Quantum Gates

How to implement a XOR b?

a\b	0	1
0	0	1
1	1	0

but $(a,b) \mapsto a \text{ XOR } b$ not unitary

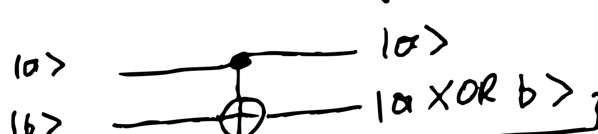
hence $(a,b) \mapsto (a, a \text{ XOR } b)$

(a,b)	$\|00\rangle$	$\|01\rangle$	$\|10\rangle$	$\|11\rangle$
$(a, a \text{ XOR } b)$	$\|00\rangle$	$\|01\rangle$	$\|11\rangle$	$\|10\rangle$

is permutation $\Rightarrow$ unitary

This operation is called „Controlled Not", „CNOT"

In circuit diagrams, this is denoted by



If we identify $|00\rangle, \dots, |11\rangle$ with $e_1, \dots, e_4$

we may represent CNOT by the unitary matrix

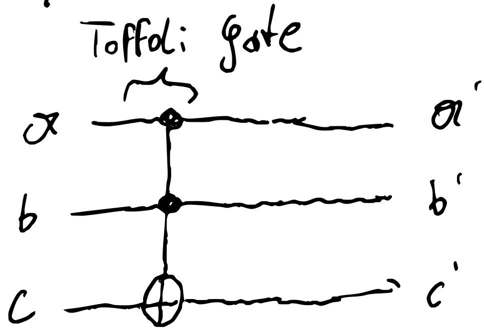
$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

What about „AND"?

a AND b $\Longleftrightarrow$

$a \backslash b$	0	1
0	0	0
1	0	1

- $(a,b) \mapsto (a, a \text{ AND } b)$ not bijective
- requires three qbits $|a,b,c\rangle \mapsto |a, b, c \text{ XOR } (a \text{ AND } b)\rangle$



unitary as composition of q-gates

$\lvert abc\rangle$	000	001	010	011	100	101	110	111
	000	001	010	011	100	101	111	110

if Input $c = |0\rangle \Rightarrow$ Output $c' = |a, b, a \text{ AND } b\rangle$

- extra qbit $|c\rangle$ is called „ancilla" qbit

ancilla = Diener

Deutsch-Jozsa-Algorithm

„Problem": Given $F(x_1, \dots, x_n) \in \{0,1\}$ $x_i \in \{0,1\}$

determine if
- $F = 0$
- $F = 1$
- F is „balanced", i.e. exactly half of $(x_1, \dots, x_n)$ lead to $F(x_1, \dots, x_n) = 1$ the other half $\Rightarrow F(x_1, \dots, x_n) = 0$

- We assume the F satisfies one of the three options
- Not useful in practice, but demonstrates that hard problems can be easier on quantum computers

---

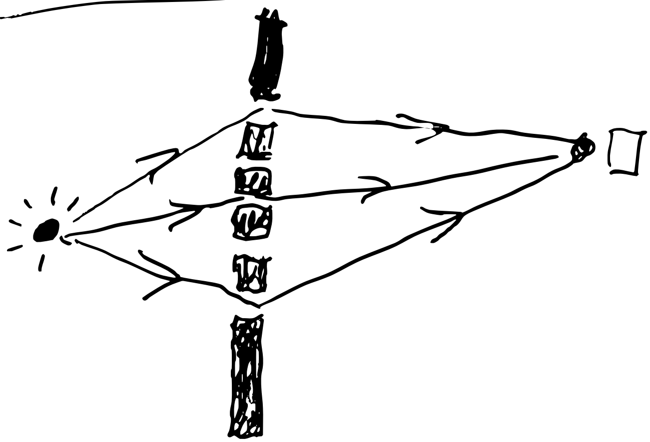
Classical algorithm:

To exclude the „balanced" case, we need to check at least $2^{n-1} + 1$ inputs

$$(x_1, \dots, x_n) \in \{0,1\}^n$$

$\Rightarrow$ Problem size $N = 2^n$, Runtime $N/2 + 1$

Physical inspiration



Intensity at detector is determined by „phase" difference of light-paths

→ Interference

Number the holes with $x \in \{0,1\}^n$ $\Big\}$ 2^n holes

Assume the holes are located such that the light amplitude is $\simeq (-1)^{F(y)}$ at detector if light travels through hole y.

Light intensity at detector is given by

$$\simeq \sum_{y \in \{0,1\}^n} (-1)^{F(y)} \quad \begin{aligned} &= 0 \text{ if } F \text{ is balanced} \\ &= \pm 1 \text{ if } F \text{ is const} \end{aligned}$$

Quantum Algorithm

We may identify $(x_1, \dots, x_n) \in \{0,1\}^n$ with the basis element $|i\rangle$, where $i = x_1 + x_2 2 + x_3 4 + \dots + x_n 2^{n-1}$

In this sense, define $f(|i\rangle) := |f(x_1 \dots, x_n)\rangle$

An essential part of this (and many other algorithms) is the „Query":

Given 1-qbit state $|b\rangle$ and $f(x_1 \dots, x_n) \in \{0,1\}$, define

$$Q_f(|i\rangle \otimes |b\rangle) = |i\rangle \otimes (|b\rangle \oplus f(|i\rangle))$$

Particularly for $|b\rangle = |-\rangle := \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$, we have

$$Q_f(|i-\rangle) = |i\rangle \otimes \frac{1}{\sqrt{2}}\left[f(|i\rangle) - |1\rangle \oplus f(|i\rangle)\right]$$

$$= \begin{cases} |i\rangle \otimes \frac{1}{\sqrt{2}}[|0\rangle - |1\rangle] & f(|i\rangle) = |0\rangle \\ |i\rangle \otimes \frac{1}{\sqrt{2}}[|1\rangle - |0\rangle] & f(|i\rangle) = |1\rangle \end{cases}$$

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

Note that we identify $f(|i\rangle)$ with $f(x_1 \dots, x_n)$ instead of $|f(x_1 \dots, x_n)\rangle$, i.e., the last equality only makes sense for basis element $|i\rangle$

Note that Q_F is unitary since

$$|b\rangle \mapsto (|b\rangle \oplus F(|a\rangle))$$

is permutation for any value of $F(a)$

We make extensive use of the Hadamard gate $\boxed{H}$ given by $\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$

we already know:

$$H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

$$H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$$

Lemma 1: There holds

$$H^{\otimes n}|0^n\rangle := \bigotimes_{i=1}^{n} H|0\rangle = \bigotimes_{i=1}^{n} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

$$= \sum_{\substack{(x_1,\dots,x_n)\\ \in\{0,1\}^n}} \frac{1}{\sqrt{2^n}}|x_1 \dots x_n\rangle = \frac{1}{\sqrt{N}}\sum_{i=0}^{2^n-1}|i\rangle$$

as well as for $i = x_1 + 2x_2 + \dots + 2^{n-1}x_n$

$$H^{\otimes n}|i\rangle := \bigotimes_{i=1}^{n} H|x_i\rangle = \bigotimes_{i=1}^{n}\frac{1}{\sqrt{2}}(|0\rangle + (-1)^{x_i}|1\rangle)$$

$$= \sum_{\substack{(y_1\dots y_n)\\ \in\{0,1\}^n}} \frac{1}{\sqrt{N}}(-1)^{x\cdot y}|y_1 \dots y_n\rangle$$

$$= \frac{1}{\sqrt{N}}\sum_{j=0}^{2^n-1}(-1)^{i\cdot j}|j\rangle, \quad \text{where}$$

$$i\cdot j := x\cdot y := \sum_{k=1}^{n} x_k y_k .$$

The Algorithm reads:

1) Initial state $|0^n\rangle := \underbrace{|0\rangle \otimes \cdots \otimes |0\rangle}_{n\text{-times}}$

2) Apply $\boxed{H}$ to each qbit to obtain uniform state

$$\frac{1}{\sqrt{N}} \sum_{i=0}^{2^n-1} |i\rangle$$

3) Tensorize result with $|-\rangle \Rightarrow \frac{1}{\sqrt{N}} \sum |i-\rangle$

3) Apply query $\boxed{Q_F}$ with $|b\rangle = |-\rangle$ to obtain

$$\frac{1}{\sqrt{N}} \sum_{i=0}^{2^n-1} (-1)^{F(|i\rangle)} |i-\rangle$$

4) Ignore last qbit $|-\rangle$ and apply $\boxed{H}$ to remaining state to obtain (Lemma 1)

$$\frac{1}{N} \sum_{i=0}^{2^n-1} \sum_{j=0}^{2^n-1} (-1)^{i \cdot j} (-1)^{F(|i\rangle)} |j\rangle$$

5) The amplitude of the $j=0 \Leftrightarrow |0^n\rangle$-state is

$$\frac{1}{N} \sum_{i=0}^{2^n-1} \underbrace{(-1)^{i \cdot 0}}_{=1} (-1)^{F(|i\rangle)}$$

- If F is „balanced" $\Rightarrow$

$$\frac{1}{N}\sum_{i=0}^{2^n-1}(-1)^{F(|i\rangle)} = 0$$

- If $F=1 \Rightarrow \frac{1}{N}\sum_{i=0}^{2^n-1}(-1)^{F(|i\rangle)} = -1$
- If $F=0 \Rightarrow \frac{1}{N}\sum_{i=0}^{2^n-1}(-1)^{F(|i\rangle)} = 1$

Complexity of Quantum Alg:

$O(1)$ steps, $O(\log_2(N))$ quantum gates

$O(\log(N))$ qbits $\Rightarrow$ „exponential speedup"

Example for F:

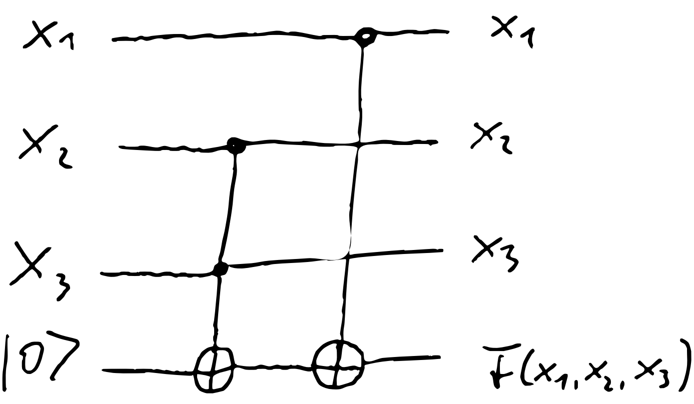
- $F(x_1, x_2, x_3) = \mathrm{mod}(x_1 + x_2 x_3, 2)$

$\Rightarrow$ F is balanced since

$$F(0, x_2, x_3) = 1 - F(1, x_2, x_3)$$

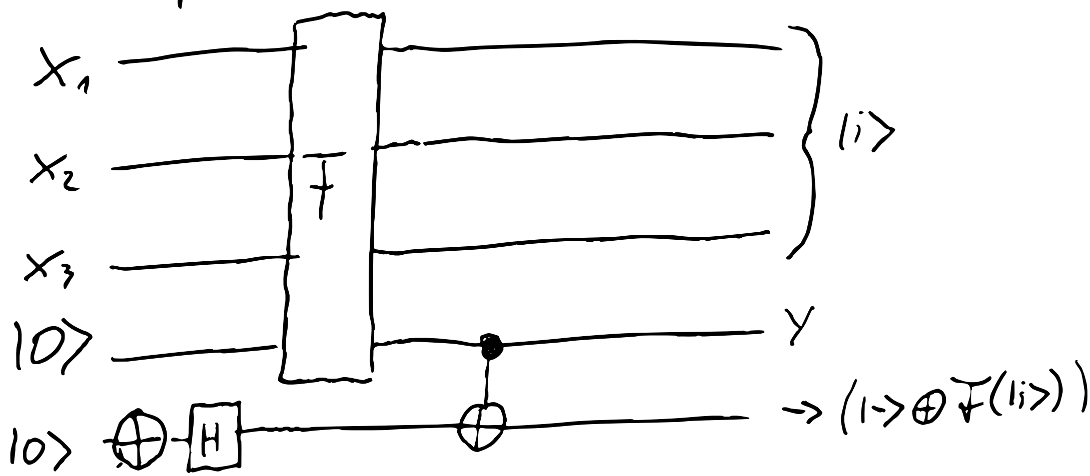
For the algorithm to work, F needs to be a quantum circuit, i.e.

$F(x_1, x_2, x_2) = x_1 \text{ XOR } (x_2 \text{ AND } x_3)$

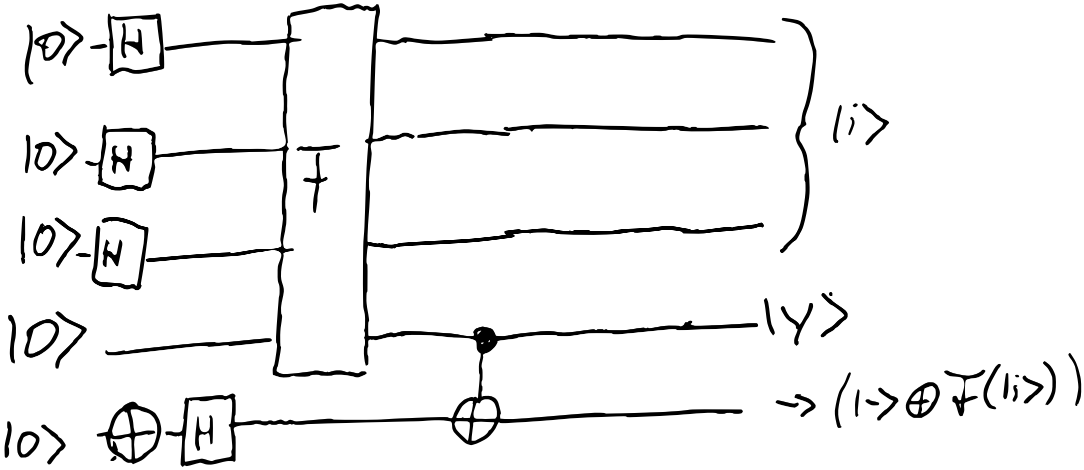


ancilla qbit

Next, implement Query $Q_F(|i-\rangle) = |i\rangle \otimes (|-\rangle \oplus F(|i\rangle))$

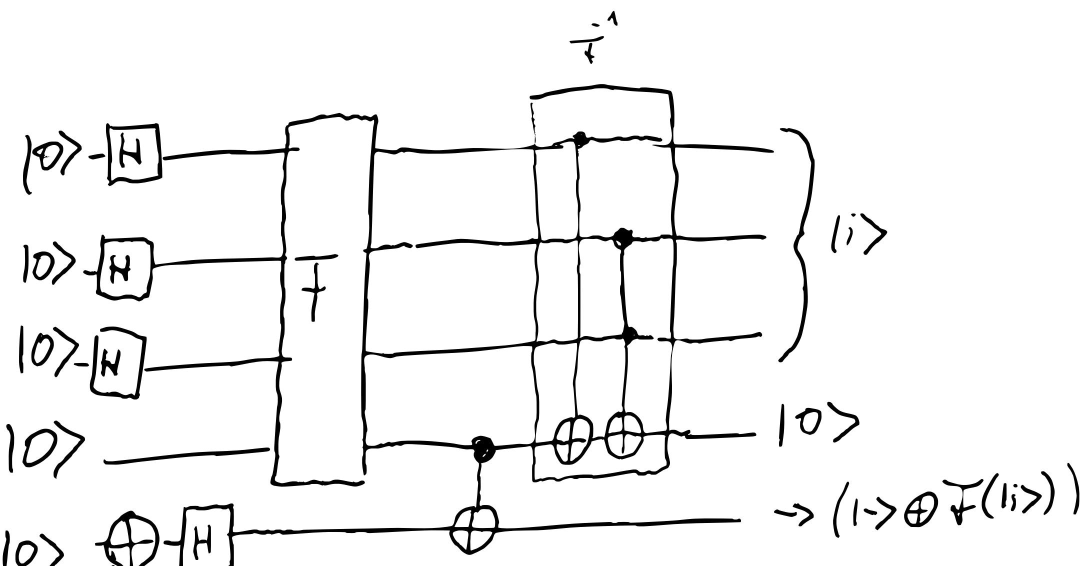


Apply Steps 1–3 of Alg



Problem: $|y\rangle$ and $|i\rangle$ are entangled

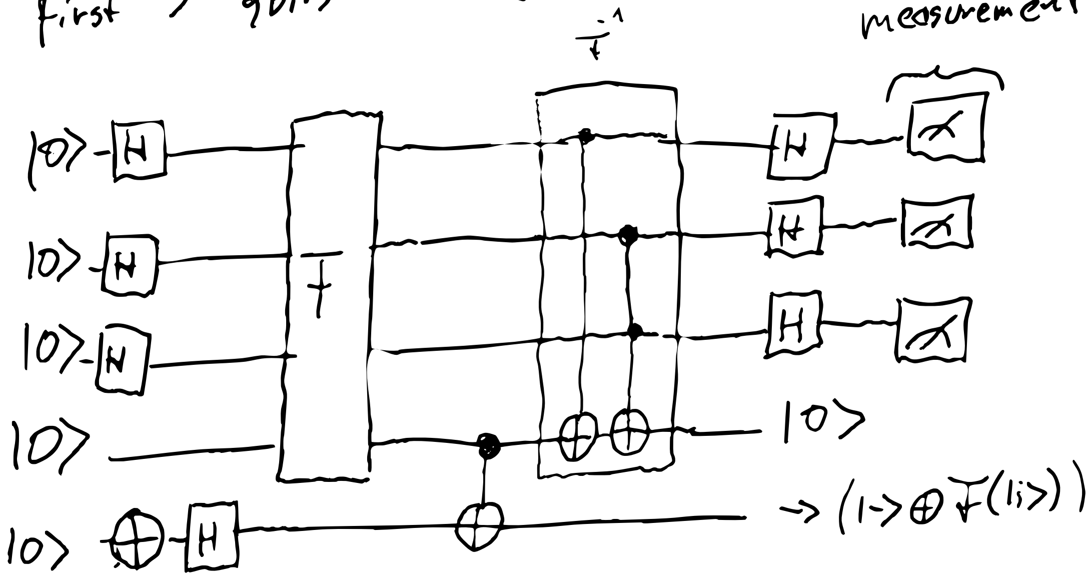
Solution: reverse F such that $|y\rangle = |0\rangle$



Now, we have the desired state

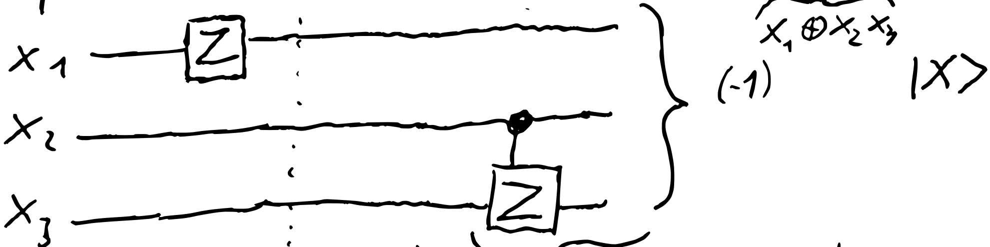
$$|i\rangle \otimes (|-\rangle \oplus F(|i\rangle)) (\otimes |0\rangle)$$

Apply step 4 of Alg and measure the first 3 qbits



The measurment must contain the projection onto $|000\rangle \otimes \text{span}\{|0\rangle, |1\rangle\}^{\otimes 2}$

Note that this is not the most efficient implementation of Q_F



at this point we have $(-1)^{x_1} |x\rangle$

Controlled Z-gate: Z-gate only active if $x_2 = 1$

$\boxed{Z} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$

$\Rightarrow (-1)^{x_2 x_3} |x\rangle$

Remarks on Deutsch-Josza:

- First algorithm with exponential speedup over classical computer

- However, consider the following randomized classical algorithm $R(F)$:

  1) generate two random $x, y \in \{0,1\}^n$

  2) Evaluate $F(x)$, $F(y)$

  3) Output: $R(F) = 1$ if $F(x) = F(y) = 1$

     $R(F) = 0$ if $F(x) = F(y) = 0$

     $R(F)$ „balanced" if $F(x) \neq F(y)$

- The algorithm is correct if F is constant and answers correctly with prob. $\frac{1}{2}$ if F is balanced.

- Apply the Alg k-times with i.i.d random samples to get $(R_1(F), \ldots, R_k(F))$. Return

  1 if $R_1(F) = \ldots = R_k(F) = 1$

  0 if $R_1(F) = \ldots = R_k(F) = 0$

  „balanced" else

$\Rightarrow$ Probability of error 2^{-k}, cost $O(k)$

Remark: Why do you have to „uncompute F" in order to remove entanglement?

After step 3, we have

$$\frac{1}{\sqrt{N}} \sum_{i=0}^{2^n-1} (-1)^{F(|i\rangle)} |i-\rangle$$

but in implementation, we actually have

$$\frac{1}{\sqrt{N}} \sum_{i=0}^{2^n-1} (-1)^{F(|i\rangle)} |i\rangle \otimes |0_i\rangle \otimes |-\rangle$$

for some ancilla qbit $|0_i\rangle$. We may ignore last qbit since not entangled.

Step 4 applys $H^{\otimes n}$ to obtain

$$\frac{1}{N} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (-1)^{F(|i\rangle)} (-1)^{i \cdot j} |j\rangle \otimes |0_i\rangle$$

$\Rightarrow$ Amplitude of $|j\rangle = |0\rangle$ $\Rightarrow$

$$\underline{\frac{1}{N} \sum_{i=0}^{N-1} (-1)^{F(|i\rangle)} |0\rangle \otimes |0_i\rangle}$$

Might be non-zero even for balanced F

Simon's algorithm

First quantum alg. with exp speedup over any (randomized) classical alg.

Given $i, j \in \{0,1\}^n$, define $i \oplus j := (i_1 \oplus j_1, \ldots, i_n \oplus j_n)$

Simon's problem: Let $F : \{0,1\}^n \to \{0,1\}^n$ s.t.
there exists $s \in \{0,1\}^n$

$$F(x) = F(y) \iff x = y \text{ or } x \oplus s = y \quad \forall x, y \in \{0,1\}^n.$$

Goal: Find s.

Quantum Algorithm

1) Start with $|0^n\rangle|0^n\rangle$ and apply $\boxed{H}$ to the first n qbits to obtain

$$\frac{1}{\sqrt{N}} \sum_{i=0}^{2^n-1} |i\rangle|0^n\rangle$$

2) Apply the query $|i\rangle \otimes |b^n\rangle \mapsto |i\rangle \otimes (|b^n\rangle \oplus F(|i\rangle))$ with $|b^n\rangle = |0^n\rangle$ to obtain

$$\frac{1}{\sqrt{N}} \sum_{i=0}^{2^n-1} |i\rangle|F(|i\rangle)\rangle$$

3) Measure the second n qbits in computational basis, i.e. $\mathrm{span}\{|i\rangle, i=0,\ldots,2^{n}-1\} \otimes \mathrm{span}\{|j\rangle\}$ to
$j = 0, \ldots, 2^n-1$

obtain some output

$$|\alpha\rangle \otimes |j\rangle = \frac{P_j\left(\frac{1}{\sqrt{N}}\sum |i\rangle F(|i\rangle)\right)}{\| P_j(\cdots)\|}$$

Note that

$$|i\rangle F(|i\rangle) \perp \operatorname{ran} P_j \quad \text{for} \quad F(|i\rangle) \neq |j\rangle$$

by assumption, there exist exactly two inputs $|i_0\rangle$ and $|i_1\rangle = |i_0 \oplus s\rangle$ with $F(|i_k\rangle) = |j\rangle \quad k = 0,1$.

Hence

$$|\alpha\rangle = \frac{1}{\sqrt{2}}\left(|i_0\rangle + |i_0 \oplus s\rangle\right)$$

4) Ignore second n qbits and apply $\boxed{H}$ to the first n to obtain

$$\frac{1}{\sqrt{2N}} \sum_{j=0}^{2^n-1} \left[(-1)^{i\cdot j} + (-1)^{(i\oplus s)\cdot j}\right] |j\rangle$$

5) Measure in computational basis

- $|j\rangle$ has non-zero amplitude if

$$(i_0 \oplus s)\cdot j \underset{\text{mod } 2}{=} i_0 \cdot j \iff s\cdot j = 0 \mod 2$$

$\Rightarrow$ We obtain random element of the set

$$\{ j \mid s\cdot j = 0 \mod 2 \}$$

6) Repeat the procedure to obtain $n-1$ linearly ind. elements $j^{(1)}, \dots, j^{(n-1)}$ with

$$j^{(i)} \cdot s = 0 \mod 2$$

or

$$\begin{pmatrix} j^{(1)} \\ \vdots \\ j^{(n-1)} \end{pmatrix} s = 0 \mod 2$$

7) Solve linear system mod 2 on classical computer in $O(n^3)$

$\boxed{\begin{array}{l} \text{Note that } \#\,\mathrm{span}\{j^{(1)}, \dots, j^{(k)}\} \le 2^k \\ \text{Hence, with probability } \frac{2^n - 2^k}{2^n} = 1 - 2^{k-n} \ge \frac{1}{2} \\ \text{for } k \le n-1, \text{ we find} \\ \text{a linearly independent vector } j_{k+1} \end{array}}$

<u>Conclusion</u>: number of qbits $O(n)$, number of gates $O(n)$, number of iterations $O(n)$ with high probability $+ O(n^3)$

Classical algorithms for Simon's problem:

Lemma 2 Any (randomized) classical algorithm with less than $\delta 2^{\frac{n}{2}}$ queries to f fails with probability $\geq e^{-\frac{5}{4}\delta^2}$. (deterministic Alg. fail certainly if less than $2^{\frac{n}{2}}$ queries).

Proof Every Alg generates a sequence of queries $x_1, \dots, x_n$ with $\underset{y_1}{f(x_1)}, \dots, \underset{y_n}{f(x_n)}$

If all y_i are distinct, the Alg can't distinguish between $s=0$ and $s \neq 0$.

Assume all $y_1 \dots y_k$ are distinct. Then the Alg chooses x_{k+1} based on some prob. measure μ on $\{0,1\}^n$ [in the det. case, μ is a delta distribution].

$$\sum_{k=1}^{c} \sum_{s \in \{0,1\}^n} \sum_{i=1}^{k} \mu(x_i \oplus s) = \sum_{k=1}^{c} \sum_{i=1}^{k} \underbrace{\sum_{s} \mu(x_i \oplus s)}_{=1}$$

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

$$\Rightarrow \exists s \in \{0,1\}^n: \quad \sum_{k=1}^{c} \underbrace{\sum_{i=1}^{k} \mu(x_i \oplus s)}_{p_i \leq \frac{1}{2}} \leq \frac{c(c+1)}{2^{n+1}} \leq \frac{1}{2}$$

for $c(c+1) \leq 2^n$

Hence,

$$\mathbb{P}(y_1 \dots y_{k+1} \text{ distinct}) = \underbrace{\mathbb{P}(y_{k+1} \notin \{y_1 \dots y_k\} | y_1 \dots y_k \text{ distinct})}_{\geq 1 - p_i} \cdot \mathbb{P}(y_1 \dots y_k \text{ distinct})$$

Iterate the argument to obtain

$$\mathbb{P}(y_1 \cdots y_c \text{ distinct}) \ge \prod_{i=1}^{c} (1-p_i)$$

Taking logarithms shows

$$\log\left(\prod_{i=1}^{c} (1-p_i)\right) = \sum_{i=1}^{c} \log(1-p_i)$$

$$\left(p_i \le \tfrac{1}{2}\right) \to \ge -\frac{5}{4} \sum_{i=1}^{c} p_i \ge -\frac{5}{4} \frac{c(c+1)}{2^{n+1}}$$

$$\Rightarrow \mathbb{P}(y_1 \dots y_c \text{ distinct}) \ge e^{-\frac{5}{4}\frac{c(c+1)}{2^{n+1}}}$$

Choosing $c+1 = \delta 2^{\frac{n}{2}}$ satisfies $c(c+1) \le 2^n$

and $\mathbb{P}(y_1, \dots, y_c \text{ distinct}) \ge \underline{e^{-\frac{5}{4}\delta^2}}$

□

$$\log(1-p) = 0 + \frac{1}{1}(-p) - \frac{1}{2\xi^2}(-p)^2 \quad \text{for some } 1-p \le \xi \le p$$

$$\ge -p - \frac{p^2}{2} = -p\left(1 + \frac{p}{2}\right)$$

$$\underline{\ge -\frac{5}{4}p}$$

The quantum Fourier Transform

Disrete FT: for $x_0, \dots, x_{N-1}$ define

$$\hat{x}_k := \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j \, e^{-\frac{2\pi i}{N} kj}$$

Inverse

$$x_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \hat{x}_j \, e^{\frac{2\pi i}{N} kj}$$

in matrix notation

$$\hat{x} = F_N x \quad \text{with} \quad F_N \in \mathbb{C}^{N \times N}$$

$$(F_N)_{kj} := \frac{1}{\sqrt{N}} e^{-\frac{2\pi i}{N} kj}$$

F_N is unitary matrix ✓

<u>Fast FT</u>: Assume $N = 2^n$

$$\hat{x}_k = \frac{1}{\sqrt{2}} \Bigg(\frac{1}{\sqrt{N/2}} \sum_{\substack{j=0 \\ j \text{ even}}}^{N-1} x_j \, e^{-\frac{2\pi i}{N/2} k \frac{j}{2}} + e^{-\frac{2\pi i}{N} k} \frac{1}{\sqrt{N/2}} \sum_{j \text{ odd}} e^{-\frac{2\pi i}{N/2} k \frac{j-1}{2}} \Bigg)$$

$\Rightarrow$ split FT into 2 FTs of half size

leads to $O(N \log N)$ complexity.

The QFT maps a state

$$|x\rangle = \sum_{j=0}^{N-1} \hat{x}_i |j\rangle \quad \text{to} \quad |y\rangle = \sum_{j=0}^{N-1} x_j |j\rangle$$

where x_j is given by the classical IFT

This is a convention. Everything works with FT

If $|x\rangle = |j_0\rangle$, then $x_j = \delta_{jj_0}$ and hence

$$\hat{x}_k = \frac{1}{\sqrt{N}} e^{\frac{2\pi i}{N} k j_0}$$

$$\Rightarrow \quad |j\rangle \xmapsto{\text{QFT}} \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{\frac{2\pi i}{N} kj} |k\rangle$$

$$= F_N |j\rangle$$

To implement F_N efficiently, we observe

$$\frac{k}{N} = \frac{k}{2^n} = \sum_{\ell=1}^{n} k_\ell 2^{-\ell} \quad \text{if } k = k_1 2^{n-1} + k_2 \cdot 2^{n-2} + \dots + k_n$$

$$\Downarrow$$

$$F_N |j\rangle = \frac{1}{\sqrt{2^n}} \sum_{k \in \{0,1\}^n} \underbrace{e^{2\pi i j \sum_{\ell=1}^{n} k_\ell 2^{-\ell}} |k_1 \dots k_n\rangle}_{\bigotimes_{\ell=1}^{n} e^{-2\pi i j k_\ell 2^{-\ell}} |k_\ell\rangle}$$

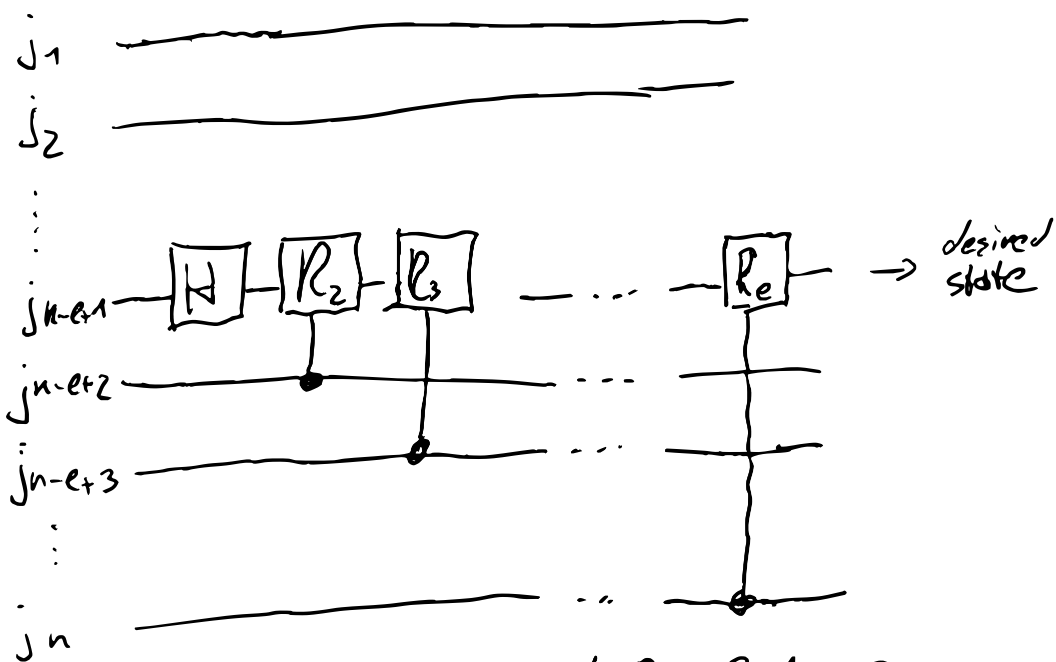
$$= \bigotimes_{\ell=1}^{n} \left(|0\rangle + e^{2\pi i j / 2^\ell} |1\rangle \right) \frac{1}{\sqrt{2}}$$

Furthermore note that

$$e^{2\pi i j/2^\ell} = \underbrace{e^{2\pi i \overbrace{\sum_{m=1}^{n-\ell} j_m 2^{n-m-\ell}}^{\in \mathbb{N}}}}_{=1} \; e^{2\pi i \sum_{m=n-\ell+1}^{n} j_m 2^{n-m-\ell}}$$

$\Rightarrow$ the first $n-\ell$ significant bits of j do not matter.

To implement $\frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i \sum_{m=n-\ell+1}^{n} j_m 2^{n-m-\ell}} \cdot |1\rangle\right)$ we use the $\boxed{R_s}$ gate or $\boxed{P}$ in IBM q-Composer given by the matrix $\begin{pmatrix} 1 & 0 \\ 0 & e^{\frac{2\pi i}{2^s}} \end{pmatrix}$



need to repeat that for $\ell = 1, \dots, n$

Cost of QFT:

we need n qbits and $O(n)$ gates per qbit

$\Rightarrow O(n^2)$ gates

$\Rightarrow$ exponential speedup over FFT with $O(n 2^n)$ operations.

<u>Remark</u> Strictly speaking, QFT does something different than FT. The state

$$QFT(|x\rangle) = \sum_{j=0}^{N-1} x_j |j\rangle$$

can only be assessed via measurement and hence will collapse to some $|j'\rangle$ with a certain probability. We will see that this is still very useful.

<u>Remark</u> R_s gates don't do very much for large s. One can show that $O(n \log n)$ gates suffice if one accepts a small error probability

<u>Remark</u> reversing the order of the gates and using the adjoint gates gives an efficient implementation of the inverse QFT F_N^{-1}

Application Phase estimation

Suppose we have unitary operator $U: V \to V$, $\dim V = 2^n$
with eigenvector ψ

$$\Rightarrow \quad U\psi = \underbrace{e^{2\pi i \phi}}_{\lambda} \psi \quad \text{for some } \phi \in [0,1)$$

$$\left(|U\psi|^2 = \lambda^2 |\psi|^2 = |\psi|^2 \Rightarrow |\lambda| = 1 \right)$$

Assume that $\phi = \sum_{j=1}^{n} \phi_j 2^{-j}$ can be written with n bits

Since U is operator on 2^n-dim Hilbertspace classical computation of $U\psi \cdot \psi$ costs at least $O(2^n)$

Quantum algorithm

1) Start with $|0^n\rangle |\psi\rangle$
2) Apply $H^{\otimes n}$ to the first n qbits to obtain

$$\frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} |j\rangle |\psi\rangle \qquad \text{(Applying } QFT_N \text{ would do the same)}$$

3) Apply $|j\rangle|\psi\rangle \mapsto |j\rangle U^j |\psi\rangle$ to obtain

$$\frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} e^{2\pi i j \phi} |j\rangle |\psi\rangle$$

Note that the first n qbits satisfy

$$\frac{1}{\sqrt{N}}\sum_{j=0}^{N-1} e^{2\pi i j \phi} |j\rangle = \frac{1}{\sqrt{N}}\sum e^{2\pi i j \frac{N\phi}{N}} |j\rangle = F_N(|N\phi\rangle)$$

→ This might be a bit confusing since suddenly $\phi \in \mathbb{R}$ is interpreted as an element of the vector space in which ψ is contained

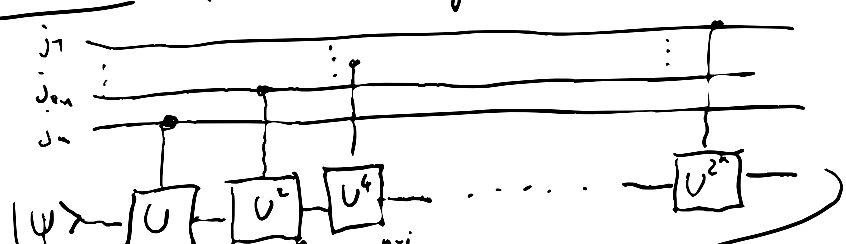
But $N\phi = \sum_{j=1}^{n} \phi_j 2^{n-j}$ is a basis element and hence this makes sense

4) Apply IQFT to the first n qbits to obtain

$$F_N^{-1}\left(\frac{1}{\sqrt{N}}\sum e^{2\pi i j \phi} |j\rangle\right) = |N\phi\rangle$$

5) Measure in computational basis to obtain $N\phi$ and hence ϕ

<u>Remark</u> How to implement step 3?



results in $U^{\sum_{i=1}^{n} j_i 2^{n-i}} |\psi\rangle$

But we will need to assume that $\boxed{U^{2^k}}$ can be implemented effectively (This might not be true in general, but is in the applications below)

Note the the input doesn't need to be a single eigenvector. Let $|\psi\rangle, |\psi'\rangle$ denote two EVs with phase ϕ, ϕ'. Linearity implies that input $\frac{1}{\sqrt{2}}(|\psi\rangle + |\psi'\rangle)$ produces output

$$\frac{1}{\sqrt{2}}(|N\phi\rangle + |N\phi'\rangle)$$

A measurement will produce ϕ, ϕ' will equal probability

---

if ϕ requires more than n bits, the final state is a small perturbation δ of $F_N(|Nk\rangle)$ with

$$|\delta| = \left| \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \underbrace{\left[e^{2\pi i j \phi} - e^{2\pi i j \phi_n} \right]}_{O(2^{-n})} |j\rangle \right| \Big/ \left| \frac{1}{\sqrt{N}} \sum e^{2\pi i j \phi} |j\rangle \right|$$

$$\simeq O(2^{-n})$$

Shor's integer factorization

Factoring problem: given $n \in \mathbb{N} \setminus \{\text{primes}\}$ find $1 < k < n$ with $\frac{n}{k} \in \mathbb{N}$.

- $n \in \mathbb{N}$ can be defined using $\log_2(n)$ bits $\Rightarrow$ Poly. complexity of algs means $\mathcal{O}(\log_2(n)^p)$ operations
- There exist efficient classical algorithms to check whether n is prime or power of prime (see, e.g. AKS-test or BPSW-test)
- Security of many crypto-systems is based on the fact that integer factorization is hard
- Best known classical algorithm is the general number field sieve with runtime $\mathcal{O}\left(e^{(\log_2 n)^{\frac{1}{3}} (\log_2 \log_2 n)^{\frac{2}{3}}}\right)$

Note that the runtime is a conjecture!

- It is not known that classical algs can not be faster. Latest paper which (falsely) claimed an $O(\log n^p)$-alg for factorization was by Schnorr (Bonn) in 2021

Reduction to period finding

Want to factor $N \in \mathbb{N}$.

1) Choose random $x \in \{2, \dots, N-1\}$ with $\gcd(x, N) = 1$

$\Rightarrow x \in (\mathbb{Z}/N\mathbb{Z})^\times$ multiplicative group mod n

(Lemma 3)

$\Rightarrow x$ has period r with $x^r \bmod N = 1$

$\Rightarrow$ with prob. $\frac{1}{2}$ r is even and

$x^{\frac{r}{2}} \pm 1 \not\equiv 0 \mod N$

$\Rightarrow (x^{\frac{r}{2}} - 1)(x^{\frac{r}{2}} + 1) \cong x^r - 1 \equiv 0 \mod N$

$\Rightarrow \gcd(x^{\frac{r}{2}} - 1, N), \gcd(x^{\frac{r}{2}} + 1, N)$ are non-trivial factors of N

Lemma Every $x \in (\mathbb{Z}/N\mathbb{Z})^\times$ has period r which is minimal $r \in \mathbb{N}$ with $x^r \equiv 1 \mod N$.

Proof Consider the set $S = \{1, x^1, \dots\} \subseteq (\mathbb{Z}/N\mathbb{Z})^\times$

Since $(\mathbb{Z}/N\mathbb{Z})^\times$ is finite, there exist $j \neq k \in \mathbb{N}$ with $x^j = x^k \mod N$. W.L.o.g assume $j < k$

$\Rightarrow x^{k-j} = 1 \mod N$. $\square$

Euler totient func: $\varphi(n) := \# \{1 \le k \le n : \gcd(k,n) = 1\}$

$\varphi(n) = n \prod_{\substack{p \mid n \\ p \text{ prime}}} (1 - \frac{1}{p}) \Rightarrow \varphi(p^\alpha) = p^\alpha (1 - \frac{1}{p}) = p^{\alpha-1}(p-1)$

Lemma [Chinese remainder Thm]

Let $m_1, \dots, m_n \in \mathbb{N}$ with $\gcd(m_i, m_j) = 1$ $i \neq j$. Then

$$\left[\text{Find } x \text{ s.t. } \begin{array}{ll} x = a_1 & \mod m_1 \\ \vdots & \vdots \\ x = a_n & \mod m_n \end{array} \right]$$

is solvable and any two solutions are equal mod $M = m_1 m_2 \cdots m_n$

Proof Define $M_i := \frac{M}{m_i} \Rightarrow \gcd(M_i, m_i) = 1$

$\Rightarrow M_i$ has mult. inverse mod m_i denoted by N_i

$\Rightarrow x = \sum_i a_i M_i N_i$ is solution since

$M_i N_i \equiv 1 \mod m_i$ and $M_i N_i \equiv 0 \mod m_j$ $j \neq i$

Two solutions x, x' satisfy $x - x' \equiv 0 \mod m_i$

$\Rightarrow M = m_1 m_2 \cdots m_n$ divides $x - x'$ (since m_i are coprime) $\square$

Lemma 4 Let $p > 2$ prime. Let 2^d maximal power of 2 dividing $\varphi(p^\alpha)$. Let x denote randomly chosen element in $\left(\frac{\mathbb{Z}}{p^\alpha \mathbb{Z}}\right)^*$.

$$\Rightarrow \quad \mathbb{P}(2^d \text{ divides order of } x) = \frac{1}{2}$$

Proof: $\varphi(p^\alpha) = p^{\alpha-1}(p-1)$ is even $\Rightarrow d \geq 1$.

Let g denote generator of $\left(\frac{\mathbb{Z}}{p^\alpha \mathbb{Z}}\right)^* \Rightarrow x = g^k \bmod p^\alpha$ for some $k \in \{1, \dots, \varphi(p^\alpha)\}$. Let r denote order of x.

1) k is odd: $g^{kr} \equiv 1 \bmod p^\alpha \Rightarrow \varphi(p^\alpha)$ divides kr since $\varphi(p^\alpha)$ is minimal with $g^{\varphi(p^\alpha)} = 1$

since k is odd $\Rightarrow 2^d$ divides r

2) k is even: $g^{k\varphi(p^\alpha)/2} = 1 \bmod p^\alpha$

$\Rightarrow r$ divides $\varphi(p^\alpha)/2$ since $\underbrace{x^r = g^{kr}}_{r \text{ is minimal with}} = 1 \bmod p^\alpha$

$\Rightarrow 2^d$ does not divide r

Hence, for exactly half of $x \in \left(\frac{\mathbb{Z}}{p^\alpha \mathbb{Z}}\right)^*$ we have $x = g^k$ with k even/odd $\square$

Lemma 5 Let $N = p_1^{\alpha_1} \cdots p_m^{\alpha_m}$ prime factorization of odd N. Let x be random in $\left(\frac{\mathbb{Z}}{N\mathbb{Z}}\right)^*$ with order r mod N.

$$\Rightarrow \mathbb{P}\left(r \text{ is even and } x^{\frac{r}{2}} \not\equiv -1 \bmod N\right) \geq 1 - 2^{-m+1}$$

Proof Chinese remainder theorem $\Rightarrow$ choose x random is equivalent to choose x_j rand in $\left(\frac{\mathbb{Z}}{p_j^{\alpha_j}\mathbb{Z}}\right)^*$ with $x = x_j \bmod p_j^{\alpha_j}$ $\forall j = 1, \dots, m$

(This is because $x \longleftrightarrow (x_1, \dots, x_m)$ is one-to-one)

Let r_j be order x_j mod $p_j^{\alpha_j}$ and let 2^{d_j} max power of 2 dividing r_j. We will show $\underbrace{r \text{ is odd or } x^{\frac{r}{2}} \equiv -1 \bmod N \Rightarrow}$ $d_1 = \dots = d_m$. If this is the case, the following argument concludes the proof:

Let d_j' maximal s.t. $2^{d_j'}$ divides $\varphi(p_j^{\alpha_j})$

Lemma 4 $\Rightarrow \mathbb{P}(d_j = d_j') = \frac{1}{2} \Rightarrow \mathbb{P}(d_j = k) \leq \frac{1}{2}$ $\forall k \in \mathbb{N}$. Since d_j independent

$$\Rightarrow \mathbb{P}(d_1 = \dots = d_m) \leq 2^{-m+1}$$

- It remains to show $d_1 = \dots = d_m$.

Case 1: r is odd. $x^r = 1 \bmod N \Rightarrow x^r = 1 \bmod p_j^{\alpha_j}$

$\Rightarrow r_j$ divides $r \Rightarrow r_j$ is odd $\Rightarrow d_j = 0$.

Case 2: r is even and $x^{\frac{r}{2}} = -1 \bmod N$

$\Rightarrow x^{\frac{r}{2}} = -1 \bmod p_j^{\alpha_j}$. if r_j would divide $\frac{r}{2}$, we would have

$$\left.\begin{aligned} x^{\frac{r}{2}} &= \underbrace{x}\,^{k r_j} = -1 \bmod p_j^{\alpha_j} \\ &= x_j^{k r_j} \bmod p_j^{\alpha_j} \\ &= 1 \bmod p_j^{\alpha_j} \end{aligned}\right\} \Rightarrow r_j \text{ does not divide } \frac{r}{2}.$$

but r_j divides r. Hence largest power of 2 dividing r must be equal to d_j.

$\square$

Note that if r is period of $x \bmod N$ and r even, we have $x^{\frac{r}{2}} \neq 1 \bmod N$

$$\Rightarrow \mathbb{P}(r \text{ even } x^{\frac{r}{2}} \neq \pm 1 \bmod N) \geq 1 - 2^{-m+1} \geq \frac{1}{2}$$

if $m \geq 2$, i.e, if N is not a prime power.

Define $U|y\rangle := |xy \bmod N\rangle$

Note that $|j\rangle \longmapsto |xj \bmod N\rangle$ is bijective

since $xj = xj' \bmod N \qquad x(j-j') = 0 \bmod N$

$\Rightarrow \gcd(x,N) \neq 1$.

$\Rightarrow U$ is permutation on basis elements $\Rightarrow$ unitary

<u>Lemma</u> Let r denote the period of x in $\left(\mathbb{Z}/N\mathbb{Z}\right)^{*}$.

Then, $$|u_s\rangle := \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} \exp\left(\frac{-2\pi i s k}{r}\right) |x^k \bmod N\rangle$$

are EVs of U with eigenvalues

$$\lambda_s := \exp\left(\frac{2\pi i s}{r}\right), \quad s = 0, \dots, r-1$$

<u>Proof</u>

$$U|u_s\rangle = \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} \exp\left(\frac{-2\pi i s k}{r}\right) |x^{k+1} \bmod N\rangle$$

$$= \lambda_s \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} \exp\left(\frac{-2\pi i s (k+1)}{r}\right) |x^{k+1} \bmod N\rangle$$

$x^r = x^0 \bmod N$

$e^{-2\pi i s} = e^0$

$$= \lambda_s |u_s\rangle \qquad \square$$

Note that there holds

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

$$= \frac{1}{r} \sum_{k=0}^{r-1} \underbrace{\sum_{s=0}^{r-1} e^{-\frac{2\pi i k s}{r}}}_{\hookrightarrow \begin{cases} 0 & k \neq 0 \\ r & k=0 \end{cases}} |x^k \bmod N\rangle$$

$$= |x^0 \bmod N\rangle = |1\rangle$$

$\Rightarrow$ Quantum phase estimation for U with input $|1\rangle$ produces the state

$$\frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} |2^n \tfrac{s}{r}\rangle$$

note that $r \leq N$ hence $\frac{s}{r}$ can be exactly represented with n bits

- To officiently implement $U^{2^k}|y\rangle = |x y^{2^k} \bmod N\rangle$ we use the fact

$$y^{2^k} = \underbrace{(((y^2)^2)^{\cdot^{\cdot^{\cdot}}})^2}_{k\text{-times}}.$$ Hence, $|x y^{2^k} \bmod N\rangle$

requires 1 multiplication, k squares mod N

$\Rightarrow$ can be implemented as in classical circuits with XOR & AND gates.

Shor's period finding algorithm

Note that this is the only quantum part of Shor's alg. $N \le 2^n$

1) Prepare $|\psi\rangle = |1\rangle = |\overbrace{0\cdots01}^{n}\rangle$ and U as before

2) Use quantum phase estimation with U and $|\psi\rangle$ to obtain the state

$$\frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} |2^n \tfrac{s}{r}\rangle$$

3) Measurement gives a random number $\frac{s}{r}$ from the set $\{\frac{1}{r}, \dots, \frac{r-1}{r}\} \subseteq \{\frac{i}{2^n} : i=0,\dots,2^n-1\}$

4) Write $\frac{s}{r} = \frac{s_0}{r_0}$ with $\gcd(s_0, r_0) = 1$. If s, r were coprime already, we have $r = r_0$ and found the period. Otherwise, repeat.

Remark: ~~There~~ There are $O(\frac{r}{\log\log r})$ — Euler's totient function — numbers $1 \le s < r$ with $\gcd(s,r) = 1$.

$\Rightarrow$ Success prob of step 4 is $O(\frac{1}{\log\log r})$

Since $\frac{1}{\log\log r} \ge \frac{1}{\log\log N} \ge \frac{1}{\log n}$, step 4 requires on average $\log n$ days.

Remark

Implicitely, we used QFT to find the frequency of

$$x^k \bmod N, \text{ i.e. the period of } x$$

Grover's algorithm

Problem: given $f : \{0,1\}^n \to \{0,1\}$, find $x \in \{0,1\}^n$ with $f(x) = 1$ or determine $f = 0$.

Recall the query

$$Q_f(|i\rangle) = (-1)^{f(i)} |i\rangle$$

(again we identify $x \in \{0,1\}^n$ with $i = 0, \dots, 2^{n-1}$)

Define the Grover diffusion operator

$$U_s := 2 \underbrace{|s\rangle\langle s|}_{\text{projection onto } |s\rangle} - I \, ,$$

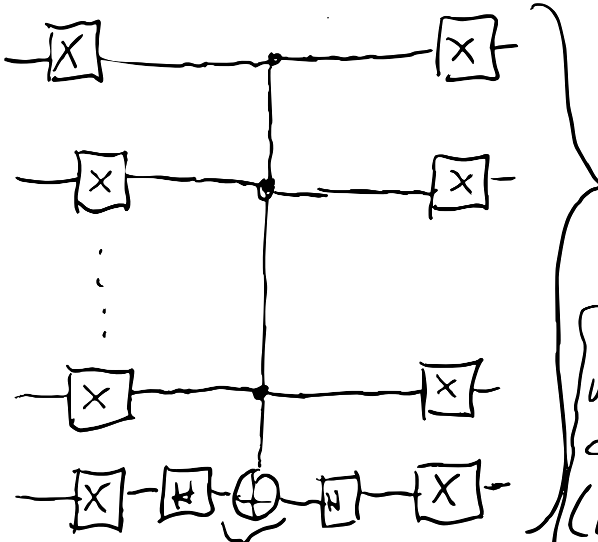
where $|s\rangle$ denotes the uniform state

$$|s\rangle = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} |j\rangle$$

Note that $|s\rangle = H^{\otimes n} |0^n\rangle$ and hence

$$U_s = H^{\otimes n} \underbrace{\left(2|0^n\rangle\langle 0^n| - I \right)}_{\text{corresponds to the matrix}} H^{\otimes n}$$

$$\begin{pmatrix} +1 & & 0 \\ & -1 & \\ & & \ddots & \\ 0 & & & -1 \end{pmatrix} \in \mathbb{R}^{2^n \times 2^n}$$



$$-\left(2|0\rangle\langle 0| - I\right)$$

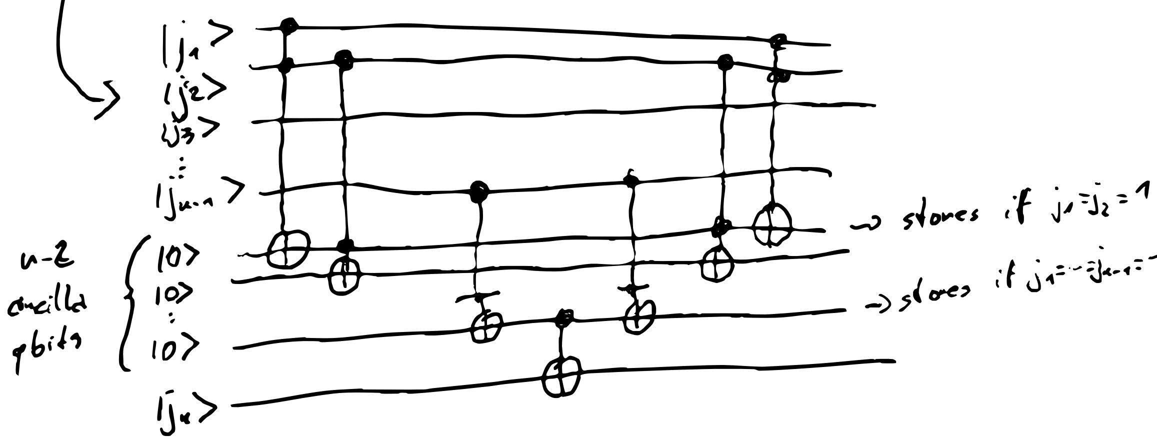
since we can only measure amplitudes, a global phase change (multiplication with $\lambda \in \mathbb{C}$, $|\lambda| = 1$) is not noticable

alternatively, one can apply $\boxed{Y}-\boxed{X}-\boxed{Z}-\boxed{X}$ to obtain

$$\begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix}$$

$$\begin{pmatrix} 1 & & & \\ & \ddots & & \\ & & 1 & \\ & & & \begin{matrix} 0 & 1 \\ 1 & 0 \end{matrix} \end{pmatrix}$$

generalized Toffoli gate



Algorithm

1) Apply $H^{\otimes n}$ to obtain $|s\rangle = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} |j\rangle$

2) For $k = 1, \dots, r(N)$ do

2a) Apply Q_F to current state $\sum_{j=0}^{N-1} \alpha_j |j\rangle$ to obtain $\sum_{j=0}^{N-1} (-1)^{F(j)} \alpha_j |j-\rangle$

2b) Apply U_s to first n qbits

3) Measure in computational basis

Thm: With $r(N) = \frac{\arccos(\sqrt{\frac{t}{N}})}{2\arcsin(\sqrt{\frac{t}{N}})}$, Grover's alg. finds state $|x\rangle$ with $F(x)=1$ with probability $\geq 1 - \frac{t}{N}$

Proof:

Define $|B\rangle := \frac{1}{\sqrt{N-t}} \sum_{F(|j\rangle)=0} |j\rangle$, where $t = \# F^{-1}(\{|1\rangle\})$

Note that the first n qbits of $|x-\rangle \mapsto Q_F(|x-\rangle)$ satisfy

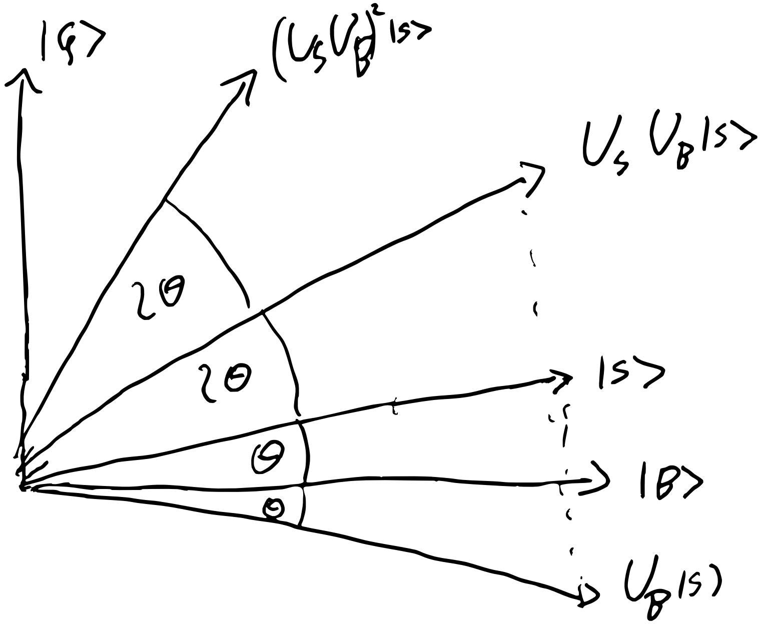
$$|x\rangle \mapsto (2|B\rangle\langle B| - I)|x\rangle = U_B(|x\rangle)$$

(Check for basis elements $|x\rangle = |j\rangle$)

Define $|G\rangle := \frac{1}{\sqrt{t}} \sum_{F(|j\rangle)=1} |j\rangle$ and note

$|s\rangle \in \mathrm{span}\{|G\rangle, |B\rangle\}$; $U_s, U_B : \mathrm{span}\{|G\rangle, |B\rangle\} \circlearrowleft$

hence, the Grover iteration never leaves the plane $\mathrm{span}\{|G\rangle, |B\rangle\}$



Each application of $U_s U_B$ rotates a state $|x\rangle \in \operatorname{span}\{|B\rangle, |\varphi\rangle\}$ towards $|\varphi\rangle$ by an angle θ given by

$$\cos\theta = \frac{\langle s|B\rangle}{|s||B|} = \frac{1}{\sqrt{N}} \frac{1}{\sqrt{N-t}} \sum_{F(|j\rangle)=0} \langle j|j\rangle = \frac{N-t}{\sqrt{N(N-t)}}$$

$$= \sqrt{1-\frac{t}{N}}$$

$$\Rightarrow \sin^2\theta = 1-\cos^2\theta = 1-1+\frac{t}{N} \Rightarrow \sin\theta = \sqrt{\frac{t}{N}}$$

$$\Rightarrow \theta \approx \sqrt{\frac{t}{N}}$$

for large N.

Since the angle between $|\varphi\rangle$ and $|s\rangle$ is

$$\cos\alpha = \frac{1}{\sqrt{N}} \frac{1}{\sqrt{t}} \sum_{F(|j\rangle)=1} \langle j|j\rangle = \frac{\sqrt{t}}{\sqrt{N}} \Rightarrow r(N) = \operatorname{round}\left(\frac{\arccos\left(\frac{\sqrt{t}}{\sqrt{N}}\right)}{2\arcsin\frac{\sqrt{t}}{\sqrt{N}}}\right)$$

to obtain a state $|x\rangle$ with $\langle g|x\rangle \geq \cos(\theta)$

$$\Rightarrow |\langle g|x\rangle|^2 \geq \cos\theta^2 = 1-\frac{t}{N}$$

$\square$

<u>Remark</u> in practise, we don't know when $|x\rangle$ gets close to $|g\rangle$. But we can measure and check whether $F(x)=1$. If not, restart the algorithm. If $t \ll N$

$\Rightarrow$

$$r(N) = \frac{\arccos\sqrt{\frac{t}{N}}}{2\arcsin\sqrt{\frac{t}{N}}} \approx \frac{\pi}{4}\sqrt{\frac{N}{t}}$$

<u>Remark</u> If $t=1$, a classical algorithm requires at least N evaluations of F. Grovers algorithm only needs $O(\sqrt{N})$ iterations with $O(n)$ gates

Optimality of Grover's Alg

Lemma Any Quantum Alg based on the Query Q_F requires at least $O(\delta\sqrt{N})$ applications of Q_F to succeed with prob. $\geq \delta^2$.

Proof Any Q-Alg starts with some state $|\varphi\rangle$ and applies unitary transformations as well as Q_F. I.e. we may write the state after k applications of Q_F as

$$|\psi_k^F\rangle := \underbrace{U_k \, Q_F \, U_{k-1} \, Q_F \cdots U_1}_{\text{unitary}} \, Q_F \, |\varphi\rangle$$

Additionally, we will consider

$$|\psi_k\rangle := U_k U_{k-1} \cdots U_1 |\varphi\rangle$$

Let $F_j : \{0,1\}^n \to \{0,1\}$ with $F_j(|i\rangle) = \delta_{ij}$

and define

$$D_k = \sum_{j=0}^{N-1} \| \psi_k^{F_j} - \psi_k \|^2$$

Idea: If D_k is small, the evaluation of F doesn't make a big difference and it will be hard to find $F(|i\rangle)=1$.

Step 1: Show that $D_k \leq 4k^2$ by induction.

$k=0$: $D_k = 0$ ✓

$k \mapsto k+1$:
$$D_{k+1} = \sum_{j=0}^{N-1} \| U_{k+1} Q_{F_j} \psi_k^{F_j} - U_{k+1} \psi_k \|^2$$
$$= \sum_{j=0}^{N-1} \| Q_{F_j} \psi_k^{F_j} - \psi_k \|^2$$
$$= \sum \| Q_{F_j} (\psi_k^{F_j} - \psi_k) + (Q_{F_j} - I) \psi_k \|^2$$

Note that $Q_{F_j} = I - 2|j\rangle\langle j|$ $\Rightarrow (Q_{F_j} - I)|\psi_k\rangle$
$$= -2|j\rangle(\langle j|\psi_k\rangle)$$

$\Rightarrow$
$$D_{k+1} \leq \sum_{j=0}^{N-1} \|\psi_k^{F_j} - \psi_k\|^2 + 4\|\psi_k^{F_j} - \psi_k\| |\langle j|\psi_k\rangle| + 4|\langle j|\psi_k\rangle|^2$$

Cauchy-Schwartz shows

$$D_{k+1} \leq D_k + 4\left(\sum_j \|\psi_k^{F_j} - \psi_k\|^2\right)^{\frac{1}{2}} \underbrace{\left(\sum_j |\langle j|\psi_k\rangle|^2\right)^{\frac{1}{2}}}_{=1}$$

$$+ 4 \underbrace{\langle\psi_k|\psi_k\rangle}_{=1}^2$$

$$\leq D_k + 4\sqrt{D_k} + 4$$

Induction Hyp. $\rightarrow$ $\leq 4k^2 + 8k + 4 = \underline{4(k+1)^2}$ ✓

This concludes the induction and shows

$$\underline{D_k \leq 4k^2}$$

<u>Step 2:</u> Assume $|\langle j|\psi_k^{F_j}\rangle|^2 \geq \delta^2 > 0 \ \forall j = 0, \dots, N-1$, i.e. the Alg. works with high prob. for each input.

Replacing $|j\rangle$ with $e^{i\theta_j}|j\rangle$ does not change success prob. $\Rightarrow$ we may assume $|\langle j, \psi_k^{F_j}\rangle| = \langle j, \psi_k^{F_j}\rangle$.

$$\Rightarrow \|\psi_k^{F_j} - j\|^2 = 2 - 2|\langle j, \psi_k^{F_j}\rangle| \leq \underline{2(1-\delta)}$$

Define $E_k := \sum_{j=0}^{N-1} \|\psi_k^{F_j} - j\|^2 \Rightarrow E_k \leq 2N(1-\delta)$

$$F_k := \sum_{j=0}^{N-1} \|j - \psi_k\|^2$$

$\Rightarrow$

$$D_k = \sum_{j=0}^{N-1} \| (\psi_k^{T_j} - j) + (j - \psi_k) \|^2$$

$$\geq \sum_j \| \psi_k^{T_j} - j \|^2 - 2 \| \psi_k^{T_j} - j \| \| j - \psi_k \| - \| j - \psi_k \|^2$$

$$= E_k + F_k - 2 \left(\sum_j \| \psi_k^{T_j} - j \|^2 \right)^{\frac{1}{2}} \left(\sum_j \| j - \psi_k \|^2 \right)^{\frac{1}{2}}$$

$$= E_k + F_k - 2 \sqrt{E_k F_k} = \underline{\left(\sqrt{E_k} - \sqrt{F_k} \right)^2}$$

Note that any state $|\phi\rangle$ satisfies

$$\sum_{j=0}^{N-1} \| \phi - j \|^2 = \sum_j \underbrace{\| \phi \|^2}_{=1} - 2 \langle \phi | j \rangle + \underbrace{\| j \|^2}_{=1}$$

$$\geq 2N - 2 \sqrt{\sum_j 1} \underbrace{\sqrt{\sum_j |\langle \phi | j \rangle|^2}}_{=1}$$

$$= \underline{2(N - \sqrt{N})}$$

This implies $F_k \geq 2(N - \sqrt{N})$ and hence

$F_k \geq E_k$ for sufficiently large N and

$$\sqrt{F_k} - \sqrt{E_k} \geq \sqrt{2(N-\sqrt{N})} - \sqrt{2(1-\delta)N} = \frac{2\delta N - 2\sqrt{N}}{\sqrt{2(N-\sqrt{N})} + \sqrt{2(1-\delta)N}}$$

$$\geq \frac{\cancel{2}\delta N - \cancel{2}\sqrt{N}}{\cancel{2}\sqrt{2N}} = \frac{\delta}{\sqrt{2}}\sqrt{N} - \frac{1}{\sqrt{2}}$$

$$\Rightarrow D_k \geq \left(\sqrt{F_k} - \sqrt{E_k}\right)^2 \simeq \delta^2 N$$

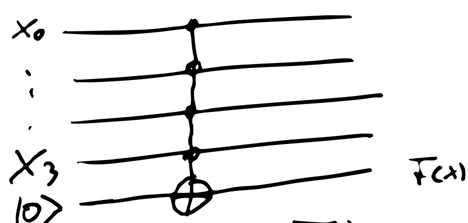
Since $D_k \leq 4k^2 \Rightarrow k \simeq \delta\sqrt{N}$ $\square$

Example in Quantum Composer

Link on webpage, $N = 16$, $n = 4$

$$F(x) = F(x_0, \dots, x_3) = x_0 \text{ AND } x_1 \text{ AND } x_2 \text{ AND } x_3$$
$$= \delta_{x=(1,1,1,1)} \Rightarrow t = 1$$

$F \Longleftrightarrow$



There holds: $\dfrac{\arccos\left(\sqrt{\frac{1}{16}}\right)}{2\arcsin\left(\sqrt{\frac{1}{16}}\right)} \simeq 2.6083$

$\Rightarrow$ optimal $r(N) = 3$.

Probability of success $\geq 1 - \frac{1}{N} = \frac{15}{16} \simeq 0.9375$

Numerical Quadrature:

Problem: Given $f : \{0,1\}^n \to [-1,1]$, compute $\frac{1}{N}\sum_{i=0}^{N-1} f(i)$

Quantum Super Sampling

Assumption: Oracle $Q_f : |0\rangle \otimes |i\rangle \mapsto \left[\sqrt{1-f(i)}\,|0\rangle + \sqrt{f(i)}\,|1\rangle\right] \otimes |i\rangle$

1) Start with $|0^p\rangle \otimes |0\rangle \otimes |0^n\rangle$
2) Apply $H^{\otimes p} \otimes I \otimes H^{\otimes n} \to$

$$\frac{1}{\sqrt{PN}} \sum_{i=0}^{N-1} \sum_{m=0}^{P-1} |m\rangle \otimes |0\rangle \otimes |i\rangle$$

3) Apply $Q_f \to$

$$\frac{1}{\sqrt{PN}} \sum_i \sum_m |m\rangle \otimes \left[\sqrt{1-f(i)}\,|0\rangle + \sqrt{f(i)}\,|1\rangle\right] \otimes |i\rangle$$

4) Define $|G\rangle := \sqrt{\frac{\sum f(i)}{N}} \sum_i \sqrt{f(i)}\,|1\rangle \otimes |i\rangle$

$$|B\rangle := \sqrt{\frac{\sum 1-f(i)}{N}} \sum_i \sqrt{1-f(i)}\,|0\rangle \otimes |i\rangle$$

$\bar{f} := \frac{1}{N} \sum f(i)$

Note

$$U_s := \left(2|0\rangle\langle 0| - I\right) \otimes I$$

$$U_s|B\rangle = |B\rangle$$

$$U_s|G\rangle = -|G\rangle$$

$$U_\varphi = \left(2|\psi\rangle\langle\psi| - I\right)$$

with $|\psi\rangle := \sqrt{1-\bar{p}}\,|B\rangle + \sqrt{\bar{p}}\,|G\rangle$

Steps 1 - 3 provide quantum circuit to implement the map $U : |0\rangle \otimes |0^n\rangle \longmapsto |\psi\rangle$

$$\Rightarrow U_\psi = U^{-1}\left(2|0^{n+1}\rangle\langle 0^{n+1}| - I\right) U$$

as in Grovers Algorithm.

5) State reads

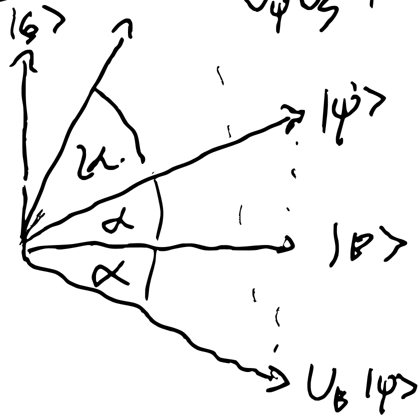
$$\frac{1}{\sqrt{P}}\sum_{m=0}^{P-1}|m\rangle\otimes\left[\sqrt{1-\bar{f}}\,|B\rangle+\sqrt{\bar{f}}\,|g\rangle\right]$$

for $\bar{f} := \frac{1}{N}\sum f(i)$

6) Apply Grover iteration $U_\psi U_s$ m-times to second $n+1$ qbits to obtain

$$\frac{1}{\sqrt{P}}\sum_{m=0}^{P-1}|m\rangle\otimes\left(U_\psi U_g\right)^m\left[\sqrt{1-\bar{f}}\,|B\rangle+\sqrt{\bar{f}}\,|g\rangle\right]$$

<u>Explanation:</u> $U_\psi U_s|\psi\rangle$ Let $\sin\theta=\sqrt{\bar{f}}$



$\Rightarrow |\psi\rangle = \cos\theta\,|B\rangle + \sin\theta\,|g\rangle$

$$\left(U_\psi U_g\right)^m|\psi\rangle = \cos\left((2m+1)\theta\right)|B\rangle + \sin\left((2m+1)\theta\right)|g\rangle$$

$\boxed{U_s \text{ acts like reflection across } |B\rangle}$

<u>Note:</u> m-times application of $U_\psi U_B$ as in "phase estimation" - blp. But: Cost in general $O(Pm)$

7) $|\psi\rangle$ rotates with rate 2θ in $|B\rangle - |G\rangle$ plane. Use QFT to obtain rate.

Measure last $n+1$ qbits in $\{|G\rangle, |B\rangle\}$-basis

$$\frac{1}{\sqrt{P}} \sum_{m=0}^{P-1} |m\rangle \otimes \left[\cos(2m+1)\theta \, |B\rangle + \sin(2m+1)\theta \, |G\rangle \right]$$

to obtain

$$\frac{1}{C} \sum_{m=0}^{P-1} \underbrace{\sin\left[(2m+1)\theta\right]}_{\hat{x}_m} |m\rangle \quad \begin{pmatrix} \text{or} \\ \cos(2m+1)\theta \end{pmatrix}$$

where $C = \frac{1}{P}\sum_{m=0}^{P-1} \sin\left[(2m+1)\theta\right]^2$.

$\rightarrow$ Apply QFT to obtain $\left[\text{assume } \theta = \frac{\pi \theta_0}{P}, \theta_0 \in \mathbb{N}\right]$

$$\frac{1}{C} \sum_{m=0}^{P-1} x_m |m\rangle \quad , \text{ where } x_m \text{ is given}$$

by IFT of $\hat{x}_m$, i.e.

$$x_m = \frac{1}{\sqrt{P}} \sum_{k=0}^{P-1} \hat{x}_k \, e^{\frac{2k\pi i m}{P}}$$

$$= \frac{1}{\sqrt{P}} \sum_{k=0}^{P-1} \frac{1}{2i} \left(e^{i(2k+1)\frac{\theta_0}{P}} - e^{-i(2k+1)\frac{\theta_0}{P}} \right) e^{\frac{2k\pi i m}{P}}$$

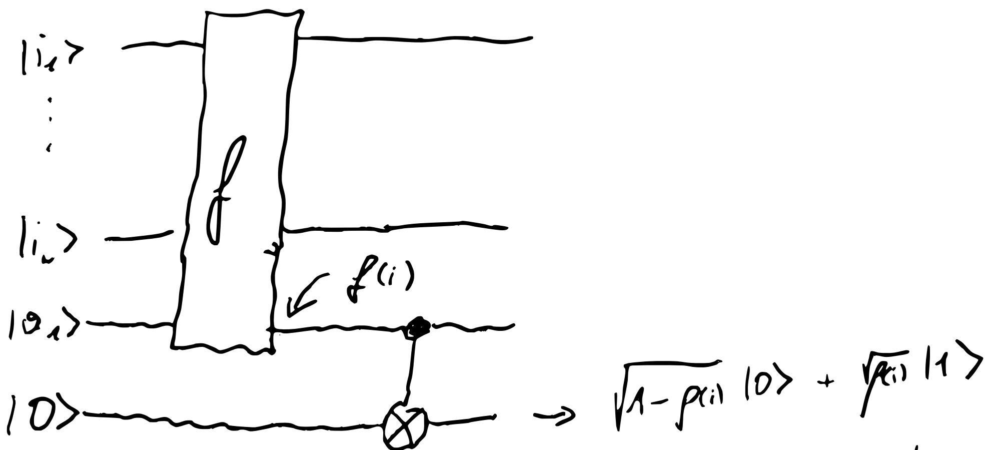
$$= e^{i\frac{\theta_0}{P}} \frac{1}{2i} \begin{cases} \sqrt{P} & m = \theta_0 \\ 0 & \text{else} \end{cases}$$

v

8) Final measurement produces $|m\rangle = |\theta_0\rangle$ and hence $\bar{f} = \sin\left(\frac{\pi \theta_0}{p}\right)^2$

<u>Remark:</u> implementation of Q_f unclear.

If $f: \{0,1\}^n \to \{0,1\}$, we can use controlled Not-gate to implement



General integrals can always be split into

$$\frac{1}{N}\sum_{i=0}^{N-1} f(i) = \frac{1}{N}\sum_{i=0}^{N-1}\sum_{k=1}^{R} 2^{-k} \underbrace{f_k(i)}_{\in\{0,1\}}$$

, i.e., R integration problems for precision 2^{-R}.