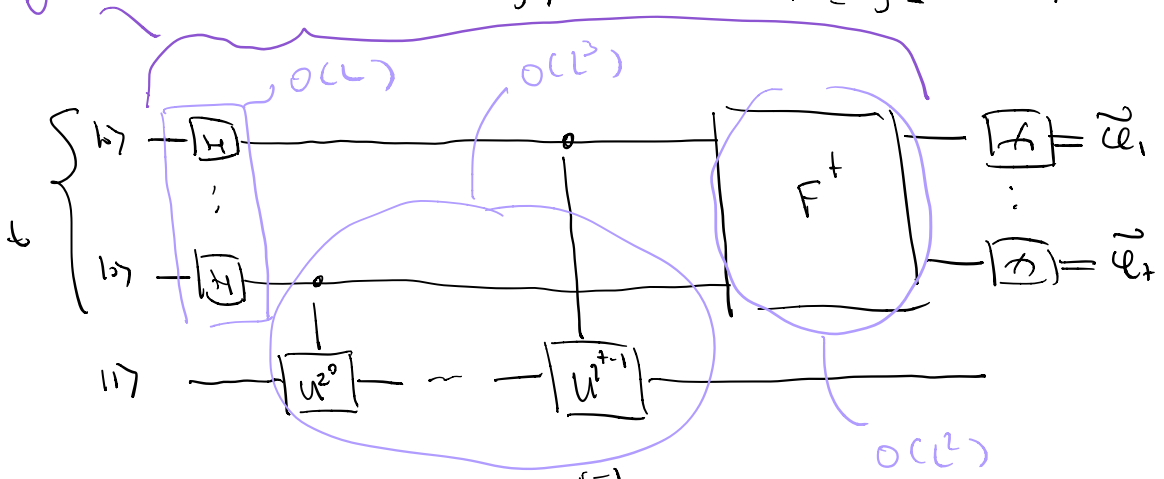
$$x^r = 1 \mod N.$$

Procedure:

Let $L = \lceil \log N \rceil$

1) Phase estimation

$$U|y\rangle = \begin{cases} |xy\rangle & 0 \le y \le N-1 \\ |y\rangle & N \le y \le 2^L - 1 \end{cases}$$



$$|1\rangle = \frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} |u_s\rangle$$

$$U|u_s\rangle = e^{2\pi i \varphi_s} |u_s\rangle \qquad \varphi_s = s/r$$

Take $t = \boxed{2L+1} + \boxed{\lceil \log\left(2 + \frac{1}{2\epsilon}\right) \rceil}$

the with success prob. $\geq \frac{1}{r}(1-\varepsilon)$

we obtain

$$|\varphi_s - \tilde{\varphi}| \leq 2^{-(2L+1)} \leq \frac{1}{2r^2}$$

($\frac{1}{r} \leq |c_{u_s}|^2$)

where $\tilde{\varphi} = 0.\tilde{\varphi}_1 \cdots \tilde{\varphi}_{2L+1}$.

2) Continued-fraction algorithm

Compute the cont. frac. expansion of $\tilde{\varphi}$ and the i-th convergent p_i/q_i for $i = 0, 1, \ldots$. Stop if for some $i = n$ we have

$$\left| \frac{p_n}{q_n} - \tilde{\varphi} \right| \leq 2^{-(2L+1)} \quad (*).$$

Note that if the algorithm terminates, i.e. $\exists n$ satisfying $(*)$, then it is unique:

Suppose there are two such numbers p/q & p'/q' s.t.

$$|p/q - \tilde{\varphi}| \leq 2^{-(2L+1)}$$

$$|p'/q' - \tilde{\varphi}| \leq 2^{-(2L+1)}$$

where $p, p', q, q' \leq 2^L$

$$p' > p \quad , \quad q' > q$$

$$\gcd(p,q) = \gcd(p',q') = 1.$$

Then

$$\left| p/q - p'/q' \right| = \left| p/q - \tilde{u} + \tilde{u} - p'/q' \right|$$

$$\leq \left| p/q - \tilde{u} \right| + \left| \tilde{u} - p'/q' \right| \leq 2^{-2L}.$$

But

$$\left| p/q - p'/q' \right| = \frac{|q'p - qp'|}{qq'}$$

$$> \frac{|q'p - qp'|}{2^{2L}} \geq \frac{1}{2^{2L}} = 2^{-2L}$$

contradiction.

a) If $\gcd(s,r) = 1$ then

$$x^{q_n} = 1 \mod N, \text{ thus } r = q_n.$$

Thm above guarantees that $\exists\, n \leq M$ s.t.

coprime $\frac{p_n}{q_n} = \frac{s}{r}$ coprime $\Big\}$ $r = q_n$, $s = p_n$

where $\left| s/r - \tilde{u} \right| \leq 2^{-(2L+1)} \leq 1/2r^2$

b) If $\gcd(s,r) \neq 1$ then repeat the algorithm.

$$p(\gcd(s,r)=1) \geq p(s \leq r \text{ and } s \text{ prime})$$

probability that s is coprime r

$$= \frac{\pi(r)}{r}$$ } # primes $\leq r$

Nielsen-Chuang Appendix

$$\geq \frac{r/2\log r}{r}$$

$$= \frac{1}{2\log r} \geq \frac{1}{2L}$$

$r \leq N \leq 2^L$

Therefore repeating the algorithm $O(L)$ times we will obtain with high probability a phase s/r such that

$$\gcd(s,r) = 1.$$

So we can use part (a) to extract r.

Performance: $O(L^4)$ ~ this becomes $O(L^3)$ ⇐ This can be pulled down to $O(1)$.

$$\Big[\underbrace{(\text{phase estimation})}_{O(L^2)} + \underbrace{(\text{cont. frac.})}_{O(L^3)}\Big] \cdot \underbrace{(\text{repeat})}_{O(L)}$$

Shor's factoring algorithm

Factoring problem: Given a composite integer N find its prime factors.

This problem can be reduced to order-finding.

1) Thm: Let N be an L bit composite number. If x is a non-trivial solution to

$$x^2 = 1 \mod N \quad \text{in the range } 1 \le x \le N,$$

where non-trivial solution means

$$x \neq 1 \mod N \quad \text{or} \quad x \neq -1 \mod N,$$

then at least one of

$$\gcd(x-1, N) \quad \text{or} \quad \gcd(x+1, N)$$

is a non-trivial factor of N. It can be computed using $O(L^3)$ operations.

Proof: Since $x^2 = 1 \mod N$ this means

$$N \text{ divides } x^2 - 1 = (x-1)(x+1).$$

Then N must have a common divisor either with $(x-1)$ or $(x+1)$

Since x is non-trivial solution $1 < x < N-1$. So $0 < x-1 < x+1 < N$.

Finally we can compute

$\gcd(x-1, N)$ and $\gcd(x+1, N)$

using Euclid's algorithm. ($O(L^3)$ operations)

Euclid's algorithm to find $\gcd(a,b)$

Suppose $a > b$

$O(L)$

$a = k_1 b + r_1$

$b = k_2 r_1 + r_2$

$r_1 = k_3 r_2 + r_3$

$\vdots$

$r_i = k_{i+2} r_{i+1} + r_{i+2}$ ← each step $O(L^2)$

$\vdots$

$r_m = k_{m+2} r_{m+1} + 0$ algorithm halts.

$r_{m+1} = \gcd(a,b)$

$O(L)$

steps $\leq 2 \lceil \log a \rceil$ follows from

claim: $r_{i+2} \leq r_i / 2$

1) $r_{i+1} \leq r_i/2$ then $r_{i+2} < r_{i+1} \leq r_i/2$

2) $r_{i+1} > r_i/2$

$r_i = 1 \cdot r_{i+1} + r_{i+2}$

$$\Rightarrow \quad r_{i+2} = r_i - r_{i+1} \leq r_i/2$$

Total # operations $= O(L^2) \cdot O(L) = O(L^3)$.

□

2) Thm: Suppose $N = p_1^{\lambda_1} \cdots p_m^{\lambda_m}$ is a prime factorization of an odd composite positive number N.

Let x be chosen uniformly at random such that

$$\left.\begin{array}{l} 1) \quad 1 \leq x \leq N-1 \\ 2) \quad x \text{ is coprime to } N. \end{array}\right\} \text{ i.e. } x \in \mathbb{Z}_N^\times$$

Let r be order of x mod N.

Then

$$P(r \text{ even } \& \; x^{r/2} \neq -1 \bmod N) \geq 1 - \frac{1}{2^{m-1}}$$

Aside:

Let $\mathbb{Z}_N$ denote the abelian group of integers modulo N:

$$\mathbb{Z}_N = \{0, 1, \ldots, N-1\} \text{ and}$$

this is an abelian group under addition modulo N.

Note that if M divides N then

$$\phi : \mathbb{Z}_N \longrightarrow \mathbb{Z}_M$$
$$k \longmapsto k \bmod M$$

is a group homomorphism, i.e.

$$\phi(\underbrace{a+b}_{\bmod N}) = \underbrace{\phi(a) + \phi(b)}_{\bmod M}.$$

Chinese remainder theorem

$$\mathbb{Z}_N \xrightarrow{\cong} \mathbb{Z}_{p_1^{\alpha_1}} \times \mathbb{Z}_{p_2^{\alpha_2}} \times \cdots \times \mathbb{Z}_{p_n^{\alpha_n}}$$
$$k \longmapsto (k \bmod p_1^{\alpha_1}) \cdots , k \bmod p_n^{\alpha_n})$$

is an isomorphism of abelian groups.

The set of elements in $\mathbb{Z}_N$ that are invertible modulo N is a group under multiplication:

$a \in \mathbb{Z}_N$ is invertible if $\exists\, b \in \mathbb{Z}_N$ s.t. $a \cdot b = 1 \bmod N$.

$$\mathbb{Z}_N^\times = \{ k \in \mathbb{Z}_N \mid k \text{ coprime to } N \}$$

The size of this group is denoted by the Euler φ function $\varphi(N)$.

Ex Let p be a prime

$$\varphi(p^\alpha) = p^{\alpha-1}(p-1)$$

↳ size of $\mathbb{Z}_{p^\alpha}^\times$

Numbers $a < p^\alpha$ which are coprime to p^α

$$a \neq p, 2p, \dots, (p^{\alpha-1}-1)p$$

So

$$\varphi(p^\alpha) = (p^\alpha - 1) - (p^{\alpha-1}-1)$$

$$= p^{\alpha-1}(p-1)$$

By chinese remainder theorem

$$\mathbb{Z}_N^\times \cong \mathbb{Z}_{p_1^{\alpha_1}}^\times \times \cdots \times \mathbb{Z}_{p_m^{\alpha_m}}^\times$$

and

$$\varphi(N) = \prod_{i=1}^{m} \varphi(p_i^{\alpha_i})$$

Thm: If p is an odd prime then $\mathbb{Z}_{p^\alpha}^\times$ is cyclic.

↳ This means that

$$\mathbb{Z}_{p^\alpha}^\times = \{ k^i \mid 1 \le i \le \varphi(p^\alpha) \}$$

Lem: Let p be an odd prime. Let 2^d be the largest power of 2 dividing $\varphi(p^\alpha)$ ($\#$ elements in $\mathbb{Z}_{p^\alpha}^\times$)

Then

$$P\left(\begin{array}{l} 2^d \text{ divides } r \text{ where } r \\ \text{is the order of } k \in \mathbb{Z}_{p^\alpha}^\times \\ \text{mod } p^\alpha \end{array}\right) = \frac{1}{2}$$

(k: randomly chosen element)

Proof: $\mathbb{Z}_{p^\alpha}^\times = \{ k^i \mid 1 \le i \le \varphi(p^\alpha) \}$

(k: called the generator of the cyclic group)

($\varphi(p^\alpha) = p^{\alpha-1}(p-1)$ even, so $d \ge 1$)

$$\mathbb{Z}_{p^\alpha}^\times = \underbrace{\{ k^i \mid i \text{ odd} \}}_{I} \xrightarrow{\times k} \underbrace{\{ k^i \mid i \text{ even} \}}_{II}$$

$$I \xleftarrow{\times k^{-1}} II$$

$I \leftrightarrow II$ are in bijective correspondence

I: i is odd:

$(k^i)^r = 1 \bmod p^\alpha$ implies that

$$\varphi(p^\alpha) \mid ir \implies 2^d \mid \varphi(p^\alpha) \mid ir$$

(ir: i odd; $\varphi(p^\alpha)$: even)

$$\implies 2^d \mid r.$$

II: i is even:

$$(k^i)^{\varphi(p^\alpha)/2} = \left(k^{\varphi(p^\alpha)}\right)^{i/2} = 1^{i/2} = 1 \bmod p^\alpha$$

Thus $r \mid \varphi(p^\alpha)/2$ and $2^d \nmid r$

only $2^{d-1} \mid \varphi(p^\alpha)/2$

Since I $\times$ II are of the same size and only I contributes to the probability we are done. ☐

Proof of Thm: We will show

$$P(r \text{ odd or } x^{r/2} = -1 \bmod N) \leq \frac{1}{2^{m-1}}.$$

Choosing $x \in \mathbb{Z}_N^\times$ uniformly at random is the same as choosing $x_j \in \mathbb{Z}_{p_j^{\alpha_j}}^\times$ uniformly at random for $j = 1, \ldots, m$.

Let r_j be the order of $x_j \bmod p_j^{\alpha_j}$.

Let 2^{d_j} (and 2^d) be the largest power of 2 dividing r_j (and r).

Claim: If r is odd or $x^{r/2} = -1 \bmod N$

then $d_1 = d_2 = \dots = d_m = d$.

a) r is odd: Since $r_j \mid r$ $\forall j$ each r_j is odd as well. Then $d_1 = \dots = d_m = d = 0$.

b) $x^{r/2} = -1 \bmod N$: Then $x^{r/2} = -1 \bmod p_j^{\alpha_j}$ hence $r_j \nmid r/2$. But since r_j is we have $d_j = d$.

We choose $x_1 \in \mathbb{Z}^*_{p_1^{\alpha_1}}$ uniform at random.

Then $x_2 \in \mathbb{Z}^*_{p_2^{\alpha_2}}$ will be chosen from the half of the set since by the Lemma x_2 with $d_2 = d_1$ either belong to I or II.

Similarly for $x_3, \dots, x_m$.

$\frac{1}{2}$ $\frac{1}{2}$

Therefore

$$p\left(r \text{ is odd or } x^{r/2} = 1 \bmod N\right) \leq \left(\frac{1}{2}\right)^{m-1}.$$

knowing d_2 restricts the x_2.

∎

Algorithm: reducing factoring to order finding

O(1) 1) If N is even return 2.

O(L²) 2) Determine whether $N = a^b$ for integers $a \geq 1$ and $b \geq 2$, and if so return the factor a.

When we move to step 3 N is odd & $M \geq 2$.

3) Randomly choose x in the range 1 to $N-1$. If $\gcd(x, N) > 1$ then return $\gcd(x, N)$.

otherwise

4) Use order-finding algorithm to find the order r of x mod N. (x & N coprime)

5) If r even and $x^{r/2} \neq -1 \bmod N$ then compute

prob of this $\geq 1 - \frac{1}{2^{M-1}} \geq \frac{1}{2}$

$\gcd(x^{r/2} - 1, N)$ and $\gcd(x^{r/2} + 1, N)$

and check to see if one of these is a non-trivial factor. If so then return this factor, otherwise the algorithm fails.

# of prime factors of N

Assume that r even & $x^{r/2} \neq -1 \bmod N$.

Then r is the order of x

we have $x^{r/2} \neq 1 \bmod N$.

Therefore one of

$\gcd(x^{r/2}-1, N)$ or $\gcd(x^{r/2}+1, N)$

is a non-trivial factor of N by Thm 1.

Note that $(x^{r/2})^2 = 1 \bmod N$.

Overall runtime of the algorithm is $O(L^3)$ operations and success probability is $O(1)$.

Combining everything we proved the following:

Theorem [Shor]

FACTORING $\in$ BQP.

More on HW 4 ...