Lecture 5
Grover's algorithm, amplitude amplification and applications to cryptography

February 12, 2020

Plan

1. Grover's algorithm
2. A generalization : amplitude amplification and application to collision finding
3. Lower bound on the query complexity

1. Grover's algorithm

- Allows a quadratic speedup for searching in an unstructured data structure
- Does not provide an exponential speedup unlike Shor's algorithm but is more widely applicable

The problem

Problem 1.
Input: *A boolean function* $f : \{0,1\}^n \to \{0,1\}$ *given as a "black box"*
Output: *an* $\mathbf{x} \in \{0,1\}^n$ *such that* $f(\mathbf{x}) = 1$.

- Can be viewed as a modeling of a data search in an unstructured database of size $N = 2^n$
- Classically a randomized algorithm would need $\Theta(N)$ queries if there are $0(1)$ elements $\mathbf{x}$ such that $f(\mathbf{x}) = 1$
- Grover can solve this problem with only $O(\sqrt{N})$ queries to f and $O(\sqrt{N}\log N)$ other gates
- This query complexity can be shown to be optimal

The algorithm

Start by applying $\mathbf{H}^{\otimes n}$ and then iterate $\sqrt{N}$ times the following steps

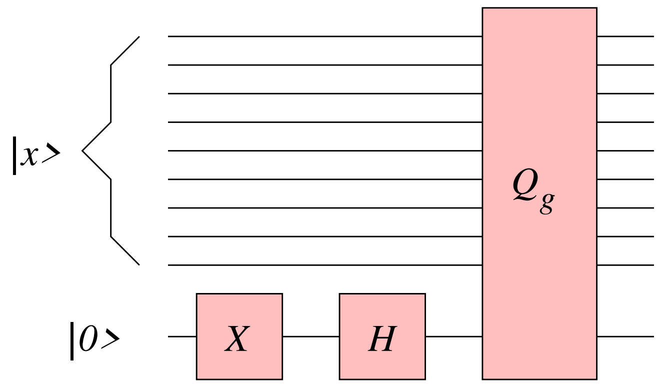
1. Perform $O_f : |x\rangle \mapsto (-1)^{f(x)} |x\rangle$
2. Perform $\mathbf{H}^{\otimes n}$
3. Perform $\mathbf{R}$ where
   - $\mathbf{R} |0\rangle = |0\rangle$
   - $\mathbf{R} |x\rangle = - |x\rangle$ for $x \neq 0$
4. Perform $\mathbf{H}^{\otimes n}$

Exercise

Give a quantum circuit of low complexity implementing $\mathbf{R}$.

Circuit for $\mathbf{R}$

Ingredient 1: from a quantum circuit Q_g performing $|x, b\rangle \mapsto |x, b \oplus g(x)\rangle$ where g is a Boolean function to a circuit performing $|x\rangle \mapsto (-1)^{g(x)} |x\rangle$:



$$|x\rangle |0\rangle \overset{\mathbf{Id}\otimes X}{\rightarrow} |x\rangle |1\rangle \overset{\mathbf{Id}\otimes H}{\rightarrow} |x\rangle \frac{|0\rangle - |1\rangle}{\sqrt{2}} \overset{Q_g}{\rightarrow} |x\rangle \frac{|g(x)\rangle - \left|\overline{g(x)}\right\rangle}{\sqrt{2}} = (-1)^{g(x)} |x\rangle \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

Ingredient 2: a quantum circuit performing

$$|x_1, \cdots, x_n\rangle |b\rangle \mapsto |x\rangle \left|b \oplus \overline{\bar{x}_1 \cdots \bar{x}_n}\right\rangle$$

Exercise

1. Let $|\psi\rangle \stackrel{\text{def}}{=} \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle$. Show that one iteration of $\mathbf{H}^{\otimes n}\mathbf{R}\mathbf{H}^{\otimes n}$ amounts to multiply the quantum state by
$$2\,|\psi\rangle\,\langle\psi| - \mathbf{Id}$$
2. Show that one iteration of $\mathbf{H}^{\otimes n}\mathbf{R}\mathbf{H}^{\otimes n}$ amounts to transform a state $\sum_x \alpha_x |x\rangle$ into
$$\sum_x (2\langle\alpha\rangle - \alpha_x)\,|x\rangle$$
where $\langle\alpha\rangle = \frac{1}{2^n}\sum_x \alpha_x$.

Grover

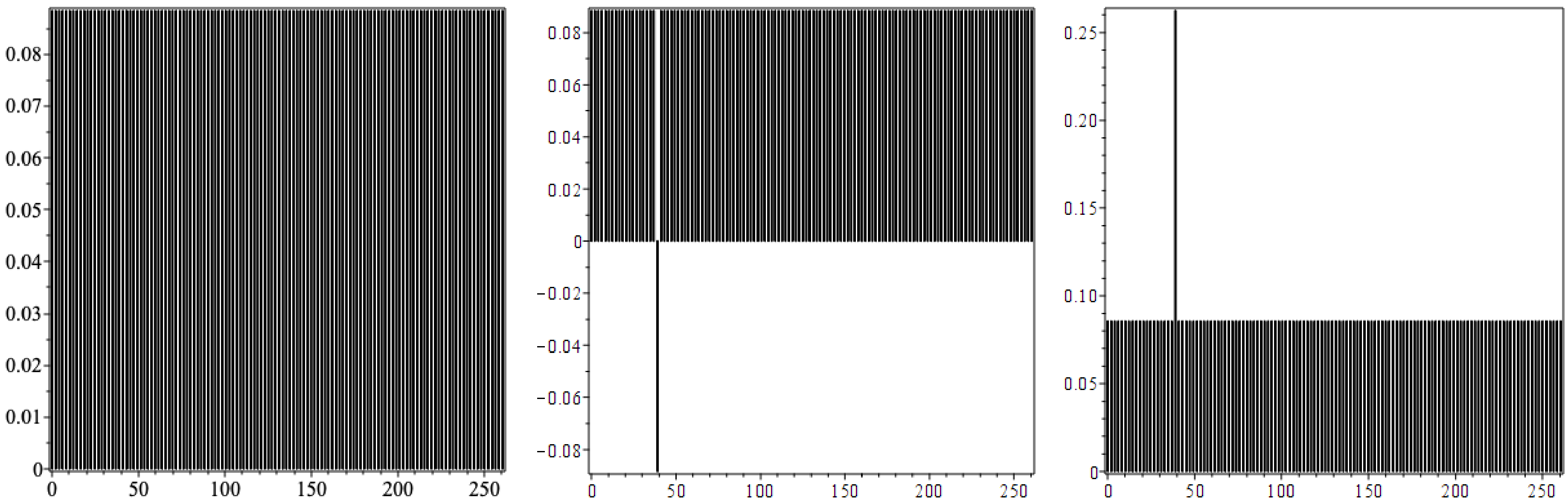
1.

$$\mathbf{R} = 2|0^n\rangle\langle 0^n| - \mathbf{Id}$$
$$\mathbf{H}^{\otimes n}\mathbf{R}\mathbf{H}^{\otimes n} = 2|\psi\rangle\langle\psi| - \mathbf{Id}$$

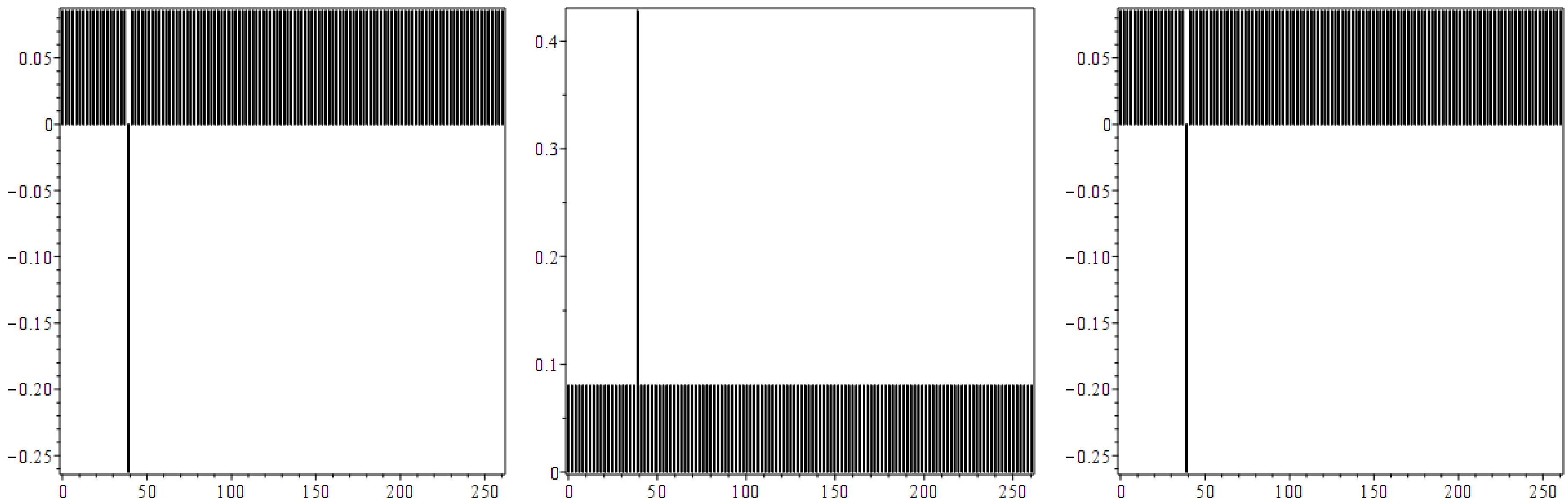
2.

$$\begin{aligned}
|\psi\rangle\langle\psi|\sum_x \alpha_x|x\rangle &= \sum_x \alpha_x|\psi\rangle\langle\psi||x\rangle \\
&= \left(\sum_x \alpha_x\langle\psi|x\rangle\right)|\psi\rangle \\
&= \left(\frac{1}{2^{n/2}}\sum_x \alpha_x\right)\frac{1}{2^{n/2}}\sum_y |y\rangle \\
&= \langle\alpha\rangle\sum_y |y\rangle
\end{aligned}$$

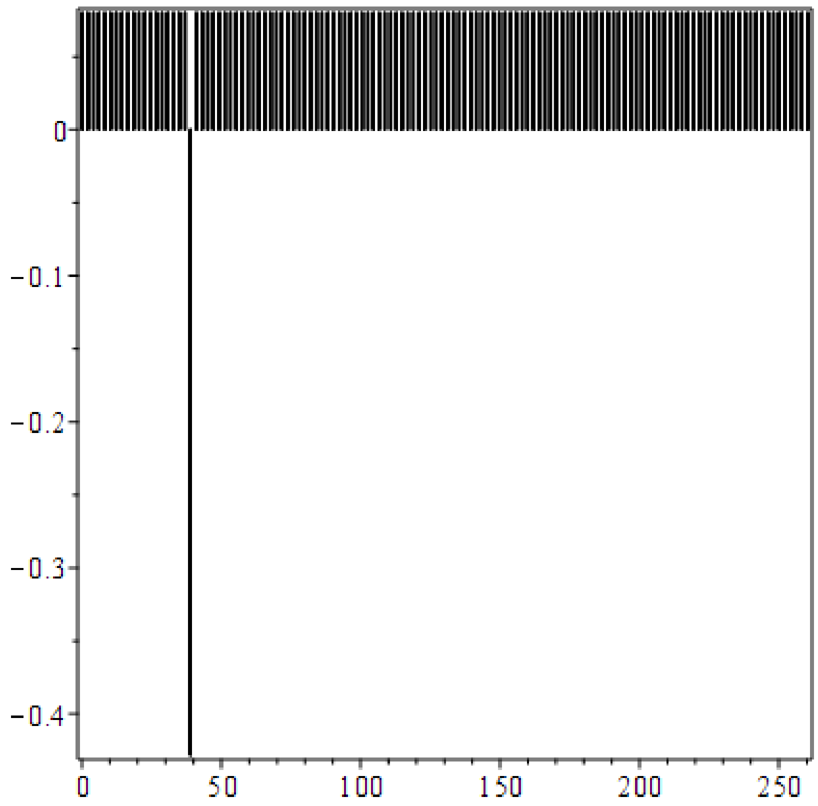
Initialisation+first step

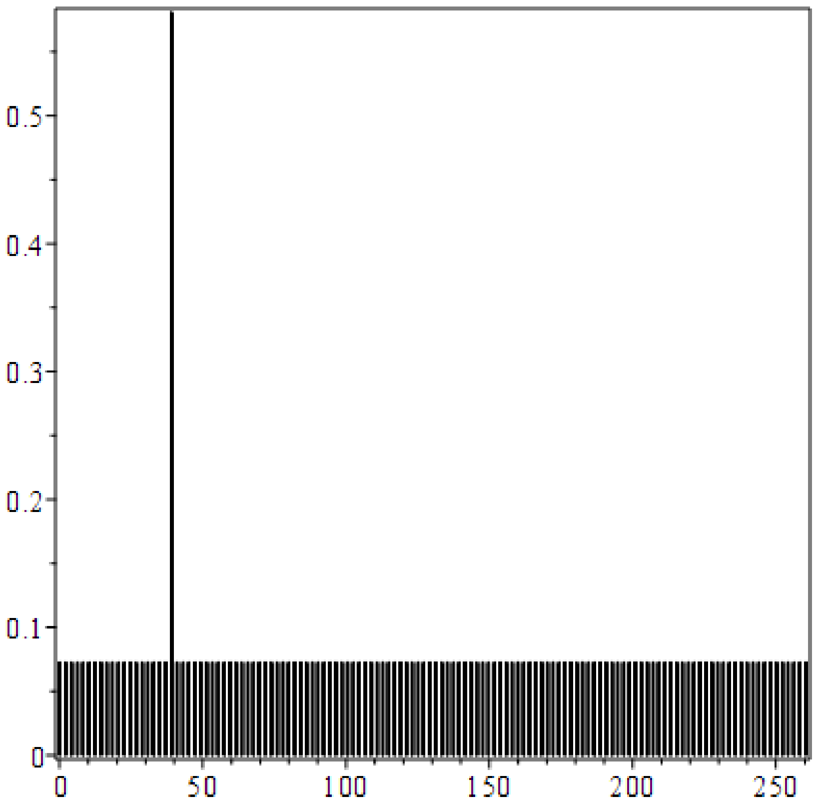


Second step

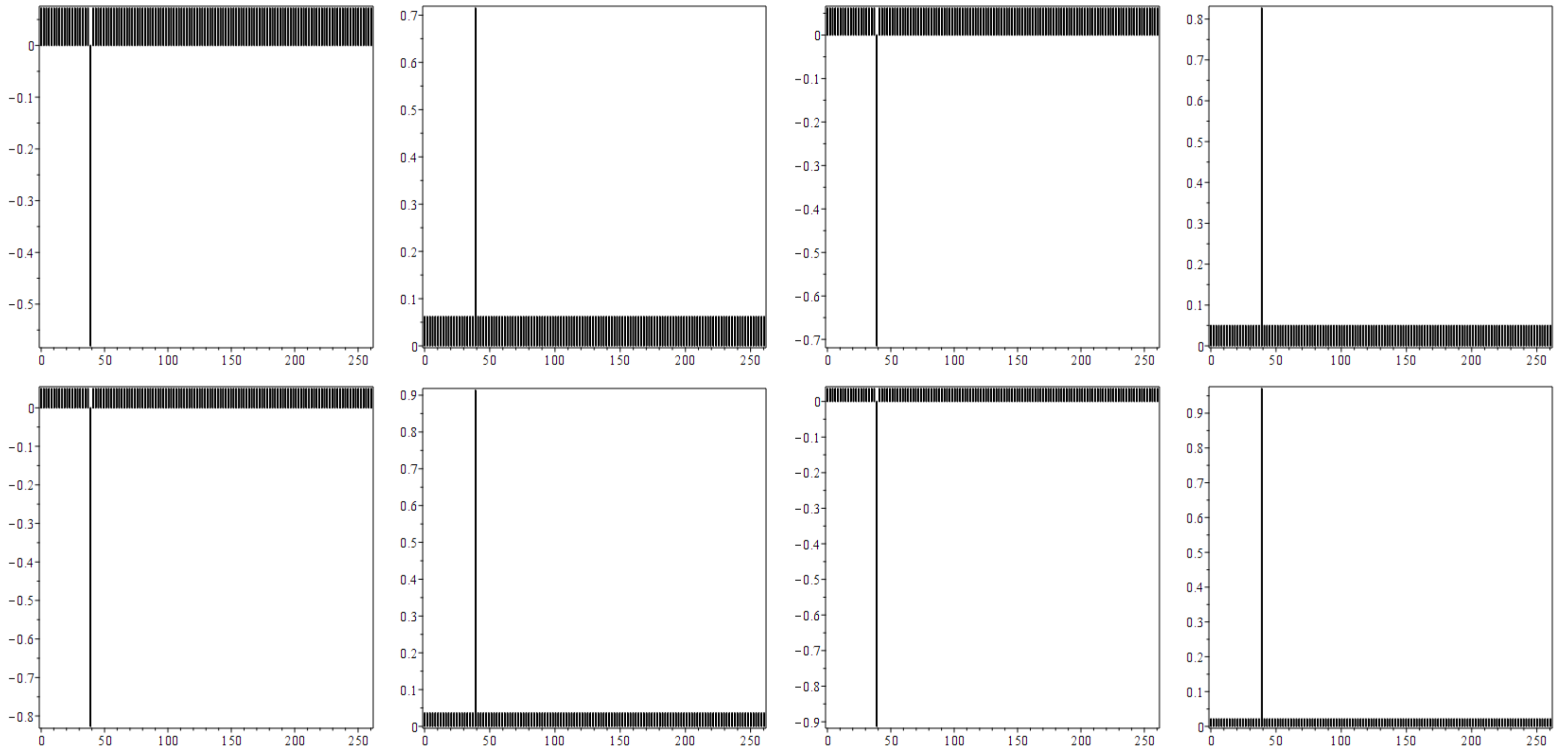


Third step

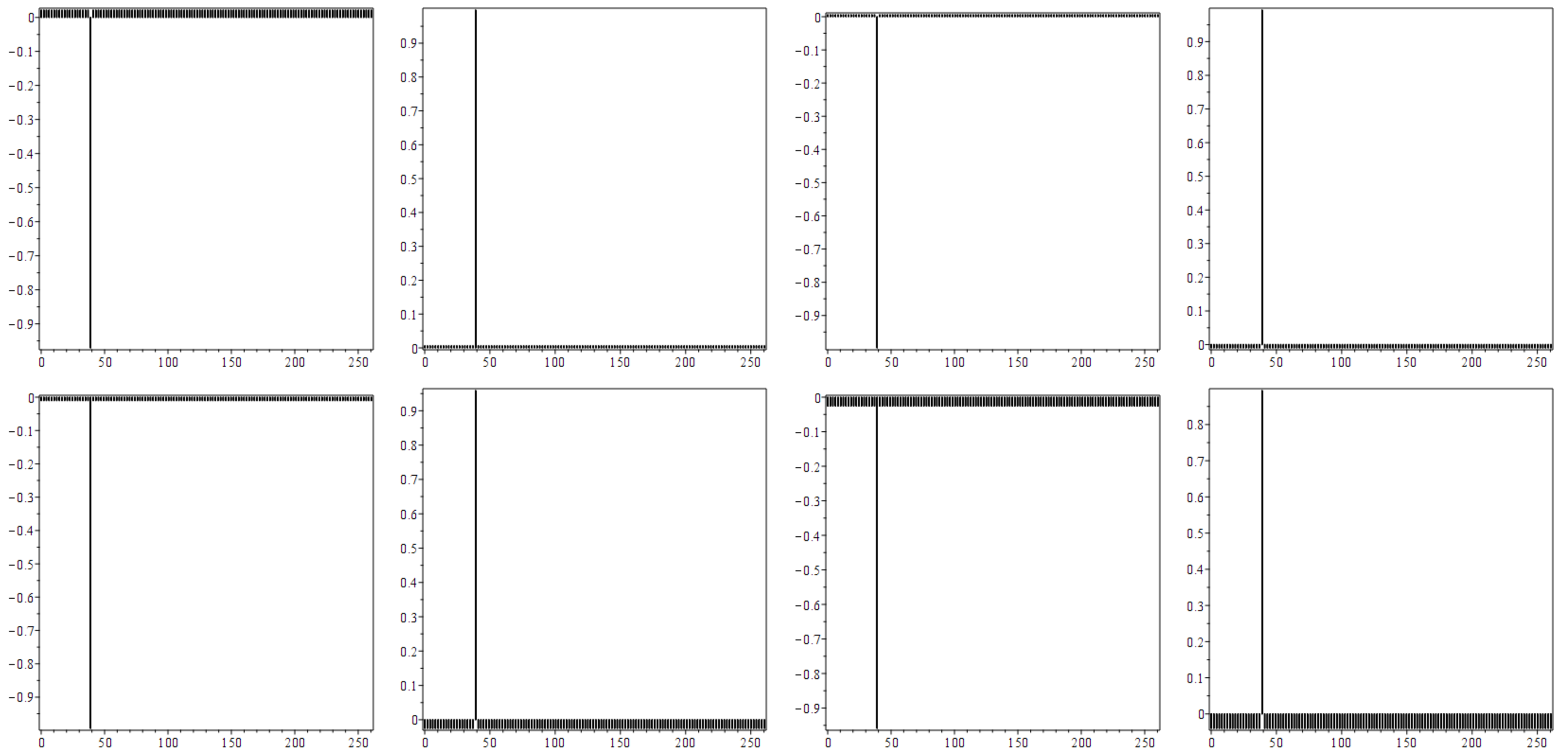

0
−0.1
−0.2
−0.3
−0.4
0
50
100
150
200
250


0.5
0.4
0.3
0.2
0.1
0
0
50
100
150
200
250

Steps 4-7



Steps 8-11


0 −0.1 −0.2 −0.3 −0.4 −0.5 −0.6 −0.7 −0.8 −0.9
0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9
0 50 100 150 200 250

An algebraic proof

$$N \stackrel{\text{def}}{=} 2^n$$

$$t \stackrel{\text{def}}{=} \#\{x : f(x) = 1\}$$

$$|\psi_k\rangle \stackrel{\text{def}}{=} \text{state after } k \text{ iterations}$$

$$|\psi_k\rangle = \sum_{x:f(x)=1} a_k |x\rangle + \sum_{x:f(x)=0} b_k |x\rangle$$

Algebraic proof(I)

$$\begin{aligned}
a_0 = b_0 &= \frac{1}{\sqrt{N}} \\
\left|\psi'_k\right\rangle &= -a_k \sum_{x:f(x)=1} |x\rangle + b_k \sum_{x:f(x)=0} |x\rangle \\
|\psi_{k+1}\rangle &= \sum_{x:f(x)=1} \underbrace{(2\langle\psi'_k\rangle + a_k)}_{a_{k+1}} |x\rangle + \sum_{x:f(x)=0} \underbrace{(2\langle\psi'_k\rangle - b_k)}_{b_{k+1}} |x\rangle \\
\langle\psi'_k\rangle &= -\frac{t}{N} a_k + \left(1 - \frac{t}{N}\right) b_k \\
a_{k+1} &= \left(1 - \frac{2t}{N}\right) a_k + \left(2 - \frac{2t}{N}\right) b_k \\
b_{k+1} &= -\frac{2t}{N} a_k + \left(1 - \frac{2t}{N}\right) b_k
\end{aligned}$$

Algebraic proof(II)

$$
\begin{aligned}
\sin\theta &\stackrel{\text{def}}{=} \sqrt{\frac{t}{N}} \\
\mathbf{P} &\stackrel{\text{def}}{=} \begin{pmatrix} 1-\frac{2t}{N} & 2-\frac{2t}{N} \\ -\frac{2t}{N} & 1-\frac{2t}{N} \end{pmatrix} \\
&= \begin{pmatrix} \cos 2\theta & 2\cos^2\theta \\ -2\sin^2\theta & \cos 2\theta \end{pmatrix}
\end{aligned}
$$

The eigenvalues of $\mathbf{P}$ are readily seen to be equal to $e^{\pm 2i\theta}$ and therefore

$$
\begin{aligned}
a_k &= A_- e^{-2ik\theta} + A_+ e^{-2ik\theta} \\
b_k &= B_- e^{-2ik\theta} + B_+ e^{-2ik\theta}
\end{aligned}
$$

Algebraic proof(III)

$$
\begin{aligned}
a_k &= \frac{1}{\sqrt{t}} \sin\left((2k+1)\theta\right) \\
b_k &= \frac{1}{\sqrt{N-t}} \cos\left((2k+1)\theta\right)
\end{aligned}
$$

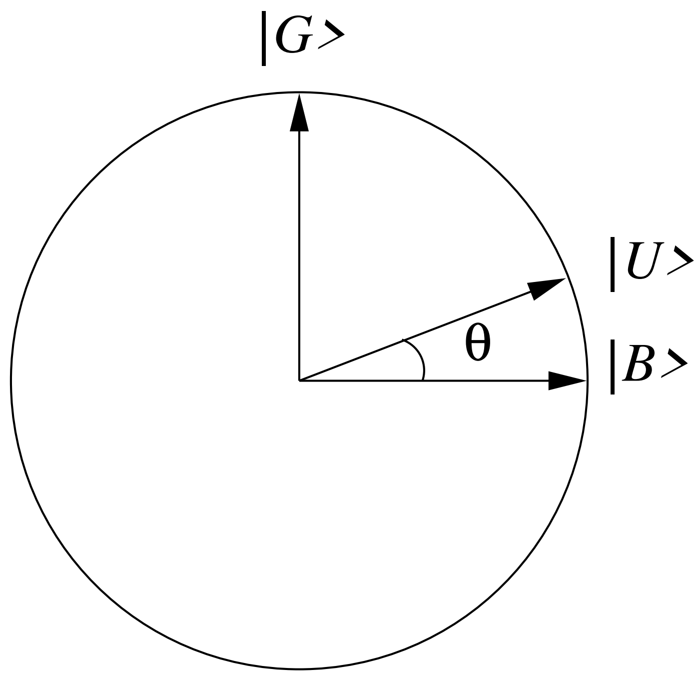
- Probability of seing a solution $P_k = \sin^2((2k+1)\theta)$

$$
\begin{aligned}
\tilde{k} &\stackrel{\text{def}}{=} \frac{\pi}{4\theta} - \frac{1}{2} \\
k &\stackrel{\text{def}}{=} \text{closest integer to } \tilde{k} \\
1 - P_k &= \cos^2((2k+1)\theta) \\
&= \cos^2((2\tilde{k}+1)\theta + 2(k-\tilde{k})\theta) \\
&= \cos^2\left(\frac{\pi}{2} + 2(k-\tilde{k})\theta\right) \\
&= \sin^2(2(k-\tilde{k})\theta) \leq \sin^2\theta = \frac{t}{N}
\end{aligned}
$$

A geometric proof

$$
\begin{aligned}
N &\stackrel{\text{def}}{=} 2^n \\
t &\stackrel{\text{def}}{=} \#\{x : f(x) = 1\} \\
|G\rangle &\stackrel{\text{def}}{=} \frac{1}{\sqrt{t}} \sum_{x:f(x)=1} |x\rangle \\
|B\rangle &\stackrel{\text{def}}{=} \frac{1}{\sqrt{N-t}} \sum_{x:f(x)=0} |x\rangle \\
|U\rangle &\stackrel{\text{def}}{=} \frac{1}{\sqrt{N}} \sum_{x} |x\rangle \\
&= \sin\theta\,|G\rangle + \cos\theta\,|B\rangle \quad \text{with} \\
\sin\theta &= \sqrt{\frac{t}{N}}
\end{aligned}
$$

The $\{|G\rangle, |B\rangle\}$ plane



Reflections

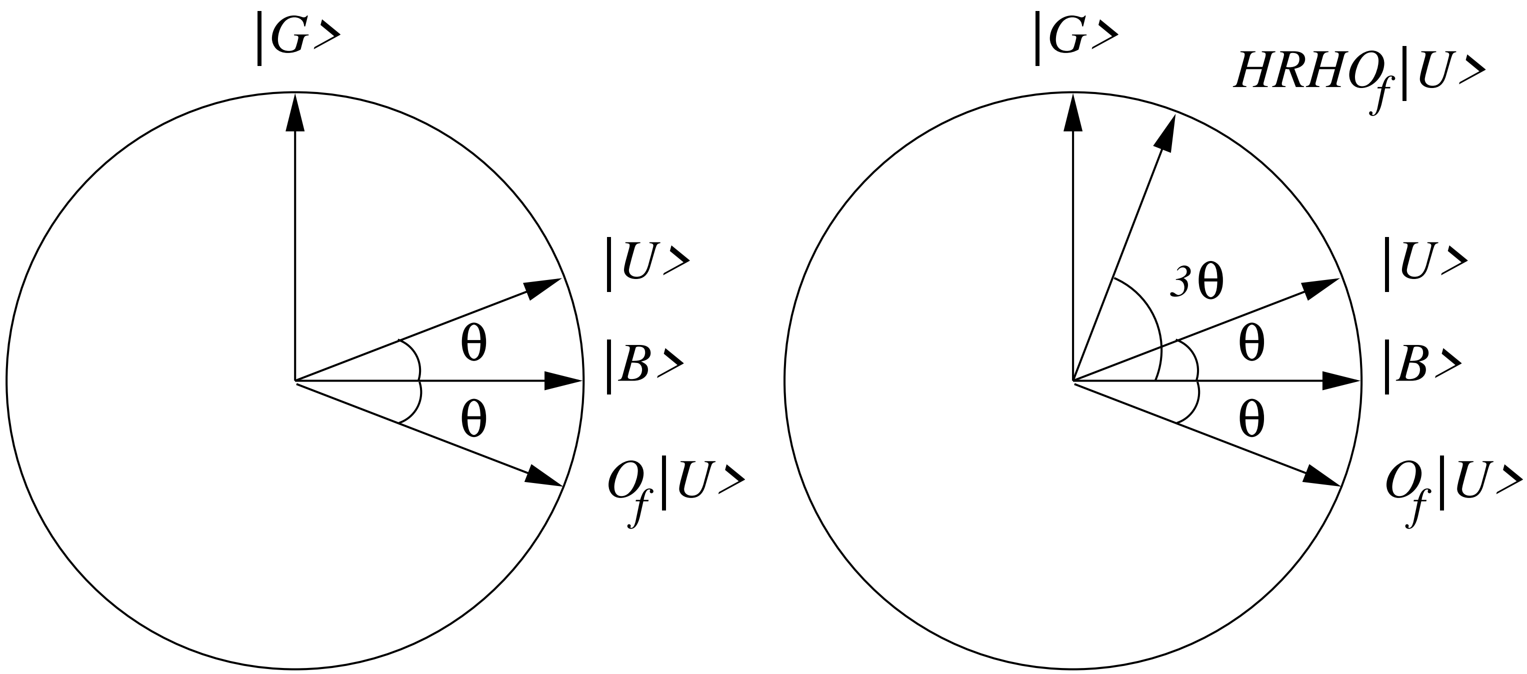
- $O_f =$ reflection through $|B\rangle$

$$\begin{aligned} O_f |B\rangle &= |B\rangle \\ O_f |G\rangle &= -|G\rangle \end{aligned}$$

- $\mathbf{H}^{\otimes n}\mathbf{R}\mathbf{H}^{\otimes n} = 2|U\rangle\langle U| - \mathbf{Id}$ reflection through $|U\rangle$

$$\begin{aligned} (2|U\rangle\langle U| - \mathbf{Id})|U\rangle &= 2\langle U|U\rangle|U\rangle - |U\rangle \\ &= |U\rangle \\ (2|U\rangle\langle U| - \mathbf{Id})\left|U^\perp\right\rangle &= 2\left\langle U|U^\perp\right\rangle|U\rangle - \left|U^\perp\right\rangle \\ &= -\left|U^\perp\right\rangle \end{aligned}$$

The picture



Iterating the reflections

- Initial state

$$\sin\theta\,|G\rangle + \cos\theta\,|B\rangle$$

- Each iteration = rotation of an angle 2θ, after k iterations we have

$$\sin((2k+1)\theta)\,|G\rangle + \cos((2k+1)\theta)\,|B\rangle$$

- Probability of seing a solution

$$P_k = \sin^2((2k+1)\theta) \geq 1 - \frac{t}{N}$$

  for k chosen as the closest integer to $\frac{\pi}{4\theta} - \frac{1}{2}$

- The algorithm given in this way needs to know t to stop when the number of iterations k is the closest integer to $\frac{\pi}{4\theta} - \frac{1}{2}$ where $\theta = \sin^{-1}\left(\sqrt{\frac{t}{N}}\right)$

Complexity = $O\left(\sqrt{\frac{N}{t}}\right)$

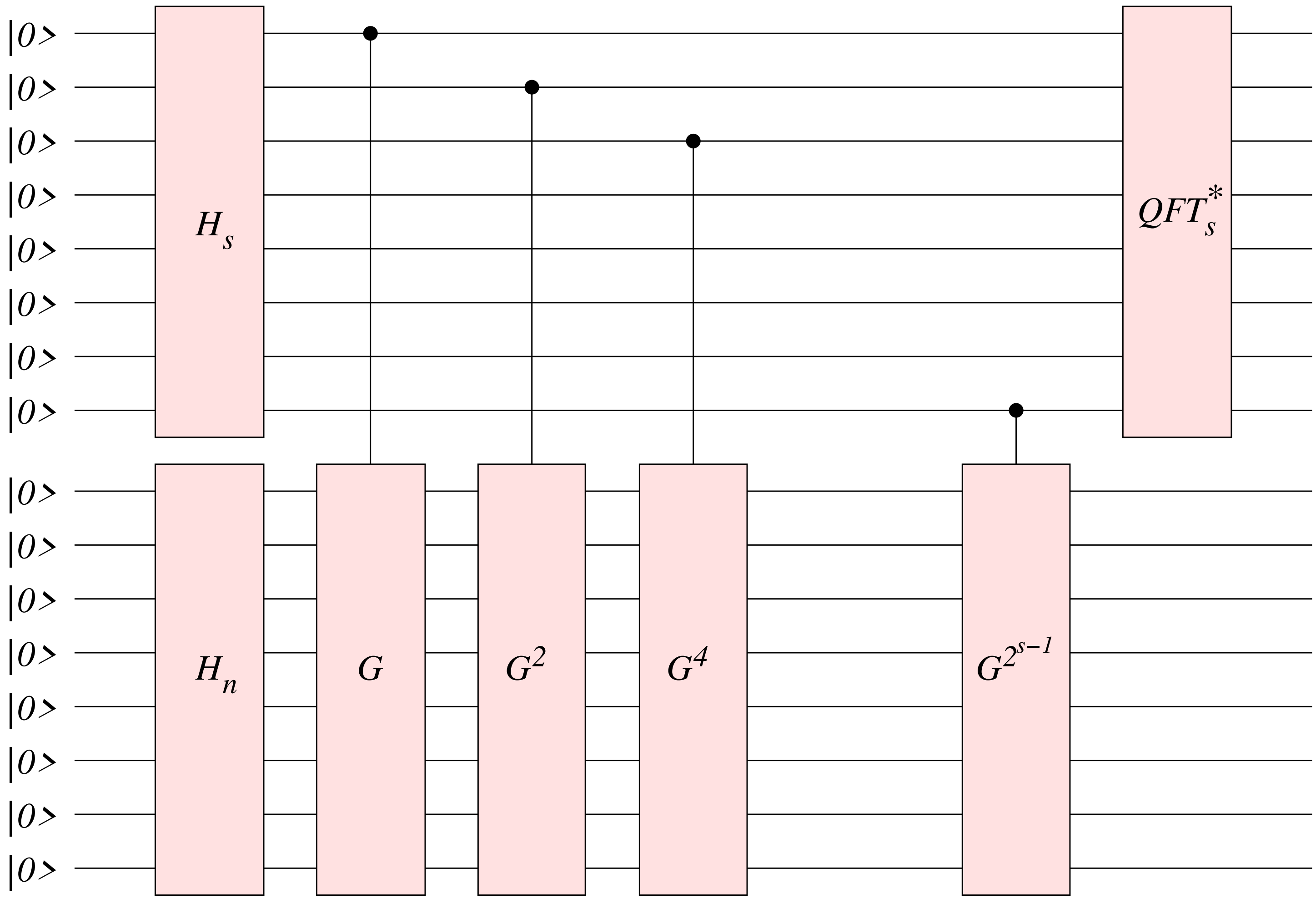
Exercise : do we need to know t? (Quantum counting)

1. Let $\mathbf{G} \stackrel{\text{def}}{=} \mathbf{H}^{\otimes n}\mathbf{R}\mathbf{H}^{\otimes n} O_f$ and let $|U\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle$. What is the dimension of the space V generated by the $\mathbf{G}^i |U\rangle$'s ?
2. What are the eigenvalues of $\mathbf{G}$ restricted to V ?
3. Give a quantum algorithm that estimates these eigenvalues up to s bits of precision.

Quantum counting

1. $\dim V = 2$ (generated by $|B\rangle$ and $|G\rangle$)
2. The eigenvalues are $e^{2i\theta}$ and $e^{-2i\theta}$ where $\sin\theta = \sqrt{\frac{t}{N}}$
3. This is the phase estimation algorithm of Lecture 4.

Quantum counting: the circuit



Quantum counting: the analysis

- Estimating the eigenvalue $\pm\theta$ can be done with a precision of 2^{-s} by using $\mathbf{QFT}^*_{2^s}$ and s auxiliary qubits. Estimation holds with some probability $\geq 1-\varepsilon$

$$
\begin{aligned}
\sin^2\theta &\stackrel{\text{def}}{=} \frac{t}{N} \\
\frac{|\Delta t|}{N} &= \left|\sin^2(\theta+\Delta\theta)-\sin^2\theta\right| \\
&< |2\sin\theta+|\Delta\theta||\,|\Delta\theta| \\
|\Delta\theta| &\leq 2^{-s} \\
\Rightarrow |\Delta t| &< \left(2\sqrt{tN}+\frac{N}{2^s}\right)2^{-s} \\
&= O(\sqrt{t}) \ \text{ for } 2^s=\sqrt{N}
\end{aligned}
$$

2. Amplitude amplification

- More general version of Grover's algorithm
  - Boolean function $\chi : X \to \{0,1\}$
  - Quantum algorithm $\mathcal{A}$ such that $\mathcal{A}|0\rangle = \sum_{x\in X} \alpha_x |x\rangle$ that has probability p of finding an element $x \in X$ for which $\chi(x) = 1$, when $\mathcal{A}|0\rangle$ is measured i.e. $p = \sum_{x:\chi(x)=1} |\alpha_x|^2$
- Classically we need to run $\mathcal{A}$ $\frac{1}{p}$ times
- Quantumly we only need to run $\mathcal{A}$ and $\mathcal{A}^{-1}$ $O(\frac{1}{\sqrt{p}})$ times

Amplitude amplification algorithm

1. Setup the starting state $|U\rangle = \mathcal{A}|0\rangle$
2. Repeat the following $O(\frac{1}{\sqrt{p}})$ times
   (a) apply $O_\chi : |x\rangle \mapsto (-1)^{\chi(x)}$ (= reflect through $|B\rangle$)
   (b) apply $\mathcal{A}\mathbf{R}\mathcal{A}^{-1}$ (=reflect through $|U\rangle$)
3. measure and verify that the outcome $|x\rangle$ is such that $\chi(x) = 1$

Amplitude amplification

- The analysis on Grover's search algorithm actually shows in this case a stronger statement. Let V be the space $\langle |x\rangle : \chi(x) = 1\rangle$. We have in our case

$$\mathcal{A}\,|0\rangle = \alpha\,|\phi_V\rangle + \beta\,\left|\phi_V^{\perp}\right\rangle$$

where $|\alpha|^2 = p$. The quantum amplitude amplification algorithm produces a state close to $|\phi_V\rangle$

Exercise : collision search

Let

$$f : \{0,1\}^n \to \{0,1\}^n$$

which is assumed to be to 2 to 1, for each $x \in \{0,1\}$ there is exactly one other y such that $f(x) = f(y)$. Such a pair is called a collision.

1. Choose S uniformly at random among the sets of size s in $\{0,1\}^n$. What is the expected number of solutions in S ?
2. Give a classical randomized algorithm that finds a collision (with probability $\geq 2/3$ say) using $O(\sqrt{2^n})$ queries to f
3. Give a quantum algorithm that finds a collision using $O(2^{n/3})$ queries to f

collision search

1. $\frac{s(s-1)}{2(2^n-1)}$
2. Choosing a set of size $\Omega(2^{n/2})$
3. Choosing a set S of size $\Omega(2^{n/3})$, check that there is no collision in it, then define

$$g(x) = 1 \text{ iff } \exists y \in S : f(y) = f(x)$$

and use Grover's algorithm

Exercise : collision finding with poly(n) quantum memory

We keep the same notation as before, but model now f as a random function. Let $S_r \stackrel{\text{def}}{=} \left\{ (x, f(x)) : \exists z \in \{0,1\}^{n-r},\ f(x) = \underbrace{0 \cdots 0}_{r \text{ times}} || z \right\}$ and consider the following algorithm

(i) Construct a list L consisting of 2^{t-r} elements from S_r. Let $g : \{0,1\}^n \rightarrow \{0,1\}$ where $g(x) = 1$ if and only if there is an $(x', f(x'))$ in L such that $f(x) = f(x')$.

(ii) apply the quantum amplification algorithm where

- the initialization consists in the construction of $|\psi\rangle \stackrel{\text{def}}{=} \frac{1}{\sqrt{|S_r|}} \sum_{(x,f(x)) \in S_r} |x, f(x)\rangle$
- the oracle is O_g

1. How do you perform (i) and (ii) ? What are the costs (complexity, quantum memory, classical memory) of steps (i) and (ii) ?
2. What are the classical and quantum memory costs of this algorithm ?
3. What is the optimal quantum complexity of this algorithm for a polynomial quantum memory cost ?

collision finding with poly(n) quantum memory

1. (i) can be done with Grover with $f_r(x) = 1$ if $(x, f(x)) \in S_r$. Probability that a given x evaluates to $1 = O(2^{-r}) \Rightarrow$ Complexity $O(2^{r/2})$ of a Grover call
   - overall quantum complexity $O(2^{t-r/2})$
   - quantum memory poly(n)
   - classical memory $O(2^{t-r})$

   (ii) detailing each step

setup: (constructing $|\phi_r\rangle$) done by amplitude amplification with $g_r(x) = 1$ if $(x, f(x)) \in S_r$ and $\mathcal{A}\,|0\rangle = \frac{1}{2^{n/2}} \sum_x |x\rangle$

- quantum complexity $O(2^{r/2})$
- quantum memory poly(n)

O_g : testing sequentially against the elements of L

- quantum complexity $O(2^{t-r})$
- quantum memory poly(n)

Step (ii) is essentially a Grover search for g with input space S_r

$$\mathbf{Prob}(g(x) = 1 | (x, f(x)) \in S_r) = O\left(\frac{2^{t-r}}{2^{n-r}}\right) = O(2^{t-n})$$

$$\Rightarrow \text{qu. comp. of (ii)} = O\left(\underbrace{2^{\frac{n-t}{2}}}_{\#\text{ Grover iter.}} \left[\underbrace{2^{r/2}}_{\text{setup}} + \underbrace{2^{t-r}}_{O_g}\right]\right)$$

Overall complexity

- time
$$O\left(2^{t-r/2} + 2^{\frac{n-t}{2}}\left[2^{r/2} + 2^{t-r}\right]\right)$$
- quantum memory poly(n)
- classical memory 2^{t-r}

2. Optimization
   - $r/2 = t - r \Rightarrow r = \frac{2}{3}t$
   - $\frac{n-t}{2} + r/2 = t - r/2 \Rightarrow \frac{n}{2} - \frac{t}{6} = \frac{2t}{3} \Rightarrow t = \frac{3n}{5}$
   - Overall complexity
     - time $O(2^{\frac{2n}{5}})$
     - classical memory $O(2^{\frac{n}{5}})$

3. Lower bound on the query complexity

- Assumptions
  - only one solution x :
$$O_x = \mathbf{Id} - 2\,|x\rangle\,\langle x|$$
  - the algorithm starts in a state $|\psi\rangle$ and applies the oracle O_x exactly k times with unitary operations $\mathbf{U}_1, \cdots, \mathbf{U}_k$ interleaved between the oracle calls

$$|\psi_k^x\rangle \stackrel{\text{def}}{=} \mathbf{U}_k O_x \mathbf{U}_{k-1} O_x \cdots \mathbf{U}_1 O_x\,|\psi\rangle$$
$$|\psi_k\rangle \stackrel{\text{def}}{=} \mathbf{U}_k \mathbf{U}_{k-1} \cdots \mathbf{U}_1\,|\psi\rangle$$
$$D_k \stackrel{\text{def}}{=} \sum_x \left\| |\psi_k^x\rangle - |\psi_k\rangle \right\|^2$$

It turns out that

(i) $D_k \leq 4k^2$
(ii) to distinguish among N alternatives we need $D_k = \Omega(N)$

This implies
$$k = \Omega(\sqrt{N})$$

Induction for $D_k \leq 4k^2$

$$
\begin{aligned}
D_0 &= 0 \\
D_{k+1} &= \sum_x \left\| O_x \left|\psi_k^x\right\rangle - \left|\psi_k\right\rangle \right\|^2 \\
&= \sum_x \left\| O_x(\left|\psi_k^x\right\rangle - \left|\psi_k\right\rangle) + (O_x - \mathbf{Id}) \left|\psi_k\right\rangle \right\|^2 \\
&\leq \sum_x \left(\left\| \left|\psi_k^x\right\rangle - \left|\psi_k\right\rangle \right\|^2 + 4 \left\| \left|\psi_k^x\right\rangle - \left|\psi_k\right\rangle \right\| \left| \langle x|\psi_k\rangle \right| + 4 \left| \langle \psi_k|x\rangle \right|^2 \right) \qquad (1) \\
&\leq D_k + 4 \left(\sum_x \left\| \left|\psi_k^x\right\rangle - \left|\psi_k\right\rangle \right\|^2 \right)^{\frac{1}{2}} \left(\sum_x \left| \langle x|\psi_k\rangle \right|^2 \right)^{\frac{1}{2}} \leq D_k + 4\sqrt{D_k} + 4
\end{aligned}
$$

we used: $\|b + c\|^2 \leq \|b\|^2 + 2\,\|b\|\,\|c\| + \|c\|^2$ with

$$
\begin{aligned}
b &\stackrel{\text{def}}{=} O_x(\left|\psi_k^x\right\rangle - \left|\psi_k\right\rangle) \\
c &\stackrel{\text{def}}{=} (O_x - \mathbf{Id}) \left|\psi_k\right\rangle \\
&= -2 \langle x|\psi_k\rangle \left|x\right\rangle \quad \text{for (1)} \qquad (2) \\
\sum_x \left| \langle x|\psi_k\rangle \right|^2 &= 1 \quad \text{for the last inequality} \qquad (3)
\end{aligned}
$$

Exercise: $D_k = \Omega(N)$

Let

$$E_k \stackrel{\text{def}}{=} \sum_x \left\| |\psi_k^x\rangle - |x\rangle \right\|^2$$

$$F_k \stackrel{\text{def}}{=} \sum_x \left\| |x\rangle - |\psi_k\rangle \right\|^2$$

1. Show by using the Cauchy-Schwarz inequality that $D_k \geq (\sqrt{F_k} - \sqrt{E_k})^2$
2. Show that $F_k \geq 2N - 2\sqrt{N}$
3. Show that if the probability of recovering the right x for any x is greater than $\frac{1}{2}$ then $E_k \leq (2 - \sqrt{2})N$
4. Show that under the same assumption as in the previous point, we have $D_k = \Omega(N)$

The polynomial method

- The query model:
  - want to compute some function $f : \{0,1\}^N \to \{0,1\}$ on a given input $\mathbf{x} = x_0 \cdots x_{N-1}$
  - $\mathbf{x}$ is not given explicitly can be queried through a quantum operation

$$O_x : |i, b\rangle \mapsto |i, b \oplus x_i\rangle$$

  - cost : number of queries to O_x, i.e. T when we perform

$$\mathbf{U}_T O_x \mathbf{U}_{T-1} O_x \cdots O_x \mathbf{U}_1 O_x \mathbf{U}_0 |0 \cdots 0\rangle$$

- Example:
  $f(x) = x_0 \vee x_1 \vee \cdots \vee x_{N-1}$ and $N = 2^n$
  $\Leftrightarrow$ knowing whether one of the x_i's evaluate to 1
  $\Leftrightarrow$ the function $g(i) = x_i$ evaluates to 1 on at least one entry

From quantum queries to polynomials

$$p(x_0, \ldots, x_{N-1}) = \sum_{S \subseteq \{0, \cdots, N-1\}} a_s \Pi_{i \in S} x_i$$

$$\deg(p) \stackrel{\text{def}}{=} \max\{|S| : a_S \neq 0\}$$

Fact 1. *The final state of a T query algorithm with input $\mathbf{x} \in \{0,1\}^N$ acting on an m-qubit space can be written as*

$$\sum_{z \in \{0,1\}^m} a_z(\mathbf{x}) |z\rangle$$

where each $a_z(\mathbf{x})$ is a polynomial in $\mathbf{x}$ of degree at most T

Proof of the fact

By induction on T. Clearly true for $T = 0$. Assume that the property holds for T queries. Applying a unitary does not change the state of the state $\Rightarrow$ the $a_z(\mathbf{x})$'s are polynomial in $\mathbf{x}$ of degree $\leq T$. Register of the form

$$|i, b, w\rangle$$

Query swaps $|i, 0, w\rangle$ and $|i, 1, w\rangle$ iff $x_i = 1$, therefore

$$\alpha(x)\,|i, 0, w\rangle + \beta(x)\,|i, 1, w\rangle \mapsto ((1-x_i)\alpha(x) + x_i\beta(x))\,|i, 0, w\rangle + ((1-x_i)\beta(x) + x_i\alpha(x))\,|i, 1, w\rangle$$

$$\Rightarrow \deg \alpha^{T+1}(x) \leq T + 1$$

The second ingredient

Assume algorithm $\mathcal{A}$ works on m qubits and the outcome is the first qubit. The probability of output 1 is therefore

$$p(x) = \sum_{z \in \{1\} \times \{0,1\}^{m-1}} |\alpha_z(x)|^2$$

and $p(x)$ is a polynomial of degree $\leq 2T$.

$\mathcal{A}$ computes f with err. prob. $\leq \frac{1}{3}$

$\Downarrow$

if $f(x) = 0$ then $p(x) \in [0, 1/3]$

if $f(x) = 1$ then $p(x) \in [2/3, 1]$

$\Downarrow$

p approximates f

Application

- symmetric function $f(x) = f(\pi(x))$ for any permutation π of the coordinates: OR, AND, Parity, Majority

In such a case $q(x)$ defined by

$$q(x) = \frac{1}{N!} \sum_{\pi \in S_N} p(\pi(x)) = \sum_{i=0}^{d} a_i \binom{|x|}{i}$$

also approximates f. Moreover there is a single variable polynomial r such that

$$q(x) = r(|x|)$$

(choose $r(z) \stackrel{\text{def}}{=} \sum_{i=0}^{d} a_i \binom{z}{i}$)

OR

$$
\begin{aligned}
r(0) &\in [0, 1/3] \\
r(t) &\in [2/3, 1] \qquad \text{for } t \in \{1, \cdots, N\}
\end{aligned}
$$

$$\Downarrow$$

$$\deg r \geq \Omega(\sqrt{N})$$