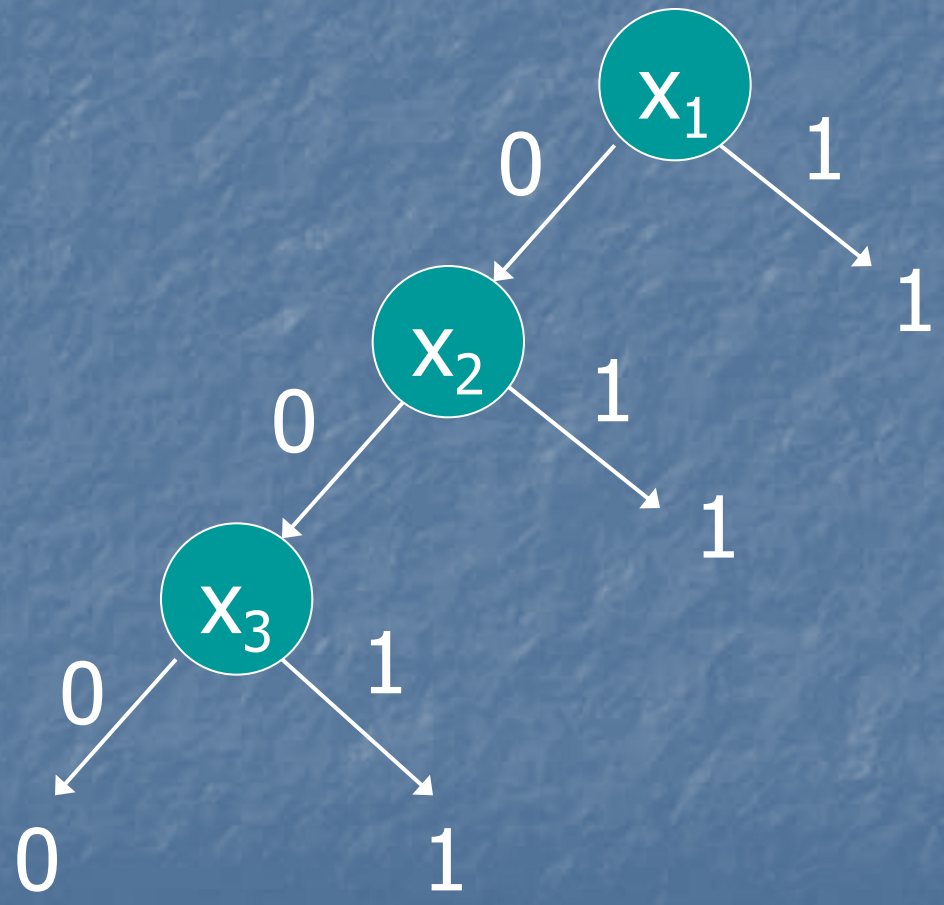
Quantum lower bounds

Andris Ambainis
University of Latvia

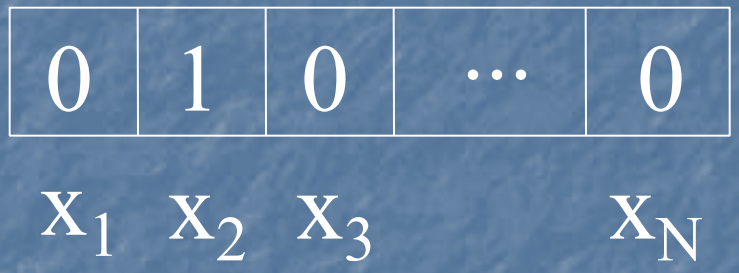
Query model

- Function $f(x_1, \ldots, x_N)$, $x_i \in \{0,1\}$.
- x_i given by a black box:

Decision trees

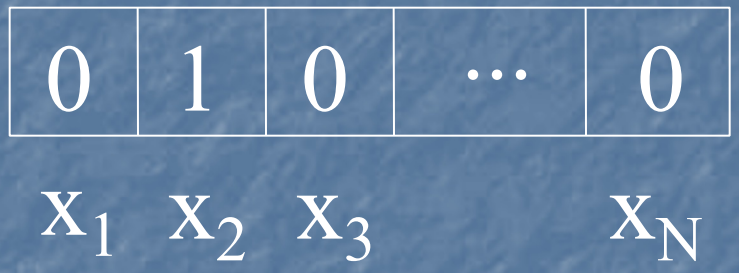


Grover's search



- Does there exist $i{:}x_i{=}1$?
- Queries: ask i, get x_i.
- Classically, N queries required.
- Quantum: $O(\sqrt{N})$ [Grover, 1996].
- This is optimal [Berstein et al., 1997].

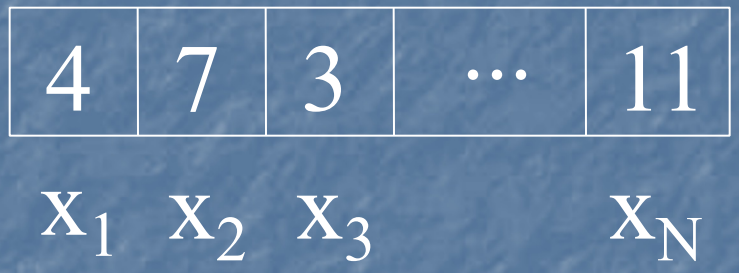
Quantum counting



- Distinguish whether the fraction of $x_i=1$ is $\leq 1/2-\varepsilon$ or $\geq 1/2+\varepsilon$.
- Classical random sampling: $O(1/\varepsilon^2)$ steps.
- Quantum: $O(1/\varepsilon)$ steps [Brassard et al., 2000].
- This is optimal [A, 2000].

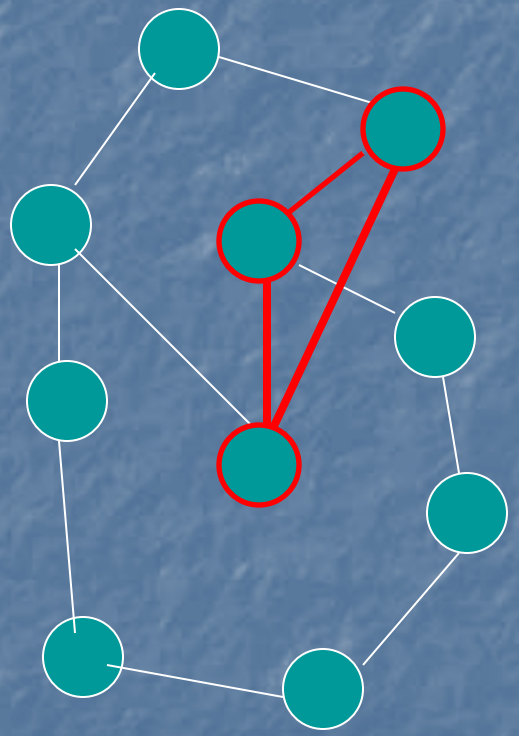
Can be used for more complicated statistics.

Element distinctness



- Numbers $x_1, x_2, \ldots, x_N$.
- Determine if two of them are equal.
- Classically: N queries.
- Quantum: $O(N^{2/3})$ [A, 2004].
- This is optimal [Aaronson, Shi, 2003].

Triangle finding



- Graph G with n vertices.
- n^2 variables x_{ij}; $x_{ij}=1$ if there is an edge (i, j).
- Does G contain a triangle?
- Classically: $O(n^2)$.
- [Lee, et al., 2013] Quantum: $O(n^{9/7})$.

Queries in the quantum world

- Basis states: $|1,1\rangle$, $|1, 2\rangle$, …, $|N, M\rangle$.
- State:

  $$\alpha_{1,1}|1,1\rangle + \alpha_{1,2}|1, 2\rangle + \ldots + \alpha_{N, M}|N, M\rangle.$$

- Query:
  - $|i, j\rangle \rightarrow |i, j\rangle$, if $x_i=0$;
  - $|i, j\rangle \rightarrow -|i, j\rangle$, if $x_i=1$;

Example

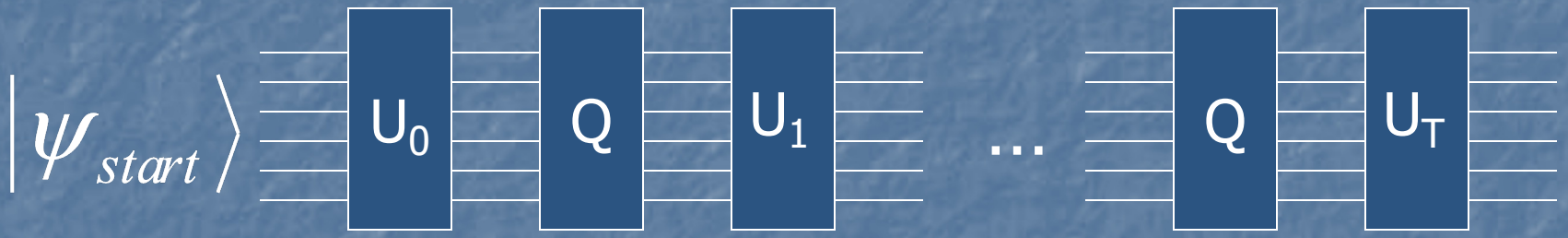
x_1	x_2	x_3
0	1	0

$$\alpha_{1,1}|1, 1\rangle+\alpha_{1,2}|1, 2\rangle+\alpha_{2,1}|2, 1\rangle+\alpha_{3,1}|3,1\rangle$$

Query

$$\alpha_{1,1}|1, 1\rangle+\alpha_{1,2}|1, 2\rangle- \alpha_{2,1}|2, 1\rangle+\alpha_{3,1}|3,1\rangle$$

Quantum query model



- Fixed starting state.
- $U_0, U_1, \ldots, U_T$ – independent of $x_1, \ldots, x_N$.
- Q – queries.
- Measuring final state gives the result.

Quantum query complexity

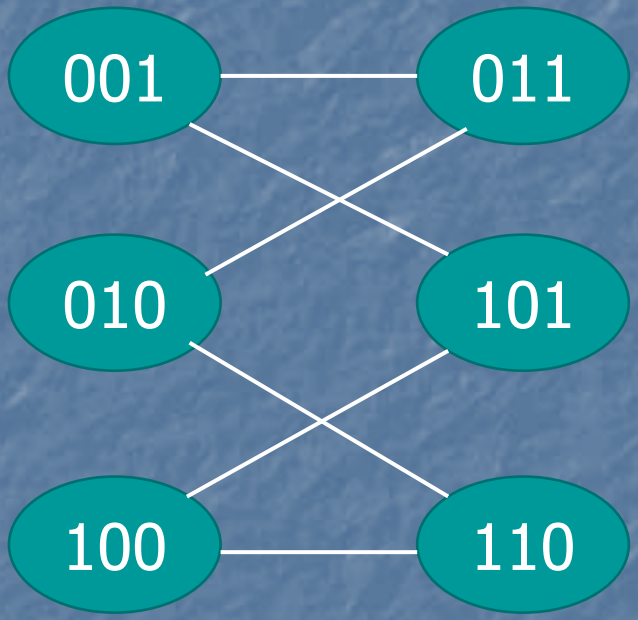
- $Q_E(f)$ – for any $x_1, \ldots, x_N$, measuring the final state of the algorithm always gives $f(x_1, \ldots, x_N)$.

- $Q_2(f)$ – for any $x_1, \ldots, x_N$, measuring the final state of the algorithm gives $f(x_1, \ldots, x_N)$ with probability $\geq 2/3$.

Methods for proving quantum lower bounds

- Adversary method [Berstein et al., 1997, A, 2000]: quantum states as vectors, distance between algorithm's states on different $x_1, \ldots, x_N$.
- Polynomials method [Beals et al., 1998]: represent quantum algorithms by polynomials.

Simple adversary



- MAJ(x_1, x_2, x_3).
- $k = 2$ ways to transform $x \in A$ to $y \in B$.
- $l = 2$ ways to transform $y \in B$ to $x \in A$.
- $\Omega(\sqrt{kl})$ queries required.

Algebraic adversary

- Adversary matrix Γ: $\Gamma_{xy} = 0$ if $f(x)=f(y)$.
- $(\Gamma^{(i)})_{xy} = \Gamma_{xy}$ if $x_i \neq y_i$.
- $(\Gamma^{(i)})_{xy} = 0$ if $x_i = y_i$.
- Theorem $\Omega\left(\frac{\|\Gamma\|}{\max\|\Gamma_i\|}\right)$ queries are needed to compute f quantumly.

Algebraic adversary

- Theorem $\Omega\left(\frac{\|\Gamma\|}{\max\|\Gamma_i\|}\right)$ queries are needed to compute f quantumly.

$$\Gamma = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \end{pmatrix} \begin{matrix} 001 \\ 010 \\ 100 \\ 011 \\ 101 \\ 110 \end{matrix}$$

Algebraic adversary

- Theorem $\Omega\left(\frac{\|\Gamma\|}{\max\|\Gamma_i\|}\right)$ queries are needed to compute f quantumly.

- Positive weight adversary: $\Gamma_{xy} \geq 0$.
- Negative weight adversary: Γ_{xy} can be negative.

Reichardt, 2011

- $F(x_1, \ldots, x_N)$ – computational problem.
- T - best quantum lower bound for F provable by negative-weight adversary method.
- Theorem There is a quantum algorithm A that computes f with $O(T)$ queries.

Proof ideas (1)

- Method for quantum algorithms:
  - Span programs [Reichardt, Špalek, 2008];
- Method for quantum lower bounds:
  - Negative-weight adversary [Hoyer, Špalek, Lee, 2007];
- Maximizing the parameters in both methods = semidefinite program (generalization of linear program).

Proof ideas (2)

- Semidefinite programming duality:

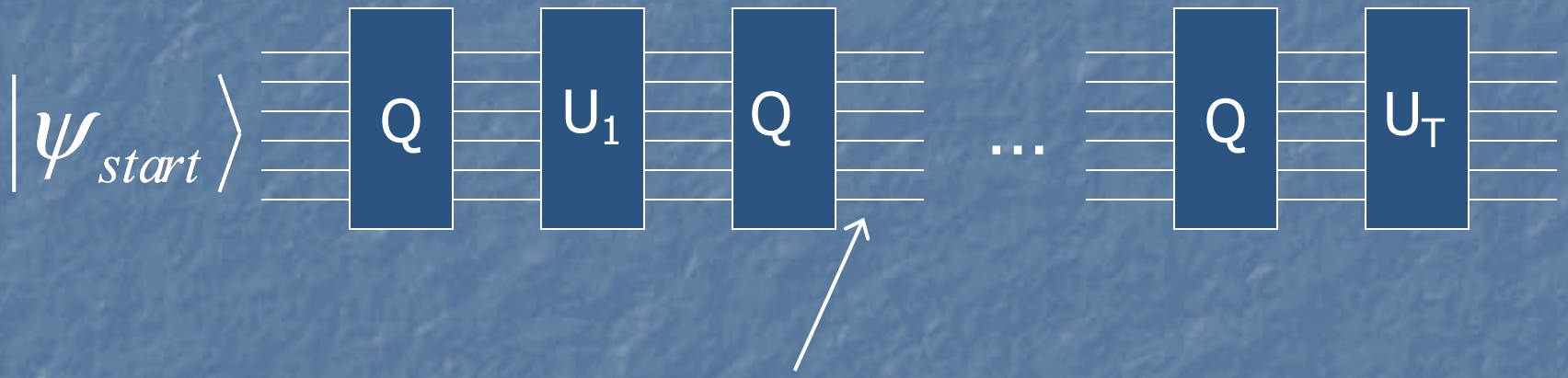
  Min (Primal program) = Max (Dual program).

- Primal program = Span program size;
- Dual program = Adversary lower bound.

Implies optimality for both methods.

Polynomials method

Analyzing query algorithms



$\alpha_{1,1}|1,1\rangle + \alpha_{1,2}|1, 2\rangle + \ldots + \alpha_{N, M}|N, M\rangle$

$\alpha_{1,1}$ is actually $\alpha_{1,1}(x_1, \ldots, x_N)$

Polynomials method

- Lemma [Beals et al., 1998] If

$$\sum_{i,j} \alpha_{i,j}(x_1, \ldots, x_N) |i, j\rangle$$

is a state after k queries, then $\alpha_{i,j}(x_1, \ldots, x_N)$ are polynomials in $x_1, \ldots, x_N$ of degree $\leq k$.

Measurement: (i, j) w. probability $\left|\alpha_{i,j}(x_1, \ldots, x_N)\right|^2$

Polynomial of degree $\leq 2k$

Implications

- Corollary 1 If f is computable with k quantum queries exactly (no error), there exists p: $\deg(p) \leq 2k$:

  $f=0 \rightarrow p=0, \quad f=1 \rightarrow p=1.$

- Corollary 2 If f is computable with k quantum queries with prob. $\geq 2/3$, there exists p: $\deg(p) \leq 2k$:

  $f=0 \rightarrow p \in [0, 1/3], \quad f=1 \rightarrow p \in [2/3, 1].$

Symmetric functions

- $F(x_1, \ldots, x_N)$ depends only on the number of $i{:}x_i=1$.
- Search: is there $i{:}x_i=1$?
- Threshold: are there at least k $i{:}x_i=1$?

- $F(m)$, m – number of $i{:}x_i=1$.

Lower bounds

- $F(m)$, m – number of $i{:}x_i=1$.

- Theorem If polynomial P satisfies
  - $P(0)\in[0, 1/3]$,
  - $P(k)\in[0, 1]$ for $k \in \{1, \ldots, N\}$,

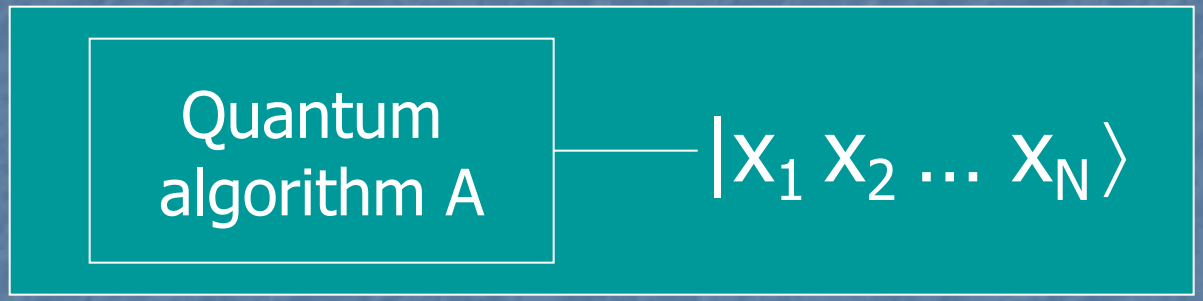
  then $\deg(P) = \Omega(\sqrt{N})$.

Applicability

- $P(x_1, \ldots, x_N)$ hard to analyze.
- Succesful applications use symmetry to reduce to $P(m)$ or $P(m, l)$.

Representation-theoretic adversary

State of algorithm's knowledge

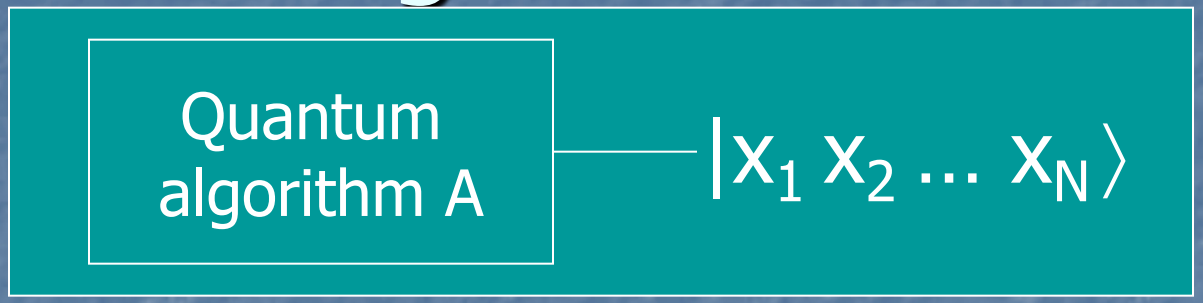


State:

$$\alpha_1|1\rangle_A \otimes |\Psi_1\rangle_I + \alpha_2|2\rangle_A \otimes |\Psi_2\rangle_I + \ldots$$

State $|\Psi_1\rangle$ quantifies algorithm's knowledge about the input if A is in state $|1\rangle$.

State of algorithm's knowledge



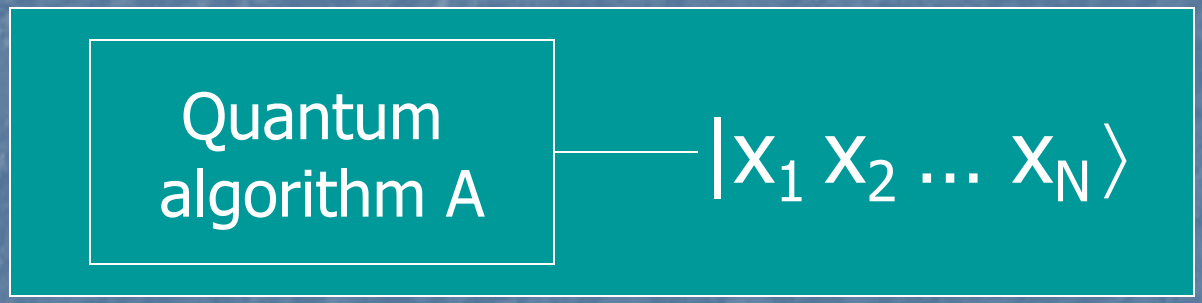
$$|\psi_0\rangle = \frac{1}{\sqrt{N}}\left(|10\ldots0\rangle + |01\ldots0\rangle + \ldots + |00\ldots1\rangle\right)$$

A has no information about the location of $x_i=1$.

$$\frac{1}{\sqrt{N}}\left(|1\rangle|10\ldots0\rangle + |2\rangle|01\ldots0\rangle + \ldots + |N\rangle|00\ldots1\rangle\right)$$

A has perfect information about the location of $x_i=1$.

Symmetrization



- For any algorithm A, there is another algorithm A′:
  - A′ has the same success probability as A.
  - State in $|x_1\, x_2 \ldots x_N\rangle$ register of A′ symmetric.

Example: Grover search

- Grover search; inputs $|10\ldots0\rangle$, $|01\ldots0\rangle$, …, $|00\ldots1\rangle$.
- σ^t - state of H_I after t steps.

$$\sigma^t = \begin{pmatrix} a & b & \cdots & b \\ b & a & \cdots & b \\ \vdots & \vdots & \ddots & \vdots \\ b & b & \ldots & a \end{pmatrix}$$

State of any search algorithm can be described by two parameters: a and b.

Representation theory

- Group G, linear space H.
- For each element $g \in G$, linear transformation U_g: $H \to H$.
- Transformations satisfy $U_{gh} = U_g U_h$.

- Irreducible representation: no decomposition $H = H_1 \oplus H_2$, $U_g: H_1 \to H_1$, $U_g: H_2 \to H_2$.

representation of G

Symmetries

- G – group of symmetries of $f(x_1, \ldots, x_N)$.
- Representation of G, for example:

$$U_g : |x_1 x_2 \ldots x_N\rangle \rightarrow |x_{\pi(1)} x_{\pi(2)} \ldots x_{\pi(N)}\rangle.$$

- Use representation theory to decompose

$$H = H_1 \oplus H_2 \oplus \ldots \oplus H_k,$$

H_i –irreducible representations.

If σ^t – state of $|x_1 x_2 \ldots x_N\rangle$ after t steps,

σ^t – invariant under all U_g.

Strategy

$$H = H_1 \oplus H_2 \oplus \ldots \oplus H_k,$$

- H_i –irreducible and invariant.
- ρ_i – completely mixed state over H_i.

Claim If σ^t – state of $|x_1 x_2 \ldots x_N\rangle$ after t steps, then

$$\sigma^t = p_{t,1}\,\rho_1 + p_{t,2}\,\rho_2 + \ldots + p_{t,k}\,\rho_k.$$

complete description of the algorithm

Examples

Example 1: Grover's search

- States of the input register

$$\alpha_1|10\ldots0\rangle+\alpha_2|01\ldots0\rangle+\ldots+\alpha_n|00\ldots1\rangle$$

- $H=H_0 \oplus H_1$.
- State after t steps: $\sigma^t = p\,\rho_0 + (1-p)\,\rho_1$.

- H_0 – no information about $i{:}x_i=1$.
- H_1 – full information about $i{:}x_i=1$.

Example 2: k-fold search [A, 2006]

- k locations $i:x_i=1$.
- Task: find all of them.
- States of the input register

$$\sum_{|\{i:x_i=1\}|=K} \alpha_{x_1 x_2 \cdots x_N} \left| x_1 x_2 \cdots x_N \right\rangle$$

- $H=H_0 \oplus H_1 \oplus \ldots \oplus H_k$.
- H_j – algorithm knows j of k locations $i:x_i=1$.
- State after t steps: $\sigma^t = \rho_0\rho_0 + \ldots + \rho_k\rho_k$.

Unconventional query models

[A, Hamburg, Unruh, 2019]

- Black-box input: $x_1, \ldots, x_N$.
- Find $i{:}x_i=1$.
- Query splits the state $\alpha_1|1\rangle + \cdots + \alpha_n|n\rangle$ into $|i\rangle$ and $\alpha_1|1\rangle + \cdots + \alpha_{i-1}|i-1\rangle + \alpha_{i+1}|i+1\rangle + \cdots + \alpha_n|n\rangle$.

- Pr[find i: $x_i = 1$] $\leq 4t/N$, t – number of queries.

[A, Rosmanis, Unruh, 2014]

- Black-box input: $x_1, \ldots, x_N$.
- k values $i{:}x_i=1$.
- Access to:
  - One copy of $|\psi_{start}\rangle = \sum_{i:x_i=1} |i\rangle$.
  - D $|\psi_{start}\rangle = - |\psi_{start}\rangle$.
  - Q$|i\rangle = - |i\rangle$ for $i{:}x_i = 1$.
- Task: find two values $i{:}x_i = 1$.

- k values $i:x_i=1$.
- Access to:
  - One copy of $|\psi_{start}\rangle = \sum_{i:x_i=1} |i\rangle$.
  - D $|\psi_{start}\rangle = - |\psi_{start}\rangle$.
  - Q $|i\rangle = - |i\rangle$ for $i:x_i = 1$.
- Task: find two values $i:x_i = 1$.

- Theorem $\Omega(\sqrt{k} + \sqrt{N/k})$ applications of D and Q are necessary.

Questions???