Quantum algorithms

Andrew Childs
Joint Center for Quantum Information and Computer Science
University of Maryland

EQuaLS 2020
3–6 November 2020

Based on slides prepared with Pawel Wocjan

Outline

I. Quantum circuits
II. Elementary quantum algorithms
III. The QFT and phase estimation
IV. Factoring
V. Quantum search
VI. Quantum walk
VII. Adversary lower bounds

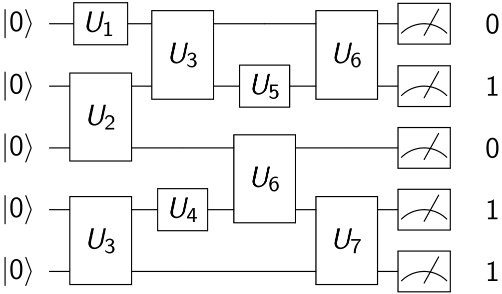
Part I

Quantum circuits

Quantum circuits

Quantum circuits are generalizations of Boolean circuits

input transformation output (probabilistic)



Quantum circuit model

To quantify complexity, a quantum algorithm must be implemented by a quantum circuit, i.e., a sequence of elementary gates

$$\text{Quantum circuit model} \quad = \quad \begin{array}{c} \text{Quantum mechanics} \\ + \\ \text{Notion of complexity} \end{array}$$

A universal gate set

Every unitary can be implemented exactly by quantum circuits using only

- CNOT gates (acting on adjacent qubits) and
- arbitrary single qubit gates

The gate complexity $\kappa(U)$ of a unitary $U \in \mathcal{U}(\mathcal{H})$ is the minimal number of elementary gates needed to implement U

Example: quantum Fourier transform has gate complexity $O(n^2)$

(Approximately) universal gate sets

For every $\epsilon \in (0,1)$ and every unitary U, there is a unitary V such that

$$\|U - V\| \leq \epsilon \quad \text{where} \quad \|U - V\| = \sup_{|\psi\rangle} \|(U - V)|\psi\rangle\|$$

where V is implemented by a quantum circuits using only

- CNOT gates (acting on adjacent qubits)
- the single-qubit gates $H, R(\frac{\pi}{4})$ where

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \quad R(\theta) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{pmatrix}$$

There are other universal gate sets

Gate complexity of unitaries

The gate complexity $\kappa_\epsilon(U)$ of a unitary U is the minimal number of gates (from a universal gate set) need to implement a unitary V with $\|U - V\| \leq \epsilon$

The Solovay-Kitaev theorem implies that

$$\kappa_\epsilon(U) = O\Big(\kappa(U) \cdot \log^c \big(\kappa(U)/\epsilon\big)\Big)$$

for some small constant c

Counting arguments show that most n-qubit unitaries have gate complexity exponential in n

Structure of quantum algorithms

An efficient quantum algorithm consists of

- preparing the initial state $|0\rangle^{\otimes n}$,
- applying a quantum circuit of polynomially many in n gates from some universal gate set, and
- measuring all qubits in the computational basis

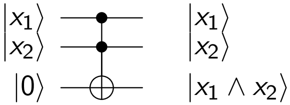
These steps can be repeated polynomially many times to collect statistics, followed by efficient classical post-processing

Reversible computing

The classical AND gate is irreversible because if the output is 0 then we cannot determine which of the three possible pairs was the actual input

x_1	x_2	$x_1 \wedge x_2$
0	0	0
0	1	0
1	0	0
1	1	1

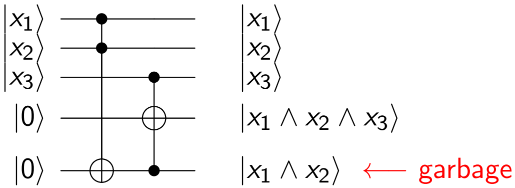
But it is easy to simulate the AND gate with one Toffoli gate



Problem of garbage

To simulate irreversible circuits with Toffoli gates, we keep the input and intermediate results to make everything reversible

Consider the function $y = x_1 \wedge x_2 \wedge x_3$

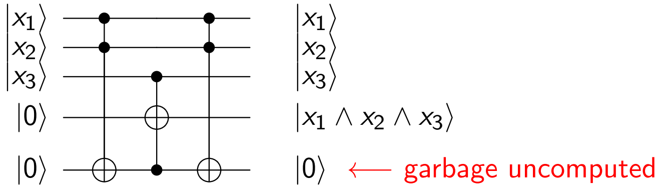


It is important to not leave any garbage; otherwise, we could not make use of quantum parallelism and constructive interference effects

Reversible garbage removal

It is always possible to reversibly remove (uncompute) the garbage

In the case $y = x_1 \wedge x_2 \wedge x_3$, this can be done with the circuit



Simulating irreversible circuits

Let $f : \{0,1\}^n \to \{0,1\}$ be any boolean function

Assume this function can be computed classically using only t classical elementary gates such as AND, OR, NAND

We can implement a unitary U_f on $(\mathbb{C}^2)^{\otimes n} \otimes \mathbb{C}^2 \otimes (\mathbb{C}^2)^{\otimes w}$ such that

$$U_f\left(|x\rangle_{\text{in}} \otimes |y\rangle_{\text{out}} \otimes |0\rangle_{\text{work}}^{\otimes w}\right) = |x\rangle \otimes |y \oplus f(x)\rangle \otimes |0\rangle^{\otimes w}$$

U_f is built from polynomially many in t Toffoli gates and the size w of the workspace register is polynomial in t

During the computation the qubits of the workspace register are changed, but at the end they reversibly reset to $|0\rangle^{\otimes w}$

Part II

Elementary quantum algorithms

Black box problems

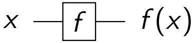
Standard computational problem: determine a property of some input data

- Example: Find the prime factors of N

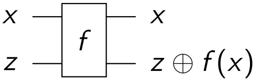
Alternate model: Input is provided by a *black box* (or *oracle*)

- Query: On input x, black box returns $f(x)$
- Determine a property of f using as few queries as possible
- The minimum number of queries is the *query complexity*
- Example: Given a black box for $f : \{1, 2, \dots, N\} \to \{0, 1\}$, is there some x such that $f(x) = 1$?
- Why black boxes?
  - Facilitates proving lower bounds
  - Can lead to algorithms for standard problems

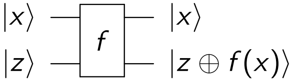
Black boxes for reversible/quantum computing

Black box 

is not reversible

Reversible version: 

Given a circuit that computes f non-reversibly, we can implement the reversible version with little overhead

Quantum version: 

A reversible circuit is a quantum circuit

Deutsch's problem

Problem

- Given: a black-box function $f : \{0,1\} \to \{0,1\}$
- Task: determine whether f is *constant* or *balanced*

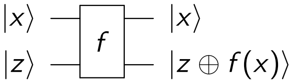
$$\underbrace{\begin{array}{c|c} x & f_1(x) \\ \hline 0 & 0 \\ 1 & 0 \end{array} \quad \begin{array}{c|c} x & f_2(x) \\ \hline 0 & 1 \\ 1 & 1 \end{array}}_{\text{constant: } f(0) = f(1)} \quad \underbrace{\begin{array}{c|c} x & f_3(x) \\ \hline 0 & 0 \\ 1 & 1 \end{array} \quad \begin{array}{c|c} x & f_4(x) \\ \hline 0 & 1 \\ 1 & 0 \end{array}}_{\text{balanced: } f(0) \neq f(1)}$$

How many queries are needed?

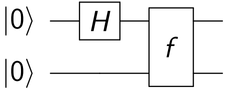
- Classically: 2 queries are necessary and sufficient
- Quantumly: ?

Toward a quantum algorithm for Deutsch's problem

Quantum black box for f:



Compute f in superposition:

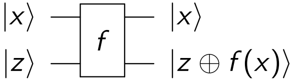


$$
\begin{aligned}
|0\rangle \otimes |0\rangle &\mapsto \frac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes |0\rangle \\
&\mapsto \frac{1}{\sqrt{2}}(|0\rangle \otimes |f(0)\rangle + |1\rangle \otimes |f(1)\rangle)
\end{aligned}
$$

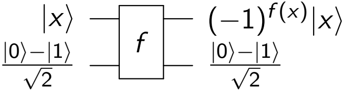
Can't extract more than one bit of information about f

Phase kickback

Quantum black box for f:

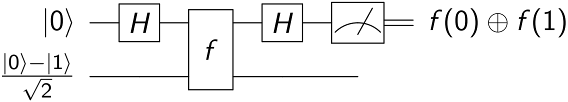


Phase kickback:



$$
\begin{aligned}
|x\rangle \otimes \tfrac{1}{\sqrt{2}}(|0\rangle - |1\rangle) &= \tfrac{1}{\sqrt{2}}(|x\rangle \otimes |0\rangle - |x\rangle \otimes |1\rangle) \\
&\mapsto \tfrac{1}{\sqrt{2}}(|x\rangle \otimes |f(x)\rangle - |x\rangle \otimes |1 \oplus f(x)\rangle) \\
&= |x\rangle \otimes \tfrac{1}{\sqrt{2}}(|f(x)\rangle - |\overline{f(x)}\rangle) \\
&= \underbrace{(-1)^{f(x)}}_{\text{not necessarily global}} |x\rangle \otimes \tfrac{1}{\sqrt{2}}(|0\rangle - |1\rangle)
\end{aligned}
$$

Quantum algorithm for Deutsch's problem



$$
\begin{aligned}
|0\rangle \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}} &\mapsto \frac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}} \\
&\mapsto \frac{(-1)^{f(0)}|0\rangle + (-1)^{f(1)}|1\rangle}{\sqrt{2}} \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}} \\
&= (-1)^{f(0)} \frac{|0\rangle + (-1)^{f(0)\oplus f(1)}|1\rangle}{\sqrt{2}} \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}} \\
&\mapsto (-1)^{f(0)} |f(0) \oplus f(1)\rangle \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}
\end{aligned}
$$

1 quantum query vs. 2 classical queries!

The Deutsch-Jozsa problem

Problem

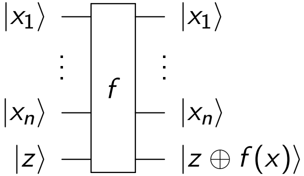
- Given: a black-box function $f : \{0,1\}^n \to \{0,1\}$
- Promise: f is either
  constant ($f(x)$ is independent of x)
  or *balanced* ($f(x) = 0$ for exactly half the values of x)
- Task: determine whether f is constant or balanced

How many queries are needed?

- Classically: $2^n/2 + 1$ queries to answer with certainty
- Quantumly: ?

Phase kickback for a Boolean function of n bits

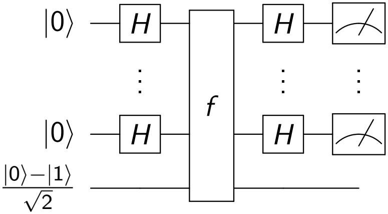
Black box function:



Phase kickback:

$$|x_1\rangle \otimes \cdots \otimes |x_n\rangle \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}} \mapsto (-1)^{f(x)} |x_1\rangle \otimes \cdots \otimes |x_n\rangle \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

Quantum algorithm for the Deutsch-Jozsa problem



$$
\begin{aligned}
|0\rangle^{\otimes n} \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}} &\mapsto \left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right)^{\otimes n} \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}} \\
&= \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}} \\
&\mapsto \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} (-1)^{f(x)} |x\rangle \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}
\end{aligned}
$$

Hadamard transform

What do the final Hadamard gates do?

$$
\begin{aligned}
H|x\rangle &= \frac{1}{\sqrt{2}}(|0\rangle + (-1)^x|1\rangle) \\
&= \frac{1}{\sqrt{2}} \sum_{y\in\{0,1\}} (-1)^{xy}|y\rangle
\end{aligned}
$$

$$
\begin{aligned}
H^{\otimes n}(|x_1\rangle \otimes \cdots \otimes |x_n\rangle) &= \bigotimes_{i=1}^{n} \left(\frac{1}{\sqrt{2}} \sum_{y_i\in\{0,1\}} (-1)^{x_i y_i}|y_i\rangle \right) \\
&= \frac{1}{\sqrt{2^n}} \sum_{y\in\{0,1\}^n} (-1)^{x\cdot y}|y\rangle
\end{aligned}
$$

Quantum D-J algorithm: Finishing up

$$\frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} (-1)^{f(x)} |x\rangle \overset{H^{\otimes n}}{\mapsto} \frac{1}{2^n} \sum_{x,y \in \{0,1\}^n} (-1)^{f(x)} (-1)^{x \cdot y} |y\rangle$$

- If f is constant, the amplitude of $|y\rangle$ is

$$\pm \frac{1}{2^n} \sum_{x \in \{0,1\}^n} (-1)^{x \cdot y} = \pm \begin{cases} 1 & \text{if } y = 0 \ldots 0 \\ 0 & \text{otherwise} \end{cases}$$

  so we definitely measure $0 \ldots 0$

- If f is balanced, the amplitude of $|0 \ldots 0\rangle$ is

$$\sum_{x \in \{0,1\}^n} (-1)^{f(x)} = 0$$

  so we measure some nonzero string

The Deutsch-Jozsa problem: Quantum vs. classical

Above quantum algorithm uses only one query.

Need $2^n/2 + 1$ classical queries to answer with certainty.

What about randomized algorithms? Success probability arbitrarily close to 1 with a constant number of queries.

Can we get a separation between randomized and quantum computation?

Simon's problem

Problem

- Given: a black-box function $f : \{0,1\}^n \to \{0,1\}^m$
- Promise: there is some $s \in \{0,1\}^n$ such that $f(x) = f(y)$ if and only if $x = y$ or $x = y \oplus s$
- Task: determine s

One classical strategy:

- query a random x
- repeat until we find $x_i \neq x_j$ such that $f(x_i) = f(x_j)$
- output $x_i \oplus x_j$

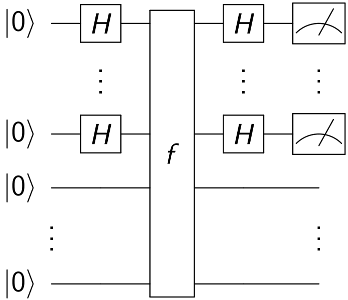
By the birthday problem, this uses about $\sqrt{2^n}$ queries.

It can be shown that this strategy is essentially optimal.

Quantum algorithm for Simon's problem

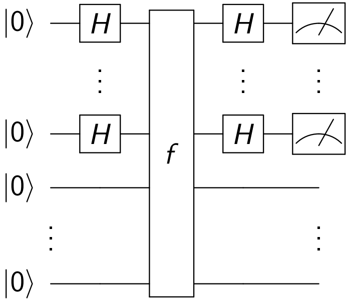
Quantum black box: $|x\rangle \otimes |y\rangle \mapsto |x\rangle \otimes |y \oplus f(x)\rangle$

$$(x \in \{0,1\}^n,\ y \in \{0,1\}^m)$$



Repeat many times and post-process the measurement outcomes

Quantum algorithm for Simon's problem: Analysis I



$$
\begin{aligned}
&|0\rangle^{\otimes n} \otimes |0\rangle^{\otimes m} \\
&\mapsto \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle \otimes |0\rangle^{\otimes m} \\
&\mapsto \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle \otimes |f(x)\rangle \\
&= \frac{1}{\sqrt{2^{n-1}}} \sum_{x \in R} \frac{|x\rangle + |x \oplus s\rangle}{\sqrt{2}} \otimes |f(x)\rangle
\end{aligned}
$$

for some $R \subset \{0,1\}^n$

Quantum algorithm for Simon's problem: Analysis II

Recall $H^{\otimes n}|x\rangle = \sum_{y\in\{0,1\}^n}(-1)^{x\cdot y}|y\rangle$

$$H^{\otimes n}\left(\frac{|x\rangle + |x\oplus s\rangle}{\sqrt{2}}\right) = \frac{1}{\sqrt{2^{n+1}}}\sum_{y\in\{0,1\}^n}[(-1)^{x\cdot y} + (-1)^{(x\oplus s)\cdot y}]|y\rangle$$
$$= \frac{1}{\sqrt{2^{n+1}}}\sum_{y\in\{0,1\}^n}(-1)^{x\cdot y}[1 + (-1)^{s\cdot y}]|y\rangle$$

Two cases:

- if $s\cdot y = 0 \bmod 2$, $1+(-1)^{s\cdot y} = 2$
- if $s\cdot y = 1 \bmod 2$, $1+(-1)^{s\cdot y} = 0$

Measuring gives a random y orthogonal to s (i.e., $s\cdot y = 0$)

Quantum algorithm for Simon's problem: Post-processing

Measuring gives a random y orthogonal to s ($s \cdot y = 0$)

Repeat k times, giving vectors $y_1, \ldots, y_k \in \{0,1\}^n$; solve a system of k linear equations for $s \in \{0,1\}^n$:

$$y_1 \cdot s = 0, \quad y_2 \cdot s = 0, \quad \ldots, \quad y_k \cdot s = 0$$

How big should k be to give a unique (nonzero) solution?

- Clearly $k \geq n - 1$ is necessary
- It can be shown that $k = O(n)$ suffices

$O(n)$ quantum queries, $O(n^3)$ quantum gates

Compare to $\Omega(2^{n/2})$ classical queries (even for bounded error)

Recap

We have seen several examples of quantum algorithms that outperform classical computation:

- Deutsch's problem: 1 quantum query vs. 2 classical queries
- Deutsch-Jozsa problem: 1 quantum query vs. $2^{\Omega(n)}$ classical queries (deterministic)
- Simon's problem: $O(n)$ quantum queries vs. $2^{\Omega(n)}$ classical queries (randomized)

Quantum algorithms for more interesting problems build on the tools used in these examples.

Part III

The QFT and phase estimation

Quantum phase estimation

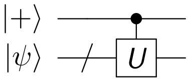
Problem

We are given a unitary U and an eigenvector $|\psi\rangle$ of U with unknown eigenvalue

We seek to estimate its eigenphase $\varphi \in [0, 1)$ such that

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

Phase kickback for U



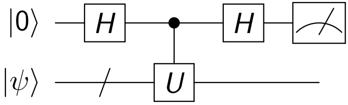
$$\frac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes |\psi\rangle \mapsto \frac{|0\rangle + e^{2\pi i \varphi}|1\rangle}{\sqrt{2}} \otimes |\psi\rangle$$

The eigenstate $|\psi\rangle$ in the target register emerges unchanged

$\Rightarrow$ It suffices to focus on the control register

The state $|0\rangle + |1\rangle$ of the control qubit is changed to $|0\rangle + e^{2\pi i \varphi}|1\rangle$

Hadamard test



$$
\begin{aligned}
& \frac{|0\rangle + e^{2\pi i\varphi}|1\rangle}{\sqrt{2}} \\
\mapsto \quad & \frac{1}{2}\big((|0\rangle + |1\rangle) + e^{2\pi i\varphi}(|0\rangle - |1\rangle)\big) \\
= \quad & \frac{1}{2}\big((1 + e^{2\pi i\varphi})|0\rangle + (1 - e^{2\pi i\varphi})|1\rangle)\big)
\end{aligned}
$$

Hadamard test

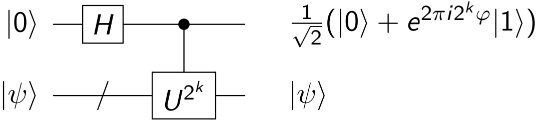
$$\frac{1}{2}\left((1+e^{2\pi i\varphi})|0\rangle+(1-e^{2\pi i\varphi})|1\rangle)\right)$$

The probability of obtaining 0 is

$$\begin{aligned}
\Pr(0) &= |\langle 0|\varphi\rangle|^2 \\
&= |\frac{1}{2}(1+e^{2\pi i\varphi})|^2 \\
&= \frac{1}{4}|e^{\pi i\varphi}+e^{-\pi i\varphi}|^2 \\
&= \frac{1}{4}|2\cos(\pi\varphi)|^2 \\
&= \cos^2(\pi\varphi)
\end{aligned}$$

Phase kickback due to higher powers of U

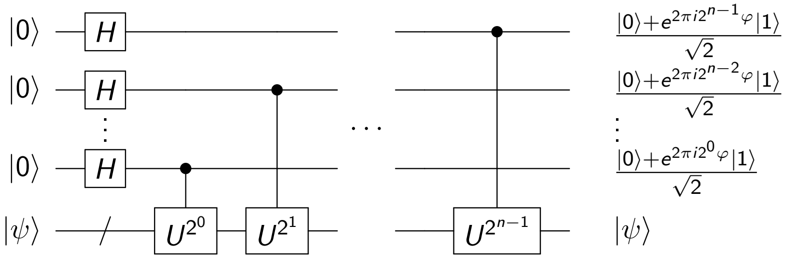
For arbitrary k, we obtain



since

$$U^{2^k}|\psi\rangle = e^{2\pi i 2^k \varphi}|\psi\rangle$$

Phase kickback part of phase estimation



We set

$$|\varphi\rangle := \frac{|0\rangle + e^{2\pi i 2^{n-1}\varphi}|1\rangle}{\sqrt{2}} \otimes \frac{|0\rangle + e^{2\pi i 2^{n-2}\varphi}|1\rangle}{\sqrt{2}} \otimes \cdots \otimes \frac{|0\rangle + e^{2\pi i 2^{0}\varphi}|1\rangle}{\sqrt{2}}$$

Binary fractions

Assume that the eigenphase φ is an exact n-bit binary fraction, i.e.,

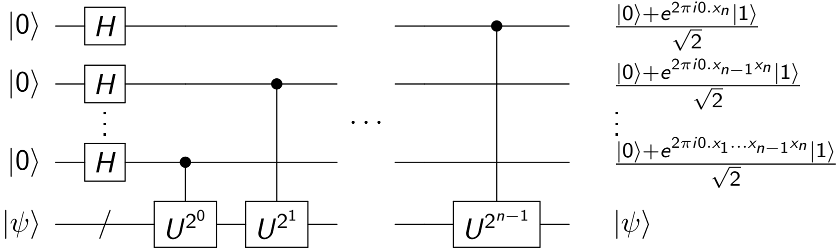
$$\varphi = 0.x_1 x_2 \ldots x_n = \sum_{i=1}^{n} \frac{x_i}{2^i}$$

For $k \in \{0, \ldots, n-1\}$, we have

$$2^k \varphi = x_1 x_2 \ldots x_k . x_{k+1} \ldots x_n$$

$$\begin{aligned}
e^{2\pi i 2^k \varphi} &= e^{2\pi i (x_1 x_2 \ldots x_k . x_{k+1} \ldots x_n)} \\
&= e^{2\pi i (x_1 x_2 \ldots x_k + 0.x_{k+1} \ldots x_n)} \\
&= e^{2\pi i (x_1 x_2 \ldots x_k)} \cdot e^{2\pi i (0.x_{k+1} \ldots x_n)} \\
&= e^{2\pi i (0.x_{k+1} \ldots x_n)}
\end{aligned}$$

Phase kickback part of phase estimation



Quantum Fourier transform

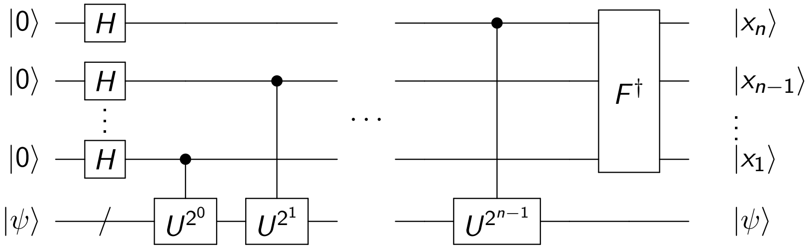
The quantum Fourier transform F is defined by

$$F\big(|x_n\rangle \otimes |x_{n-1}\rangle \otimes \cdots \otimes |x_1\rangle\big)$$

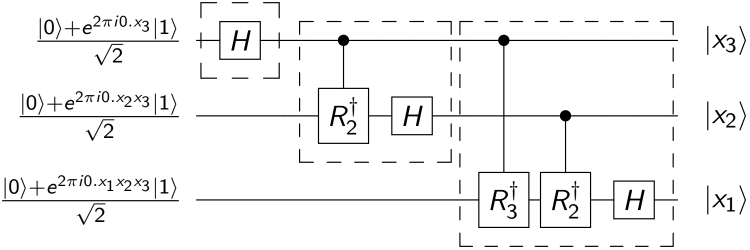
$$= \quad \frac{|0\rangle + e^{2\pi i 0.x_n}|1\rangle}{\sqrt{2}} \otimes \frac{|0\rangle + e^{2\pi i 0.x_{n-1}x_n}|1\rangle}{\sqrt{2}} \otimes \cdots \otimes \frac{|0\rangle + e^{2\pi i 0.x_1 x_2 \cdots x_n}|1\rangle}{\sqrt{2}}$$

Use inverse quantum Fourier transform $F^\dagger$ to obtain the bits of the eigenphase

Quantum circuit for phase estimation



Inverse quantum Fourier transform for 3 bits



The phase shift R_k is defined by

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

Summary of phase estimation circuit

We use phase kick back due to the controlled U^{2^k} gate to prepare the state

$$\frac{|0\rangle + e^{2\pi i 0.x_{k+1}x_{k+2}\ldots x_n}|1\rangle}{\sqrt{2}}$$

Using the previously determined bits $x_{k+2}, \ldots, x_n$, we change this state to

$$\frac{|0\rangle + e^{2\pi i 0.x_{k+1}0\ldots 0}|1\rangle}{\sqrt{2}} = \frac{|0\rangle + (-1)^{x_k}|1\rangle}{\sqrt{2}}$$

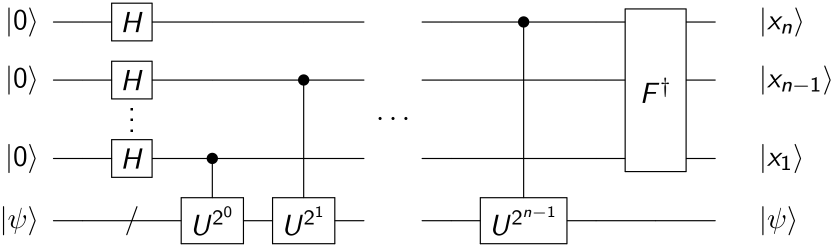
We apply the Hadamard gate to obtain

$$|x_{k+1}\rangle$$

The controlled phase shifts enable us to reduce the problem of determining each bit to distinguishing between $|+\rangle$ and $|-\rangle$ (deterministic Hadamard test)

Special case: exact n-bit binary fraction

Assume that φ is an exact n-bit binary fraction, i.e., $\varphi = 0.x_1 \ldots x_{n-1}x_n$



$\Rightarrow$ The measurment of the qubits yields the bits $x_n, x_{n-1}, \ldots, x_1$ deterministically

General case: arbitrary eigenphases

Let φ be arbitrary

Unless φ is an exact n-bit fraction, the application of the inverse quantum Fourier transform

$$F^{\dagger}|\varphi\rangle$$

produces a superposition of n-bit strings

Probability of obtaining a certain estimate

Lemma

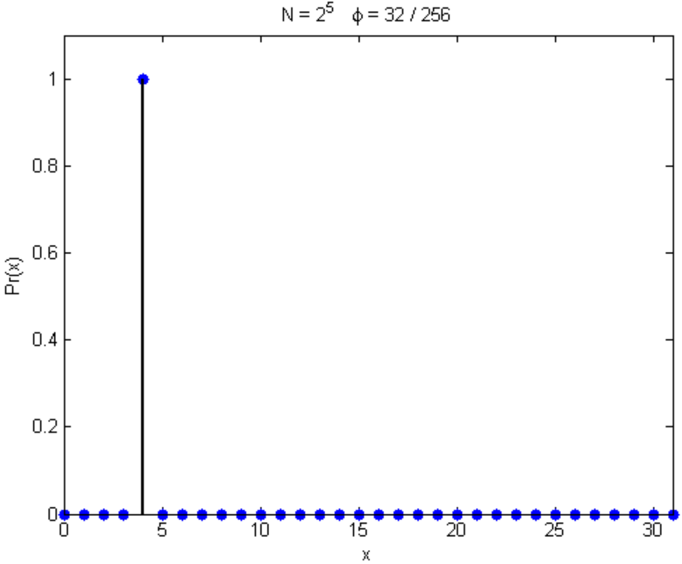
Let $x = \sum_{k=1}^{n} x_i 2^{n-i}$ *and* $\varphi_x := 0.x_1 x_2 \ldots x_n = \frac{x}{2^n}$ *be the corresponding n-bit fraction*

The probability of obtaining the estimate φ_x *when the true eigenphase is* φ *is*

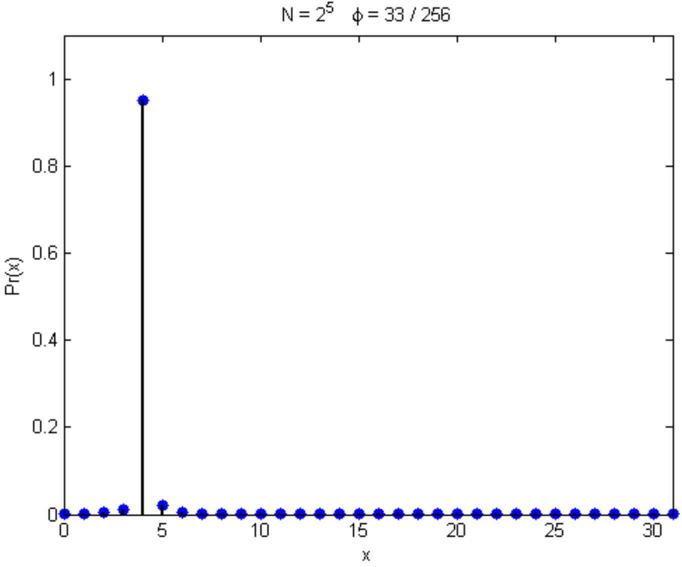
$$\Pr(x) = \frac{1}{2^{2n}} \frac{\sin^2\left(2^n \pi \left(\varphi - \varphi_x\right)\right)}{\sin^2\left(\pi \left(\varphi - \varphi_x\right)\right)}$$

This distribution is peaked around the true value

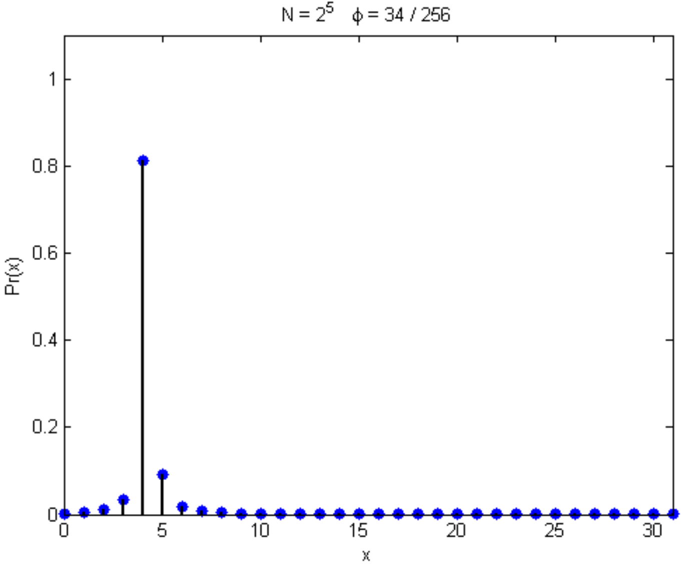
Examples of probability distributions for different φ



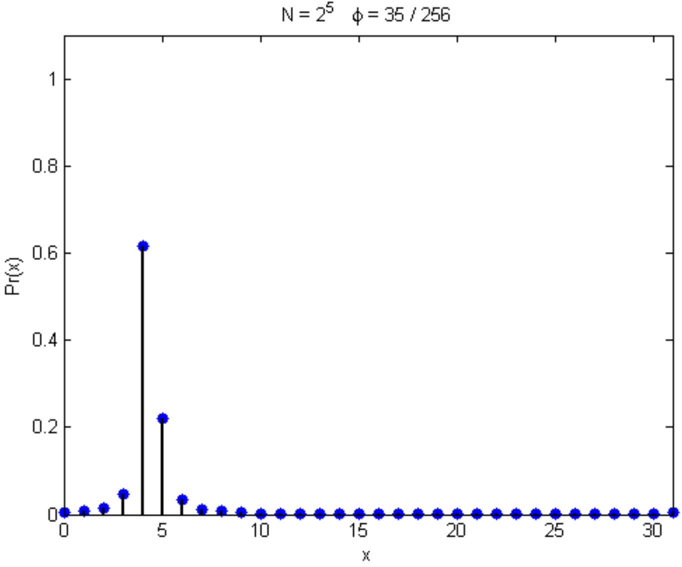
Examples of probability distributions for different φ



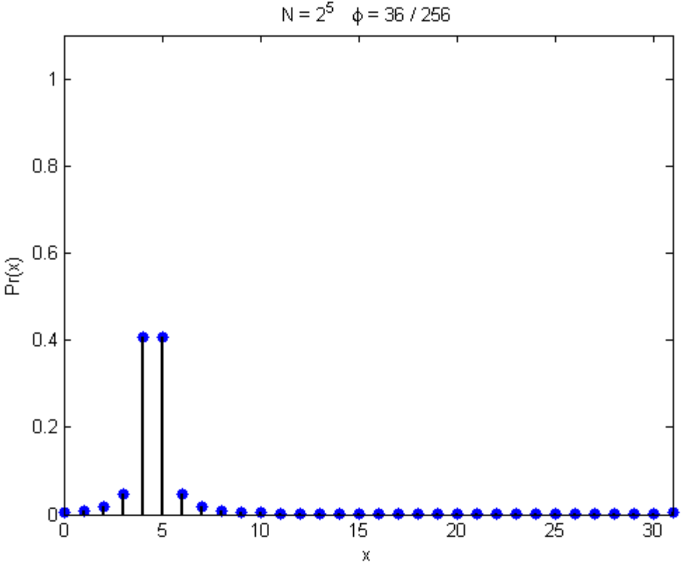
Examples of probability distributions for different φ



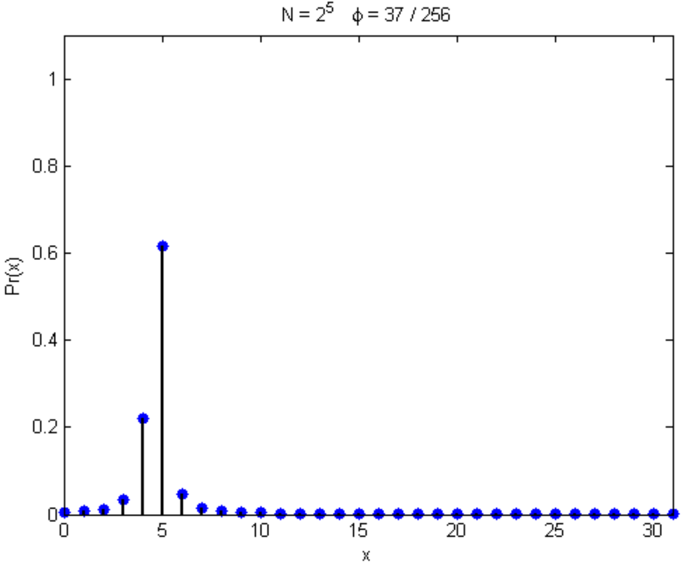
Examples of probability distributions for different φ



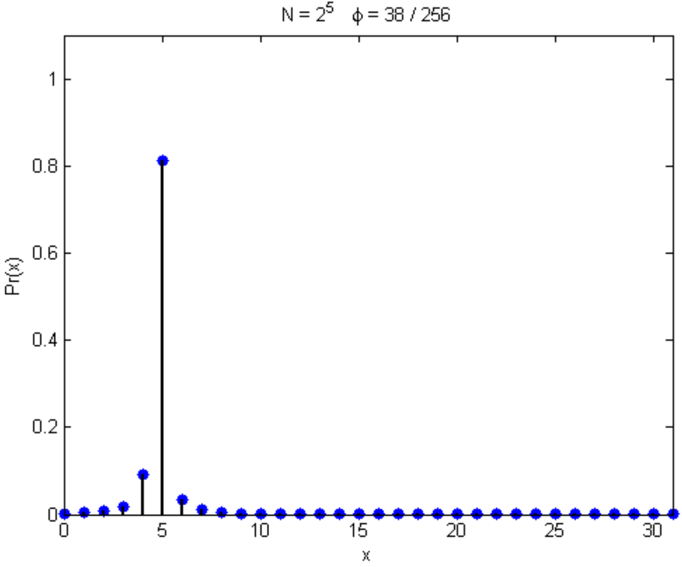
Examples of probability distributions for different φ



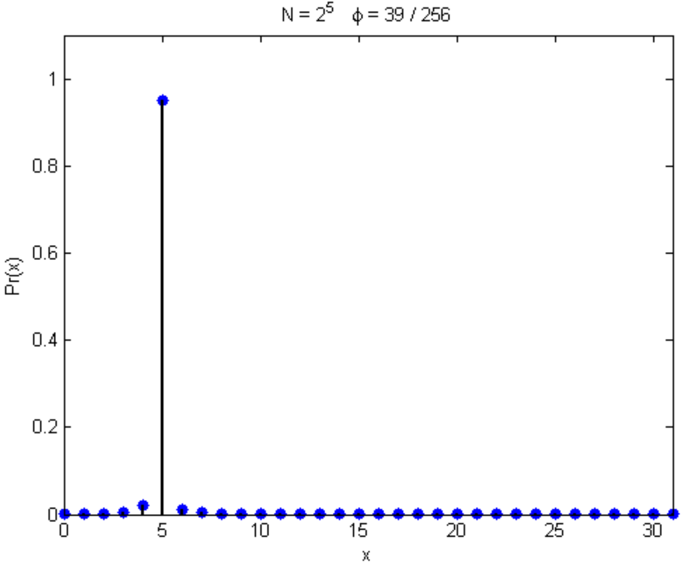
Examples of probability distributions for different φ



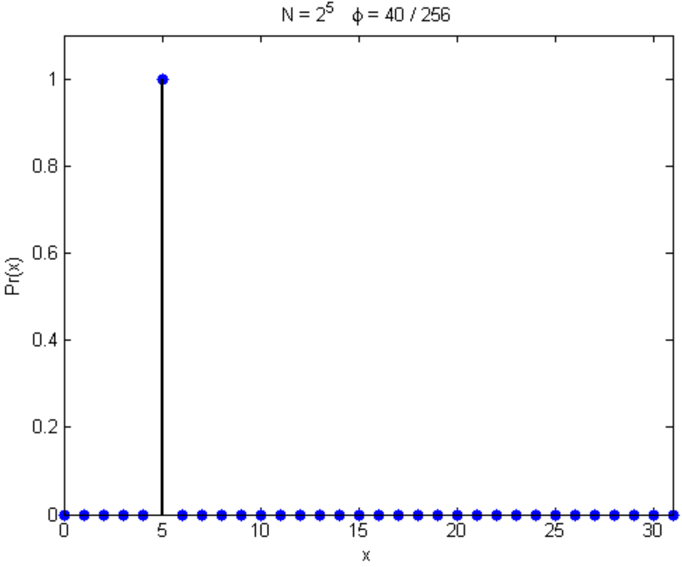
Examples of probability distributions for different φ



Examples of probability distributions for different φ



Examples of probability distributions for different φ



Lower bound on success probability

Theorem

Let x be such that $\frac{x}{2^n} \leq \varphi < \frac{x+1}{2^n}$

The probability of returning one of the two closest n-bit fractions φ_x and φ_{x+1} is at least $\frac{8}{\pi^2}$

Summary of phase estimation

We are given a unitary U and an eigenvector $|\psi\rangle$ of U with unknown eigenphase φ

We obtain an estimate $\hat{\varphi}$ such that

$$\Pr\left(|\hat{\varphi} - \varphi| \leq \frac{1}{2^n}\right) \geq \frac{8}{\pi^2}$$

To do this, we need invoke each of the controlled U, $U^2, \ldots, U^{2^{n-1}}$ gates once

We can boost the success probability to $1 - \epsilon$ by repeating the above algorithm $O(\log(1/\epsilon))$ times and outputting the median of the outcomes

Phase estimation applied to superpositions of eigenstates

We are given a unitary U with eigenvectors $|\psi_i\rangle$ and corresponding eigenphases φ_i

Let

$$|\psi\rangle = \sum_i \alpha_i |\psi_i\rangle$$

What happens if we apply phase estimation to $|0\rangle^{\otimes n} \otimes |\psi\rangle$?

After the n phase kickbacks due to U^{2^0}, U^{2^1}, $\ldots$ $U^{2^{n-1}}$, we obtain

$$\sum_i \alpha_i |\varphi_i\rangle \otimes |\psi_i\rangle$$

After applying the inverse quantum Fourier transform, we obtain

$$\sum_i \alpha_i |\tilde{x}_i\rangle \otimes |\psi_i\rangle$$

where $|\tilde{x}_i\rangle$ denotes a superpositions of n-bit estimates of φ_i

Part IV

Factoring

The fundamental theorem of arithmetic

Theorem

Every positive integer larger than 1 can be factored as a product of prime numbers, and this factorization is unique (up to the order of the factors).

$$N = 2^{n_2} \times 3^{n_3} \times 5^{n_5} \times 7^{n_7} \times \cdots$$

Examples

$$15 = 3 \times 5$$

$$239815173914273 = 15485863 \times 15486071$$

$$\begin{array}{r} 3107418240490043721350750 \\ 0358885679300373460228427 \\ 2754572016194882320644051 \\ 8081504556346829671723286 \\ 7824379162728380334154710 \\ 7310850191954852900733772 \\ 4822783525742386454014691 \\ 736602477652346609 \end{array} = \begin{array}{l} 1634733645809253848 \\ 4313388386509085984 \\ 1783670033092312181 \\ 1108523893331001045 \\ 0815121211816751157 \\ 9 \\ \times \\ 1900871281664822113 \\ 1268515739354139754 \\ 7189678996851549366 \\ 6638539088027103802 \\ 1044989571912614655 \\ 71 \end{array}$$

Why care about factoring?

"The problem of distinguishing prime numbers from composite numbers and of resolving the latter into their prime factors is known to be one of the most important and useful in arithmetic. It has engaged the industry and wisdom of ancient and modern geometers to such an extent that it would be superfluous to discuss the problem at length... Further, the dignity of the science itself seems to require that every possible means be explored for the solution of a problem so elegant and so celebrated."

– Carl Friedrich Gauss, Disquisitiones Arithmeticæ (1801)

More practically: The presumed hardness of factoring is the basis of much of modern cryptography (RSA cryptosystem)

Order finding

Definition

Given $a, N \in \mathbb{Z}$ with $\gcd(a, N) = 1$, the *order* of a modulo N is the smallest positive integer r such that $a^r \equiv 1 \pmod{N}$.

Problem

- Given: $a, N \in \mathbb{Z}$ with $\gcd(a, N) = 1$
- Task: find the order of a modulo N

Spectrum of a cyclic shift

Let P be a cyclic shift modulo r: $P|x\rangle = |x + 1 \bmod r\rangle$

Claim. For any $k \in \mathbb{Z}$, the state $|u_k\rangle := \frac{1}{\sqrt{r}} \sum_{x=0}^{r-1} e^{-2\pi i k x / r} |x\rangle$ is an eigenstate of P.

Proof.
$$
\begin{aligned}
U|u_k\rangle &= \frac{1}{\sqrt{r}} \sum_{x=0}^{r-1} e^{-2\pi i k x / r} |x + 1 \bmod r\rangle \\
&= \frac{1}{\sqrt{r}} \sum_{x=0}^{r-1} e^{2\pi i k / r} e^{-2\pi i k (x+1) / r} |x + 1 \bmod r\rangle \\
&= e^{2\pi i k / r} \frac{1}{\sqrt{r}} \sum_{x=1}^{r} e^{-2\pi i k x / r} |x \bmod r\rangle \\
&= e^{2\pi i k / r} |u_k\rangle
\end{aligned}
$$
$\square$

The multiplication-by-a map

Define U by $U|x\rangle = |ax\rangle$ for $x \in \mathbb{Z}_N$.

Computing U:

$$\begin{aligned} |x,0\rangle &\mapsto |x,ax\rangle && \text{(reversible multiplication by } a\text{)} \\ &\mapsto |ax,x\rangle && \text{(swap)} \\ &\mapsto |ax,0\rangle && \text{(uncompute reversible division by } a\text{)} \end{aligned}$$

High powers of U can be implemented efficiently using repeated squaring

Spectrum of the multiplication-by-a map

Define U by $U|x\rangle = |ax\rangle$ for $x \in \mathbb{Z}_N$.

Claim. Let r be the order of a modulo N. For any $k \in \mathbb{Z}$, the state

$$|u_k\rangle := \frac{1}{\sqrt{r}} \sum_{x=0}^{r-1} e^{-2\pi i k x / r} |a^x \bmod N\rangle$$

is an eigenstate of U with eigenvalue $e^{2\pi i k / r}$.

Proof.
Same as for the cyclic shift, due to the isomorphism

$$x \bmod r \quad \leftrightarrow \quad a^x \bmod N \qquad \square$$

Order finding and phase estimation

$U|u_k\rangle = e^{2\pi i k/r}|u_k\rangle$

Phase estimation of U on $|u_k\rangle$ can be used to approximate k/r.

Problems:

1. We don't know r, so we can't prepare $|u_k\rangle$.
2. We only get an approximation of k/r.
3. Even if we knew k/r exactly, k and r could have common factors.

Solutions:

1. Estimate k/r for a superposition of the $|u_k\rangle$.
2. Use the continued fraction expansion.
3. Show that $gcd(k, r) = 1$ with reasonable probability.

Estimating k/r in superposition

A useful identity:

$$\sum_{k=0}^{r-1} e^{2\pi i k x/r} = \begin{cases} r & \text{if } x = 0 \\ 0 & \text{otherwise} \end{cases}$$

Consider

$$\frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} |u_k\rangle = \frac{1}{r} \sum_{k,x=0}^{r-1} e^{-2\pi i k x/r} |a^x \bmod N\rangle$$
$$= |a^0 \bmod N\rangle = |1\rangle$$

Phase estimation:

$$|0\rangle \otimes |1\rangle = \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} |0\rangle \otimes |u_k\rangle \mapsto \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} |\widetilde{k/r}\rangle \otimes |u_k\rangle$$

Measurement gives an approximation of k/r for a random k

Continued fractions

Problem

Given samples x of the form $\lfloor k\frac{2^n}{r}\rfloor$, $\lceil k\frac{2^n}{r}\rceil$ ($k \in \{0, 1, \ldots, r-1\}$), determine r.

Continued fraction expansion:

$$\frac{x}{2^n} = \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{a_3 + \cdots}}}$$

Gives an efficiently computable sequence of rational approximations

Theorem

If $2^n \geq N^2$, then k/r is the closest convergent of the CFE to $x/2^n$ among those with denominator smaller than N.

Since $r < N$, it suffices to take $n = 2\log_2 N$

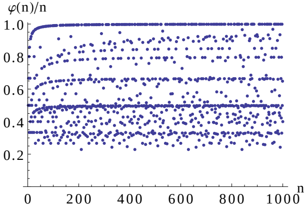
Common factors

If $\gcd(k, r) = 1$, then the denominator of k/r is r

Fact

The probability that $\gcd(k, r) = 1$ *for a random* $k \in \{0, 1, \ldots, r-1\}$ *is*

$$\frac{\phi(r)}{r} = \Omega\left(\frac{1}{\log\log r}\right)$$



Thus $\Omega(\log\log N)$ repetitions suffice to give r with constant probability

Alternatively, find two (or more) denominators and take their least common multiple; then $O(1)$ repetitions suffice

Factoring $\rightarrow$ finding a nontrivial factor

Suppose we want to factor the positive integer N.

Since primality can be tested efficiently, it suffices to give a procedure for finding a nontrivial factor of N with constant probability.

```
function factor(N)

if N is prime
     output N
else
     repeat
          x=find_nontrivial_factor(N)
     until success
     factor(x)
     factor(N/x)
end if
```

We can assume N is odd, since it is easy to find the factor 2.

We can also assume that N contains at least two distinct prime powers, since it is easy to check if it is a power of some integer.

Reduction of factoring to order finding

Factoring N reduces to order finding in $\mathbb{Z}_N^\times$ [Miller 1976].

Choose $a \in \{2, 3, \ldots, N-1\}$ uniformly at random.

If $\gcd(a, N) \neq 1$, then it is a nontrivial factor of N.

If $\gcd(a, N) = 1$, let r denote the order of a modulo N.

Suppose r is even. Then

$$\begin{gathered} a^r = 1 \bmod N \\ \Updownarrow \\ (a^{r/2})^2 - 1 = 0 \bmod N \\ \Updownarrow \\ (a^{r/2} - 1)(a^{r/2} + 1) = 0 \bmod N \end{gathered}$$

so we might hope that $\gcd(a^{r/2} - 1, N)$ is a nontrivial factor of N.

Miller's reduction

Question

Given $(a^{r/2} - 1)(a^{r/2} + 1) = 0 \bmod N$, when does $\gcd(a^{r/2} - 1, N)$ give a nontrivial factor of N?

Note that $a^{r/2} - 1 \neq 0 \bmod N$ (otherwise the order of a would be $r/2$, or smaller).

So it suffices to ensure that $a^{r/2} + 1 \neq 0 \bmod N$.

Lemma

Suppose $a \in \mathbb{Z}_N^{\times}$ is chosen uniformly at random, where N is an odd integer with at least two distinct prime factors. Then with probability at least $1/2$, the order r of a is even and $a^{r/2} \neq -1 \bmod N$.

Shor's algorithm

Input: Integer N
Output: A nontrivial factor of N

1. Choose a random $a \in \{2, 3, \ldots, N-1\}$
2. Compute $\gcd(a, N)$; if it is not 1 then it is a nontrivial factor, and otherwise we continue
3. Perform phase estimation with the multiplication-by-a operator U on the state $|1\rangle$ using $n = 2\log_2 N$ bits of precision
4. Compute the continued fraction expansion of the estimated phase, and find the best approximation with denominator less than N; call the result r
5. Compute $\gcd(a^{r/2} - 1, N)$. If it is a nontrivial factor of N, we are done; if not, go back to step 1

Quantum vs. classical factoring algorithms

Best known classical algorithm for factoring N

- Proven running time: $2^{O((\log N)^{1/2}(\log\log N)^{1/2})}$
- With plausible heuristic assumptions: $2^{O((\log N)^{1/3}(\log\log N)^{1/3})}$

Shor's quantum algorithm

- QFT modulo 2^n with $n = O(\log N)$: takes $O(n^2)$ steps
- Modular exponentiation: compute a^x for $x < 2^n$. With repeated squaring, takes $O(n^3)$ steps
- Running time of Shor's algorithm: $O(\log^3 N)$

Part V

Unstructured search

Unstructured search

Quantum computers can quadratically outperform classical computers at a very basic computational task, unstructured search

There is a set X containing N items, some of which are marked

We are given a Boolean black box $f: X \to \{0,1\}$ that indicates whether a given item is marked

The problem is to decide if any item is marked, or alternatively, to find a marked item given that one exists

Unstructured search as a model for NP

Unstructured search can be thought of as a model for solving problems in NP by brute force search

If a problem is in NP, then we can efficiently recognize a solution, so one way to find a solution is to solve unstructured search

Of course, this may not be the best way to find a solution in general, even if the problem is NP-hard: we don't know if NP-hard problems are really "unstructured"

Classical vs. quantum query complexity

It is obvious that even a randomized classical algorithm needs $\Omega(N)$ queries to decide if any item is marked

But a quantum algorithm can do much better!

Phase oracle

We assume that we have a unitary operator U satisfying

$$U|x\rangle = (-1)^{f(x)}|x\rangle = \begin{cases} |x\rangle & x \text{ is not marked} \\ -|x\rangle & x \text{ is marked} \end{cases}$$

This can be created using one query to a standard reversible oracle via phase kickback

Target state

We consider the case where there is exactly one $x \in X$ element that is marked; call this element m

Our goal is to prepare the state $|m\rangle$

Initial state

We have no information about which item might be marked

Thus we take

$$|\psi\rangle := \frac{1}{\sqrt{N}} \sum_{x=1}^{N} |x\rangle$$

as the initial state

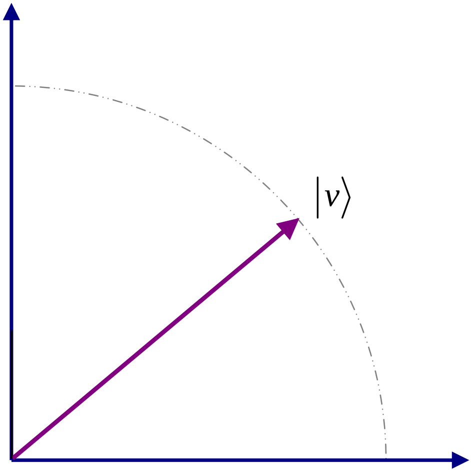
Rough idea behind Grover search

Start with the initial state $|\psi\rangle$

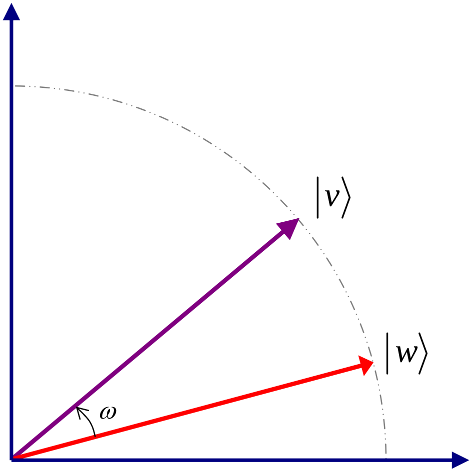
Implement a rotation that moves $|\psi\rangle$ toward $|m\rangle$

Realize the rotation with the help of two reflections

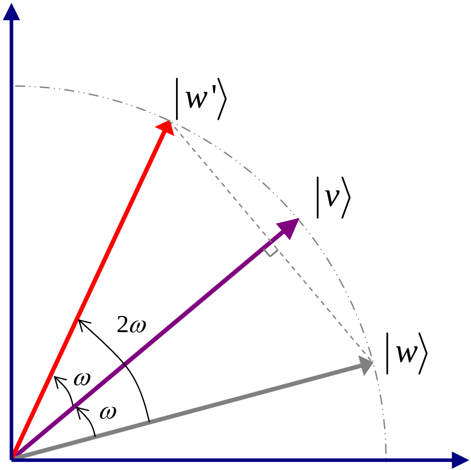
Visualization of a reflection in $\mathbb{R}^2$



Visualization of a reflection in $\mathbb{R}^2$



Visualization of a reflection in $\mathbb{R}^2$



Reflections

$U = I - 2|m\rangle\langle m|$ is the reflection about the target state $|m\rangle$

$V := I - 2|\psi\rangle\langle\psi|$ is the reflection about the initial state $|\psi\rangle$:

$$\begin{aligned} V|\psi\rangle &= -|\psi\rangle \\ V|\psi^\perp\rangle &= \;\;\; |\psi^\perp\rangle \end{aligned}$$

for any state $|\psi^\perp\rangle$ orthogonal to $|\psi\rangle$

Structure of Grover's algorithm

The algorithm is as follows:

- start in $|\psi\rangle$,
- apply the Grover iteration $G := V\,U$ some number of times,
- make a measurement and hope that the outcome is m

Invariant subspace

Observe that $\text{span}\{|m\rangle, |\psi\rangle\}$ is a U- and V-invariant subspace, and both the inital and target states belong to this subspace

$\Rightarrow$ It suffices to understand the restriction of VU to this subspace

Let $\{|m\rangle, |\phi\rangle\}$ be an orthonormal basis for $\text{span}\{|m\rangle, |\psi\rangle\}$

The Gram-Schmidt process yields

$$|\phi\rangle = \frac{|\psi\rangle - \sin\theta |m\rangle}{\cos\theta}$$

where $\sin\theta := \langle m|\psi\rangle = 1/\sqrt{N}$

Invariant subspace

Now in the basis $\{|m\rangle, |\phi\rangle\}$, we have

$$
\begin{aligned}
|\psi\rangle &= \sin\theta|m\rangle + \cos\theta|\phi\rangle \text{ where } \sin\theta = \langle m|\psi\rangle = 1/\sqrt{N} \\
U &= \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} \\
V &= I - 2|\psi\rangle\langle\psi| \\
&= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} - 2\begin{pmatrix} \sin\theta \\ \cos\theta \end{pmatrix}\begin{pmatrix} \sin\theta & \cos\theta \end{pmatrix} \\
&= \begin{pmatrix} 1 - 2\sin^2\theta & -2\sin\theta\cos\theta \\ -2\sin\theta\cos\theta & 1 - 2\cos^2\theta \end{pmatrix} \\
&= -\begin{pmatrix} -\cos 2\theta & \sin 2\theta \\ \sin 2\theta & \cos 2\theta \end{pmatrix}
\end{aligned}
$$

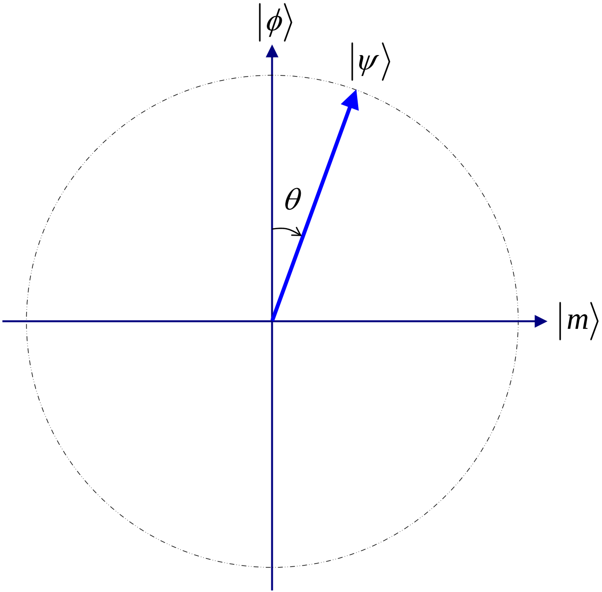
Grover iteration within the invariant subspace

$\Rightarrow$ We find

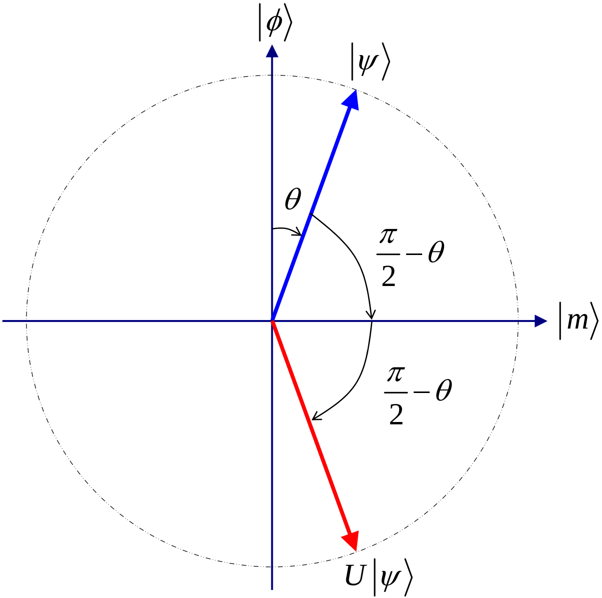
$$
\begin{aligned}
V\,U &= -\begin{pmatrix} -\cos 2\theta & \sin 2\theta \\ \sin 2\theta & \cos 2\theta \end{pmatrix} \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} \\
&= -\begin{pmatrix} \cos 2\theta & \sin 2\theta \\ -\sin 2\theta & \cos 2\theta \end{pmatrix}
\end{aligned}
$$

This is a rotation up to a minus sign

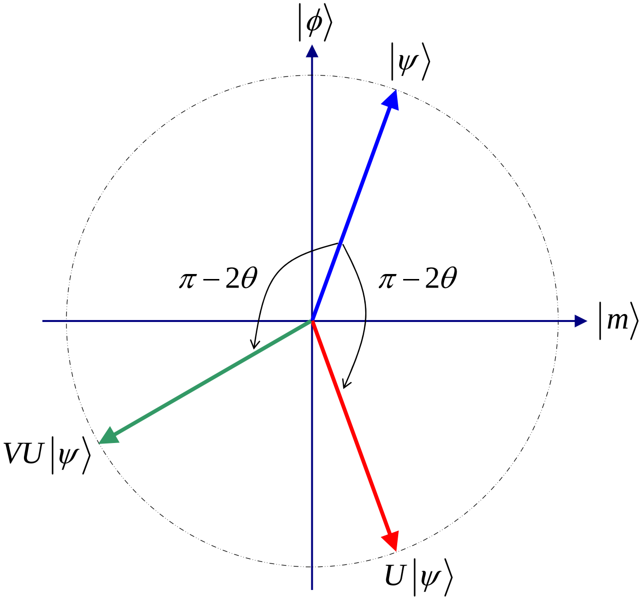
Visualization of first Grover iteration



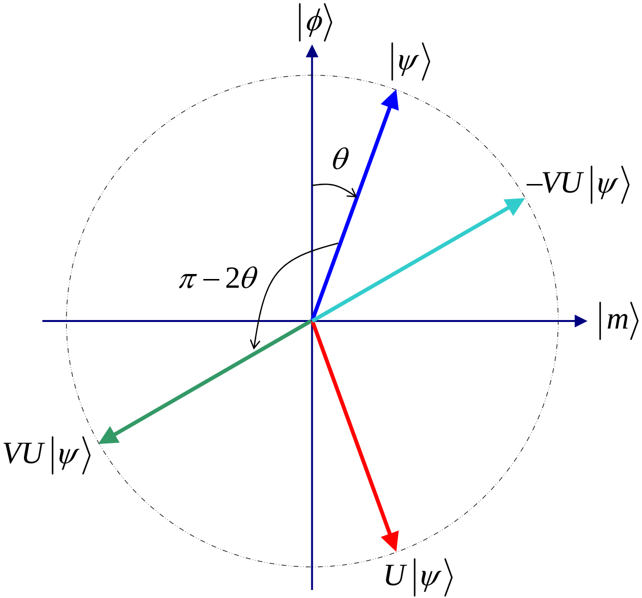
Visualization of first Grover iteration



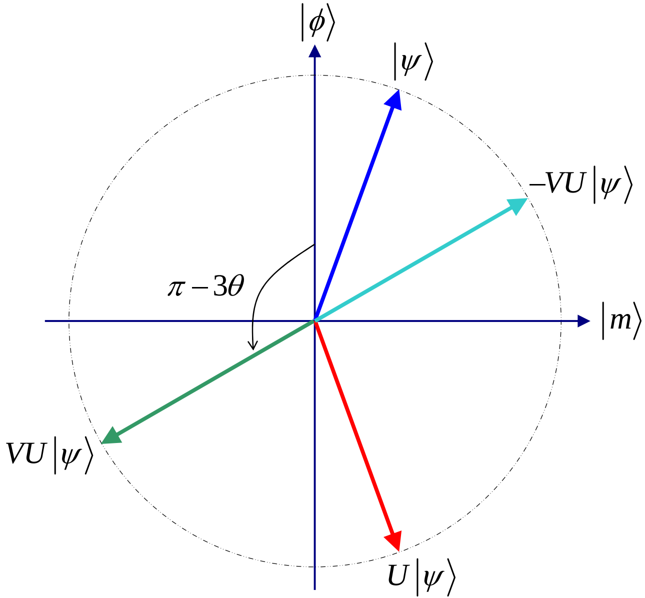
Visualization of first Grover iteration



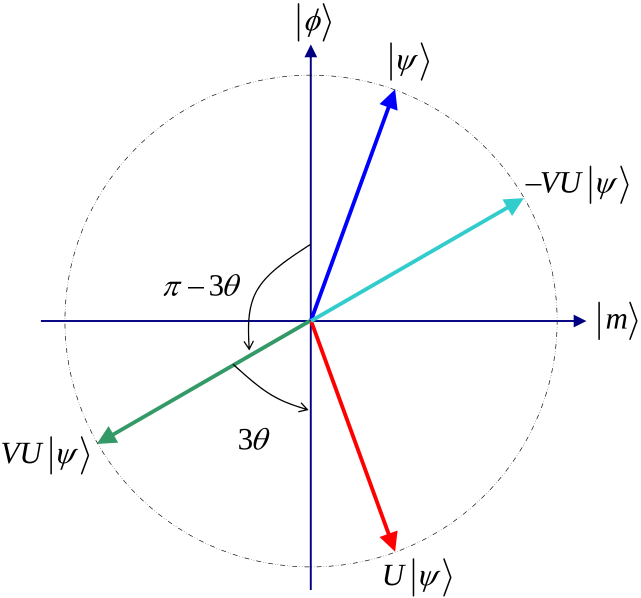
Visualization of first Grover iteration



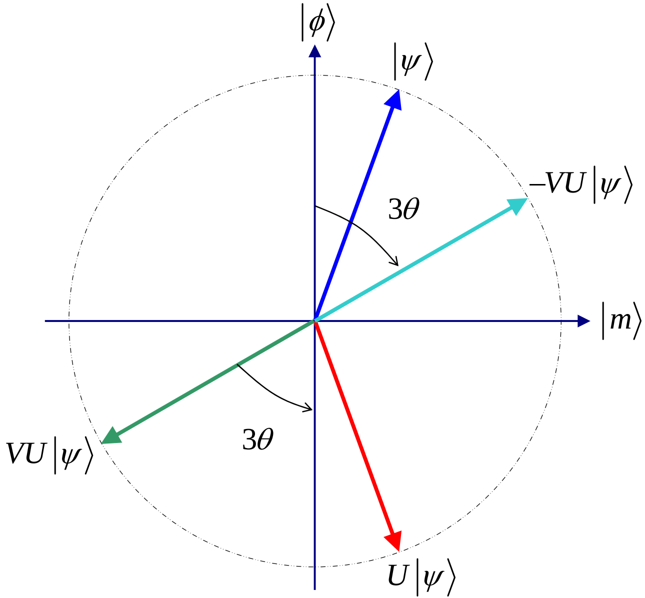
Visualization of first Grover iteration



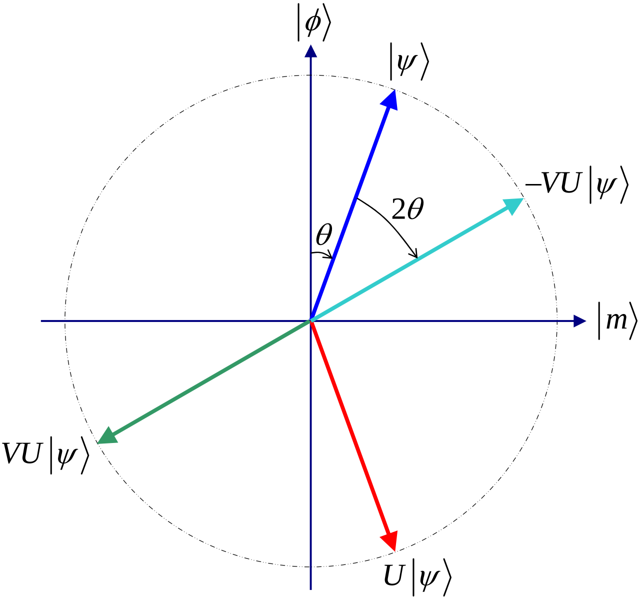
Visualization of first Grover iteration



Visualization of first Grover iteration



Visualization of first Grover iteration



The Grover iteration is a rotation

Geometrically, U is a reflection around the $|m\rangle$ axis and V is a reflection around the $|\psi\rangle$ axis, which is almost but not quite orthogonal to the $|m\rangle$ axis

The product of these two reflections is a clockwise rotation by an angle 2θ, up to an overall minus sign

From this geometric picture, or by explicit calculation using trig identities, it is easy to verify that

$$(VU)^k = (-1)^k \begin{pmatrix} \cos 2k\theta & \sin 2k\theta \\ -\sin 2k\theta & \cos 2k\theta \end{pmatrix}$$

Complexity of Grover search

Recall that our initial state is $|\psi\rangle = \sin\theta|m\rangle + \cos\theta|\phi\rangle$

How large should k be before $(VU)^k|\psi\rangle$ is close to $|m\rangle$?

We start an angle θ from the $|\phi\rangle$ axis and rotate toward $|m\rangle$ by an angle 2θ per iteration

$$|\langle m|(VU)^k|\psi\rangle|^2 = \sin^2((2k+1)\theta)$$

$\Rightarrow$ To rotate by $\pi/2$, we need

$$\begin{aligned} \theta + 2k\theta &= \pi/2 \\ k &\approx \frac{\pi}{4}\theta^{-1} \approx \frac{\pi}{4}\sqrt{N} \end{aligned}$$

Grover search

Grover's algorithm solves a completely unstructured search problem with N possible solutions, yet finds a unique solution in only $O(\sqrt{N})$ queries!

While this is only a polynomial separation, it is very generic, and it is surprising that we can obtain a speedup for a search in which we have so little information to go on

Optimality of Grover's algorithm

It can also be shown that this quantum algorithm is optimal

Any quantum algorithm needs at least $\Omega(\sqrt{N})$ queries to find a marked item (or even to decide if some item is marked)

We will prove this in the last quantum algorithms lecture

Multiple solutions

Suppose there are M marked items

Then there is a two-dimensional invariant subspace span$\{|\mu\rangle, |\psi\rangle\}$ where

$$|\mu\rangle = \frac{1}{\sqrt{M}} \sum_{x \text{ marked}} |x\rangle$$

is the uniform superposition over all marked items

The Gram-Schmidt process yields the ONB $\{|\mu\rangle, |\phi\rangle\}$ where

$$|\phi\rangle = \frac{1}{\sqrt{N-M}} \sum_{x \text{ unmarked}} |x\rangle$$

is the uniform superposition of all non-solutions

Invariant subspace

Now in the basis $\{|\mu\rangle, |\phi\rangle\}$, we have

$$
\begin{aligned}
|\psi\rangle &= \sin\theta|\mu\rangle + \cos\theta|\phi\rangle \text{ where } \sin\theta = \langle\mu|\psi\rangle = \sqrt{\frac{M}{N}} \\
VU &= -\begin{pmatrix} \cos 2\theta & \sin 2\theta \\ -\sin 2\theta & \cos 2\theta \end{pmatrix}
\end{aligned}
$$

Overshooting

The success probability is

$$\sin((2k+1)\theta) \text{ where } \sin\theta = \sqrt{\frac{M}{N}}$$

$\Rightarrow$ We need to apply VU

$$k \approx \frac{\pi}{4}\sqrt{\frac{N}{M}}$$

times

Due to the oscillatory behavior of the success probability, it is important not to overshoot: if the number of iterations is too large, the success probability will decrease

Quantum counting (1/2)

The eigenvalues of

$$-VU = \begin{pmatrix} \cos 2\theta & \sin 2\theta \\ -\sin 2\theta & \cos 2\theta \end{pmatrix}$$

are $e^{2i\theta}$ and $e^{-2i\theta}$

The initial state $|\psi\rangle$ is a superposition of the two eigenvectors corresponding to the above two eigenvalues

$\Rightarrow$ Using phase estimation, we can obtain an estimate $\tilde{\theta}$ such that

$$|\theta - \tilde{\theta}| \leq \epsilon$$

by invoking the controlled version of $-VU$

$$O(1/\epsilon) \text{ times}$$

Quantum counting (2/2)

The estimate $\tilde{\theta}$ of θ gives an estimate $\tilde{M}$ of M

Error:

$$\begin{aligned}\left|\frac{M}{N} - \frac{\tilde{M}}{N}\right| &= |\sin^2\theta - \sin^2\tilde{\theta}| \\ &= |\sin\theta + \sin\tilde{\theta}|\,|\sin\theta - \sin\tilde{\theta}| \\ &\approx 2\sqrt{\frac{M}{N}}\epsilon\end{aligned}$$

Equivalently, we get an approximation $\tilde{M} = M(1 + O(\varepsilon))$ using $O(\frac{1}{\varepsilon}\sqrt{N/M})$ queries

Amplitude amplification

Suppose we have a classical (randomized) algorithm that produces a solution to some problem with probability p

Assume we can recognize correct solutions

Classical strategy: repeat $O(1/p)$ times

Quantum amplitude amplification uses only $O(1/\sqrt{p})$ repetitions

Exercise: Quantum search and state generation

Let $|\psi\rangle$ be an unknown quantum state. Consider quantum algorithms for preparing $|\psi\rangle$ given two different black boxes.

1. Suppose you are given the unitary $U := I - 2|\psi\rangle\langle\psi|$ as a black box. Consider a quantum algorithm that starts in some known state $|\phi\rangle$ and alternates between performing U and $V := I - 2|\phi\rangle\langle\phi|$. How many queries to U are required to prepare a state close to $|\psi\rangle$? Express your answer as a function of $|\langle\psi|\phi\rangle|$.
2. Now suppose you are given a reversible black box that, on input $x \in \{1, \ldots, N\}$, returns the amplitude $\alpha_x := \langle x|\psi\rangle$ of the state $|\psi\rangle$ in the computational basis state $|x\rangle$. (You may assume that the black box specifies the complex number α_x to arbitrary precision.) Describe an algorithm that prepares a state close to $|\psi\rangle$ using $O(\sqrt{N})$ queries. (Hint: Two queries to the black box can be used to perform the isometry $|x\rangle \mapsto |x\rangle(\alpha_x|0\rangle + \sqrt{1-|\alpha_x|^2}|1\rangle)$.)

Part VI

Quantum walk

Randomized algorithms

Randomness is an important tool in computer science

Black-box problems

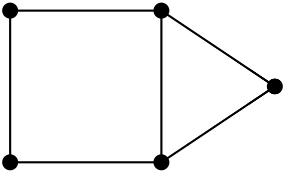
- Huge speedups are possible (Deutsch-Jozsa: $2^{\Omega(n)}$ vs. $O(1)$)
- Polynomial speedup for some total functions (game trees: $\Omega(n)$ vs. $O(n^{0.754})$)

Natural problems

- Majority view is that derandomization should be possible (P=BPP)
- Randomness may give polynomial speedups (Schöning algorithm for k-SAT)
- Can be useful for algorithm design

Random walk

Graph $G = (V, E)$



Two kinds of walks:

- Discrete time
- Continuous time

Random walk algorithms

Undirected s–t connectivity in log space

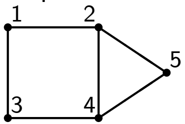
- Problem: given an undirected graph $G = (V, E)$ and $s, t \in V$, is there a path from s to t?
- A random walk from s eventually reaches t iff there is a path
- Taking a random walk only requires log space
- Can be derandomized (Reingold 2004), but this is nontrivial

Markov chain Monte Carlo

- Problem: sample from some probability distribution (uniform distribution over some set of combinatorial objects, thermal equilibrium state of a physical system, etc.)
- Create a Markov chain whose stationary distribution is the desired one
- Run the chain until it converges

Continuous-time quantum walk

Graph G



$$A = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{pmatrix} \qquad L = \begin{pmatrix} -2 & 1 & 1 & 0 & 0 \\ 1 & -3 & 0 & 1 & 1 \\ 1 & 0 & -2 & 1 & 0 \\ 0 & 1 & 1 & -3 & 1 \\ 0 & 1 & 0 & 1 & -2 \end{pmatrix}$$

adjacency matrix Laplacian

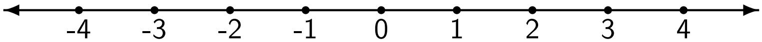
Random walk on G

- State: probability $p_v(t)$ of being at vertex v at time t
- Dynamics: $\frac{\mathrm{d}}{\mathrm{d}t}\vec{p}(t) = -L\vec{p}(t)$

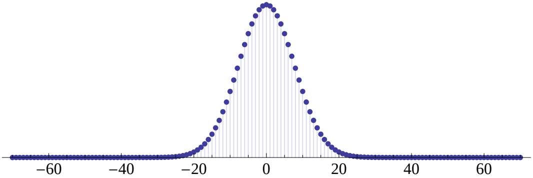
Quantum walk on G

- State: amplitude $q_v(t)$ to be at vertex v at time t
  (i.e., $|\psi(t)\rangle = \sum_{v\in V} q_v(t)|v\rangle$)
- Dynamics: $\mathrm{i}\frac{\mathrm{d}}{\mathrm{d}t}\vec{q}(t) = -L\vec{q}(t)$

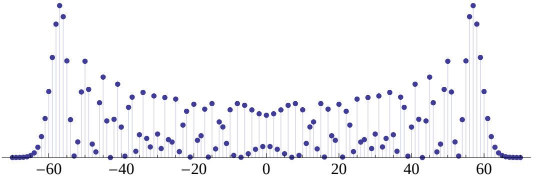
Random vs. quantum walk on the line



Classical:



Quantum:

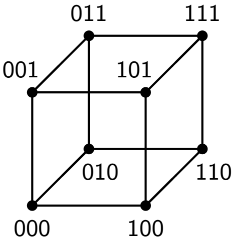


Random vs. quantum walk on the hypercube

$V = \{0,1\}^n$

$E = \{(x,y) \in V \times V :$ x and y differ in exactly one bit$\}$

$n = 3$:

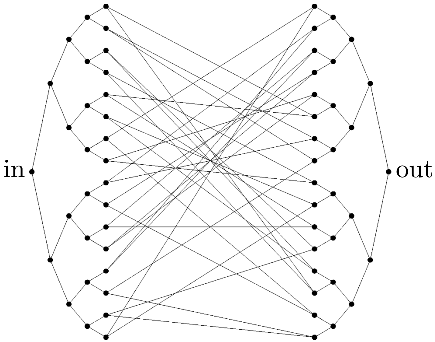


Classical random walk: reaching $11\ldots1$ from $00\ldots0$ is exponentially unlikely

Quantum walk: with $A = \sum_{j=1}^{n} X_j$,

$$e^{-\mathrm{i}At} = \prod_{j=1}^{n} e^{-\mathrm{i}X_j t} = \bigotimes_{j=1}^{n} \begin{pmatrix} \cos t & -\mathrm{i}\sin t \\ -\mathrm{i}\sin t & \cos t \end{pmatrix}$$

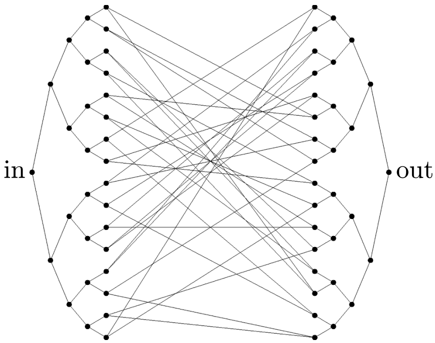
Glued trees problem



Black-box description of a graph

- Vertices have arbitrary labels
- Label of 'in' vertex is known
- Given a vertex label, black box returns labels of its neighbors
- Restricts algorithms to explore the graph locally

Glued trees problem: Classical query complexity

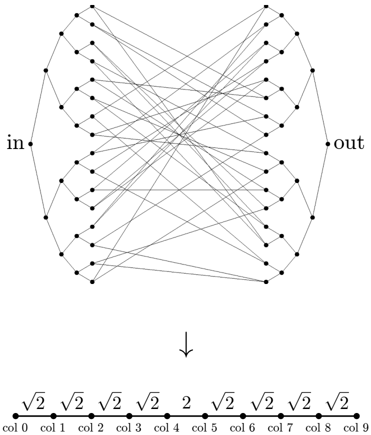


Let n denote the height of one of the binary trees

Classical random walk from 'in': probability of reaching 'out' is $2^{-\Omega(n)}$ at all times

In fact, the classical query complexity is $2^{\Omega(n)}$

Glued trees problem: Exponential speedup



Column subspace

$$|\mathrm{col}\ j\rangle := \frac{1}{\sqrt{N_j}} \sum_{v \in \mathrm{column}\ j} |v\rangle$$

$$N_j := \begin{cases} 2^j & \text{if } j \in [0, n] \\ 2^{2n+1-j} & \text{if } j \in [n+1, 2n+1] \end{cases}$$

Reduced adjacency matrix

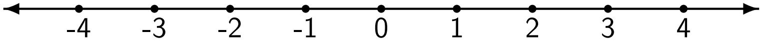
$$\langle \mathrm{col}\ j | A | \mathrm{col}\ j+1 \rangle$$
$$= \begin{cases} \sqrt{2} & \text{if } j \in [0, n-1] \\ \sqrt{2} & \text{if } j \in [n+1, 2n] \\ 2 & \text{if } j = n \end{cases}$$

Discrete-time quantum walk: Need for a coin

Quantum analog of discrete-time random walk?

Unitary matrix $U \in \mathbb{C}^{|V|\times|V|}$ with $U_{vw} \neq 0$ iff $(v, w) \in E$

Consider the line:

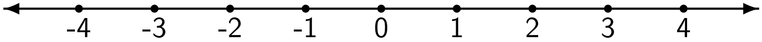


Define walk by $|x\rangle \mapsto \frac{1}{\sqrt{2}}(|x-1\rangle + |x+1\rangle)$?

But then $|x+2\rangle \mapsto \frac{1}{\sqrt{2}}(|x+1\rangle + |x+3\rangle)$, so this is not unitary!

In general, we must enlarge the state space.

Discrete-time quantum walk on a line

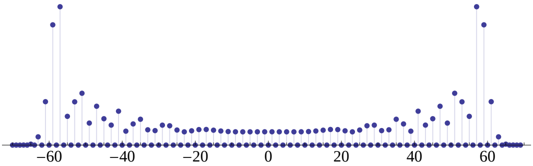


Add a "coin": state space $\mathrm{span}\{|x\rangle \otimes |{\leftarrow}\rangle, |x\rangle \otimes |{\rightarrow}\rangle : x \in \mathbb{Z}\}$

Coin flip: $C := I \otimes H$

Shift: $$\begin{aligned} S|x\rangle \otimes |{\leftarrow}\rangle &= |x-1\rangle \otimes |{\leftarrow}\rangle \\ S|x\rangle \otimes |{\rightarrow}\rangle &= |x+1\rangle \otimes |{\rightarrow}\rangle \end{aligned}$$

Walk step: SC



The Szegedy walk

State space: $\mathrm{span}\{|v\rangle \otimes |w\rangle, |w\rangle \otimes |v\rangle : (v, w) \in E\}$

Let W be a stochastic matrix (a discrete-time random walk)

$$\text{Define} \quad |\psi_v\rangle := |v\rangle \otimes \sum_{w \in V} \sqrt{W_{wv}}|w\rangle \quad (\text{note } \langle\psi_v|\psi_w\rangle = \delta_{v,w})$$

$$R := 2\sum_{v \in V} |\psi_v\rangle\langle\psi_v| - I$$

$$S(|v\rangle \otimes |w\rangle) := |w\rangle \otimes |v\rangle$$

Then a step of the walk is the unitary operator $U := SR$

Spectrum of the walk

Let $T := \sum_{v \in V} |\psi_v\rangle\langle v|$, so $R = 2TT^\dagger - I$.

Theorem (Szegedy)

Let W be a stochastic matrix. Suppose the matrix

$$\sum_{v,w} \sqrt{W_{vw} W_{wv}} |w\rangle\langle v|$$

has an eigenvector $|\lambda\rangle$ with eigenvalue λ. Then

$$\frac{I - e^{\pm \mathrm{i} \arccos \lambda} S}{\sqrt{2(1-\lambda^2)}} T|\lambda\rangle$$

are eigenvectors of $U = SR$ with eigenvalues

$$e^{\pm \mathrm{i} \arccos \lambda}.$$

Proof of Szegedy's spectral theorem

Proof sketch.

Straightforward calculations give

$$TT^\dagger = \sum_{v \in V} |\psi_v\rangle\langle\psi_v| \qquad T^\dagger T = I$$

$$T^\dagger S T = \sum_{v,w \in V} \sqrt{W_{vw} W_{wv}} |w\rangle\langle v| = \sum_\lambda |\lambda\rangle\langle\lambda|$$

which can be used to show

$$U(T|\lambda\rangle) = ST|\lambda\rangle \qquad U(ST|\lambda\rangle) = 2\lambda ST|\lambda\rangle - T|\lambda\rangle.$$

Diagonalizing within the subspace span$\{T|\lambda\rangle, ST|\lambda\rangle\}$ gives the desired result. □

Exercise. Fill in the details

Random walk search algorithm

Given $G = (V, E)$, let $M \subset V$ be a set of *marked vertices*

Start at a random unmarked vertex

Walk until we reach a marked vertex:

$$W'_{vw} := \begin{cases} 1 & w \in M \text{ and } v = w \\ 0 & w \in M \text{ and } v \neq w \\ W_{vw} & w \notin M. \end{cases}$$

$$= \begin{pmatrix} W_M & 0 \\ V & I \end{pmatrix} \quad (W_M\text{: delete marked rows and columns of } W)$$

Question. How long does it take to reach a marked vertex?

Classical hitting time

Take t steps of the walk:

$$(W')^t = \begin{pmatrix} W_M^t & 0 \\ V(I + W_M + \cdots + W_M^{t-1}) & I \end{pmatrix}$$
$$= \begin{pmatrix} W_M^t & 0 \\ V\frac{I-W_M^t}{I-W_M} & I \end{pmatrix}$$

Convergence time depends on how close $\|W_M\|$ is to 1, which depends on the spectrum of W

Lemma

Let $W = W^T$ be a symmetric Markov chain. Let the second largest eigenvalue of W be $1-\delta$, and let $\epsilon = |M|/|V|$ (the fraction of marked items). Then the probability of reaching a marked vertex is $\Omega(1)$ after $t = O(1/\delta\epsilon)$ steps of the walk.

Quantum walk search algorithm

Start from the state $\frac{1}{\sqrt{N-|M|}} \sum_{v \notin M} |\psi_v\rangle$

Consider the walk U corresponding to W':

$$\sum_{v,w \in V} \sqrt{W'_{v,w} W'_{w,v}} |w\rangle\langle v| = \begin{pmatrix} W_M & 0 \\ 0 & I \end{pmatrix}$$

Eigenvalues of U are $e^{\pm \mathrm{i} \arccos \lambda}$ where the λ are eigenvalues of W_M

Perform phase estimation on U with precision $O(\sqrt{\delta\epsilon})$

- no marked items $\implies$ estimated phase is 0
- ϵ fraction of marked items $\implies$ nonzero phase with probability $\Omega(1)$

Further refinements give algorithms for *finding* a marked item

Grover's algorithm revisited

Problem

Given a black box $f\colon X \to \{0,1\}$, is there an x with $f(x) = 1$?

Markov chain on $N = |X|$ vertices:

$$W := \frac{1}{N}\begin{pmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 1 \end{pmatrix} = |\psi\rangle\langle\psi|, \quad |\psi\rangle := \frac{1}{\sqrt{N}}\sum_{x\in X}|x\rangle$$

Eigenvalues of W are $0, 1 \implies \delta = 1$

Hard case: one marked vertex, $\epsilon = 1/N$

Hitting times

- Classical: $O(1/\delta\epsilon) = O(N)$
- Quantum: $O(1/\sqrt{\delta\epsilon}) = O(\sqrt{N})$

Element distinctness

Problem

Given a black box $f: X \to Y$, are there distinct x, x' with $f(x) = f(x')$?

Let $N = |X|$; classical query complexity is $\Omega(N)$

Consider a quantum walk on the Hamming graph $H(N, M)$

- Vertices: $\{(x_1, \ldots, x_M): x_i \in X\}$
- Store the values $(f(x_1), \ldots, f(x_M))$ at vertex $(x_1, \ldots, x_M)$
- Edges between vertices that differ in exactly one coordinate

Element distinctness: Analysis

Spectral gap: $\delta = O(1/M)$

Fraction of marked vertices: $\epsilon \geq 2\binom{M}{2} N^{M-2}/N^M = \Theta(M^2/N^2)$

Quantum hitting time: $O(1/\sqrt{\delta\epsilon}) = O(N/\sqrt{M})$

Quantum query complexity:

- M queries to prepare the initial state
- 2 queries for each step of the walk (compute f, uncompute f)
- Overall: $M + O(N/\sqrt{M})$

Choose $M = N^{2/3}$: query complexity is $O(N^{2/3})$ (optimal!)

Quantum walk algorithms

Quantum walk search algorithms

- Spatial search
- Subgraph finding
- Checking matrix multiplication
- Testing if a black-box group is abelian

Evaluating Boolean formulas

Exponential speedup for a natural problem?

Exercise: Triangle finding (1/2)

The goal of the *triangle problem* is to decide whether an n-vertex graph G contains a triangle (a complete subgraph on 3 vertices). The graph is specified by a black box that, for any pair of vertices of G, returns a bit indicating whether those vertices are connected by an edge in G.

1. What is the classical query complexity of the triangle problem?
2. Say that an edge of G is a *triangle edge* if it is part of a triangle in G. What is the quantum query complexity of deciding whether a particular edge of G is a triangle edge?
3. Now suppose you know the vertices and edges of some m-vertex subgraph of G. Explain how you can decide whether this subgraph contains a triangle edge using $O(m^{2/3}\sqrt{n})$ quantum queries.

Exercise: Triangle finding (2/2)

4. Consider a quantum walk algorithm for the triangle problem. The walk takes place on a graph $\mathcal{G}$ whose vertices correspond to subgraphs of G on m vertices, and whose edges correspond to subgraphs that differ by changing one vertex. A vertex of $\mathcal{G}$ is marked if it contains a triangle edge. How many queries does this algorithm use to decide whether G contains a triangle? (Hint: Be sure to account for the S queries used to initialize the walk, the U queries used to move between neighboring vertices of $\mathcal{G}$, and the C queries used to check whether a given vertex of $\mathcal{G}$ is marked. If the walk has spectral gap δ and an ϵ-fraction of the vertices are marked, it can be shown that there is a quantum walk search algorithm with query complexity $S + \frac{1}{\sqrt{\epsilon}}(\frac{1}{\sqrt{\delta}}U + C)$.)
5. Choose a value of m that minimizes the number of queries used by the algorithm. What is the resulting upper bound on the quantum query complexity of the triangle problem?

Part VII

Adversary lower bounds

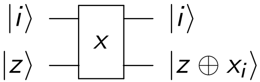
Query complexity

Task: Compute a function $f\colon S \to T$

$S \subseteq \Sigma^n$ is the set of possible inputs, where Σ is the *input alphabet*

- if $S = \Sigma^n$ then f is *total*
- if $S \subsetneq \Sigma^n$ then f is *partial*

Input $x \in S$ is specified by a black box:



where $i \in \{1, \ldots, n\}$

Query algorithms

Structure of a quantum query algorithm:

- Initial state $|\psi\rangle$ does not depend on the oracle string x
- Alternate between queries to the black box O_x and non-query operations $U_1, U_2, \ldots, U_t$

$$|\psi_x^t\rangle := U_t O_x \ldots U_2 O_x U_1 O_x |\psi\rangle$$

- End with a measurement in the computational basis

Goal: Compute $f(x)$ using as few queries as possible

Query models

Three natural models for the query complexity of f:

- $D(f)$: deterministic query complexity
  (algorithm is classical and must always work correctly)
- $R_\epsilon(f)$: randomized query complexity with (two-sided) error probability at most ϵ
- $Q_\epsilon(f)$: quantum query complexity with (two-sided) error probability at most ϵ

For any constant ϵ,
$R_\epsilon(f) = \Theta(R_{1/3}(f))$ and $Q_\epsilon(f) = \Theta(Q_{1/3}(f))$
(repeat several times and take a majority vote)

Clearly $Q_\epsilon(f) \le R_\epsilon(f) \le D(f)$

Quantum queries: Boolean case

Consider $\Sigma = \{0,1\}$

Bit flip oracle:

$$\hat{O}_x|i,b\rangle = |i, b \oplus x_i\rangle \qquad \text{for } i \in \{1,\dots,n\},\ b \in \{0,1\}$$

Phase flip oracle:

$$O_x|i,b\rangle = (-1)^{bx_i}|i,b\rangle \qquad \text{for } i \in \{1,\dots,n\},\ b \in \{0,1\}$$

Phase kickback: $O_x = (I \otimes H)\hat{O}_x(I \otimes H)$

Note: $O_x|i,0\rangle = |i,0\rangle$ for all i is wasteful; alternatively, use

$$O'_x|i\rangle = \begin{cases} (-1)^{x_i}|i\rangle & i \in \{1,\dots,n\} \\ |i\rangle & i = 0 \quad (\text{i.e., } x_0 := 1) \end{cases}$$

But the ability to not query the phase oracle is essential!

Quantum queries: General case

Similar considerations hold when $|\Sigma| = d > 2$

Let $\Sigma = \mathbb{Z}_d$ without loss of generality

Addition oracle:

$$\hat{O}_x|i, b\rangle = |i, b + x_i \bmod d\rangle \qquad \text{for } i \in \{1, \ldots, n\},\ b \in \mathbb{Z}_d$$

Phase oracle:

$$O_x|i, b\rangle = e^{2\pi i b x_i / d}|i, b\rangle \qquad \text{for } i \in \{1, \ldots, n\},\ b \in \mathbb{Z}_d$$

Phase kickback:

$$O_x = (I \otimes F^\dagger)\hat{O}_x(I \otimes F)$$

where F is the Fourier transform over $\mathbb{Z}_d$

A quantum adversary

Lower bound strategy: Oracle is operated by a malicious adversary

Adversary creates a superposition over possible inputs: $\sum_{x\in S} a_x|x\rangle$

Each query is performed by the "super-oracle"

$$O := \sum_{x\in S} |x\rangle\langle x| \otimes O_x$$

After t steps, algorithm produces the state

$$\begin{aligned}|\psi^t\rangle &:= (I \otimes U_t)O \ldots (I \otimes U_2)O(I \otimes U_1)O \left(\sum_{x\in S} a_x|x\rangle \otimes |\psi\rangle\right) \\ &= \sum_{x\in S} a_x|x\rangle \otimes |\psi_x^t\rangle\end{aligned}$$

Getting entangled with the adversary

Intuition: To learn x, the state $|\psi^t\rangle$ must be very entangled

Reduced density matrix of the oracle:

$$\rho^t := \sum_{x,y \in S} a_x^* a_y \langle \psi_x^t | \psi_y^t \rangle \, |x\rangle\langle y|$$

Initial state ρ^0 is pure

Final state ρ^t must be mixed

Quantify how much more mixed the state can become with a single query

We could consider the von Neumann entropy of ρ^t, but this is cumbersome

Distinguishing quantum states

Fact

Given one of two pure states $|\psi\rangle, |\phi\rangle$, we can make a measurement that determines which state we have with error probability at most ϵ if and only if $|\langle\psi|\phi\rangle| \leq 2\sqrt{\epsilon(1-\epsilon)}$.

Exercise. Prove this

So it's convenient to consider measures that are linear in the inner products $\langle\psi_x^t|\psi_y^t\rangle$

Adversary matrices

The adversary bound uses a matrix $\Gamma \in \mathbb{R}^{|S|\times|S|}$

$\Gamma_{x,y}$ measures how hard it is to distinguish between x and y

We say Γ is an *adversary matrix* if

1. $\Gamma_{xy} = \Gamma_{yx}$,
2. $\Gamma_{xy} \geq 0$, and
3. if $f(x) = f(y)$ then $\Gamma_{xy} = 0$

Weight function

Given an adversary matrix Γ, we define a weight function

$$W_j := \sum_{x,y \in S} \Gamma_{xy} a_x^* a_y \langle \psi_x^j | \psi_y^j \rangle$$

We show:

1. W_0 is large
2. To compute f in t queries, W_t must be small
3. W_{j+1} cannot be too much smaller than W_j

Weight function: Initial value

The initial value of the weight function is

$$W_0 = \sum_{x,y \in S} \Gamma_{xy} a_x^* a_y \langle \psi_x^0 | \psi_y^0 \rangle$$
$$= \sum_{x,y \in S} a_x^* \Gamma_{xy} a_y$$

since $|\psi_x^0\rangle$ cannot depend on x

To make this as large as possible, take a to be a principal eigenvector of Γ

$\Rightarrow W_0 = \|\Gamma\|$

Weight function: Final value

If $f(x) \neq f(y)$ then the states $|\psi_x^t\rangle, |\psi_y^t\rangle$ must be distinguishable

To succeed with error probability at most ϵ, we need $|\langle\psi_x^t|\psi_y^t\rangle| \leq 2\sqrt{\epsilon(1-\epsilon)}$

Thus

$$
\begin{aligned}
W_t &= \sum_{x,y\in S} \Gamma_{xy} a_x^* a_y \langle\psi_x^t|\psi_y^t\rangle \\
&\leq \sum_{x,y\in S} \Gamma_{xy} a_x^* a_y 2\sqrt{\epsilon(1-\epsilon)} \\
&= 2\sqrt{\epsilon(1-\epsilon)}\|\Gamma\|
\end{aligned}
$$

Weight function: Making a query (1/5)

Change in weight function:

$$W_{j+1} - W_j = \sum_{x,y \in S} \Gamma_{xy} a_x^* a_y (\langle \psi_x^{j+1} | \psi_y^{j+1} \rangle - \langle \psi_x^j | \psi_y^j \rangle)$$

Change in state: $|\psi_x^{j+1}\rangle = U_{j+1} O_x |\psi_x^j\rangle$

Gram matrix elements:

$$\begin{aligned} \langle \psi_x^{j+1} | \psi_y^{j+1} \rangle &= \langle \psi_x^j | O_x^\dagger U_{j+1}^\dagger U_{j+1} O_y | \psi_y^j \rangle \\ &= \langle \psi_x^j | O_x O_y | \psi_y^j \rangle \end{aligned}$$

Therefore

$$W_{j+1} - W_j = \sum_{x,y \in S} \Gamma_{xy} a_x^* a_y \langle \psi_x^j | (O_x O_y - I) | \psi_y^j \rangle$$

Weight function: Making a query (2/5)

$$W_{j+1} - W_j = \sum_{x,y \in S} \Gamma_{xy} a_x^* a_y \langle \psi_x^j | (O_x O_y - I) | \psi_y^j \rangle$$

We have $O_x O_y |i, b\rangle = (-1)^{b(x_i \oplus y_i)} |i, b\rangle$

Let $P_0 = I \otimes |0\rangle\langle 0|$ and $P_i = |i, 1\rangle\langle i, 1|$

Then

$$\begin{aligned} O_x O_y - I &= P_0 + \sum_{i=1}^{n} (-1)^{x_i \oplus y_i} P_i - I \\ &= -2 \sum_{i:\ x_i \neq y_i} P_i \end{aligned}$$

Weight function: Making a query (3/5)

$$O_x O_y - I = -2 \sum_{i:\ x_i \neq y_i}^{n} P_i$$

so

$$\begin{aligned}
|W_{j+1} - W_j| &= \sum_{x,y \in S} \Gamma_{xy} a_x^* a_y \langle \psi_x^j | (O_x O_y - I) | \psi_y^j \rangle \\
&= 2 \left| \sum_{x,y \in S} \sum_{i:\ x_i \neq y_i} \Gamma_{xy} a_x^* a_y \langle \psi_x^j | P_i | \psi_y^j \rangle \right| \\
&\leq 2 \sum_{x,y \in S} \sum_{i:\ x_i \neq y_i} \Gamma_{xy} |a_x^* a_y \langle \psi_x^j | P_i | \psi_y^j \rangle| && (\triangle) \\
&\leq 2 \sum_{x,y \in S} \sum_{i:\ x_i \neq y_i} \Gamma_{xy} \| a_x P_i | \psi_x^j \rangle \| \cdot \| a_y P_i | \psi_y^j \rangle \| && \text{(C-S)}
\end{aligned}$$

Weight function: Making a query (4/5)

For each $i \in \{1, \ldots, n\}$, define $\Gamma_i \in \mathbb{R}^{|S| \times |S|}$ by

$$(\Gamma_i)_{xy} = \begin{cases} \Gamma_{xy} & \text{if } x_i \neq y_i \\ 0 & \text{if } x_i = y_i, \end{cases}$$

and define vectors v_i with components $(v_i)_x = \| a_x P_i | \psi_x^j \rangle \|$

$$\begin{aligned} |W_{j+1} - W_j| &\leq 2 \sum_{x,y \in S} \sum_{i=1}^{n} (v_i)_x (\Gamma_i)_{xy} (v_i)_y \\ &= 2 \sum_{i=1}^{n} v_i^\dagger \Gamma_i v_i \\ &\leq 2 \sum_{i=1}^{n} \|\Gamma_i\| \cdot \|v_i\|^2 \end{aligned}$$

Weight function: Making a query (5/5)

$$|W_{j+1} - W_j| \leq 2 \sum_{i=1}^{n} \|\Gamma_i\| \cdot \|v_i\|^2$$

Since

$$\begin{aligned}
\sum_{i=1}^{n} \|v_i\|^2 &= \sum_{i=1}^{n} \sum_{x \in S} \|a_x P_i |\psi_x^j\rangle\|^2 \\
&\leq \sum_{x \in S} a_x^2 \||\psi_x^j\rangle\|^2 \\
&= \sum_{x \in S} a_x^2 \\
&= 1,
\end{aligned}$$

we have

$$|W_{j+1} - W_j| \leq 2 \max_{i \in \{1,\ldots,n\}} \|\Gamma_i\|$$

Weight function: Putting everything together

Since $W_0 = \|\Gamma\|$, we have

$$W_t \geq \|\Gamma\| - 2t \max_{i \in \{1,\ldots,n\}} \|\Gamma_i\|$$

So $W_t \leq 2\sqrt{\epsilon(1-\epsilon)}\|\Gamma\|$ implies

$$t \geq \frac{1 - 2\sqrt{\epsilon(1-\epsilon)}}{2} \mathrm{Adv}(f)$$

where

$$\mathrm{Adv}(f) := \max_{\Gamma} \frac{\|\Gamma\|}{\max_{i \in \{1,\ldots,n\}} \|\Gamma_i\|}$$

with the maximum taken over all adversary matrices Γ

Example: Unstructured search (1/3)

Problem: Distinguish no marked item from unique marked item

$$S = \{000\ldots00, 100\ldots00, 010\ldots00, \ldots, 000\ldots01\}$$

Adversary matrix:

$$\Gamma = \begin{pmatrix} 0 & \gamma_1 & \cdots & \gamma_n \\ \gamma_1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \gamma_n & 0 & \cdots & 0 \end{pmatrix} \qquad \gamma_1, \ldots, \gamma_n \geq 0$$

Symmetry: $\gamma_1 = \cdots = \gamma_n = 1$

Example: Unstructured search (2/3)

Consider

$$\Gamma^2 = \begin{pmatrix} n & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & \cdots & 1 \end{pmatrix}$$

$\|\Gamma^2\| = n$, so $\|\Gamma\| = \sqrt{n}$

$$\|\Gamma_i\| = \|\Gamma_1\| = \left\| \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 0 \end{pmatrix} \right\| = 1$$

Example: Unstructured search (3/3)

Our adversary matrix has $\|\Gamma\| = \sqrt{n}$, $\|\Gamma_i\| = 1$

So $\mathrm{Adv}(\mathrm{OR}) \geq \frac{\|\Gamma\|}{\|\Gamma_i\|} = \sqrt{n}$

Therefore $Q_\epsilon(\mathrm{OR}) \geq \frac{1-2\sqrt{\epsilon(1-\epsilon)}}{2}\sqrt{n}$

Thus Grover's algorithm is optimal up to a constant factor (recall that Grover's algorithm finds a unique marked item with probability $1 - o(1)$ in $(\frac{\pi}{4} + o(1))\sqrt{n}$ queries)

Other adversaries

The adversary method described above is a generalization of the method originally formulated by Ambainis, which considered only a relation between yes and no inputs and did not allow arbitrary positive weights.

Later, it was realized that one can use negative weights and still obtain a lower bound, and that sometimes this bound can be dramatically better.

In fact, it was shown by Reichardt that the adversary bound allowing negative weights is essentially tight: up to constant factors, it characterizes quantum query complexity.

Exercise: Original formulation of the adversary method

Choose $X, Y \subset \{0,1\}^n$ such that $f(x) \neq f(y)$ for all $x \in X, y \in Y$. For any relation $R \subset X \times Y$, define

$$\begin{aligned}
m &:= \min_{x \in X} |\{y \in Y \colon (x,y) \in R| \\
m' &:= \min_{y \in Y} |\{x \in X \colon (x,y) \in R| \\
\ell &:= \max_{\substack{x \in X \\ i \in \{1,\dots,n\}}} |\{y \in Y \colon (x,y) \in R \text{ and } x_i \neq y_i\}| \\
\ell' &:= \max_{\substack{y \in Y \\ i \in \{1,\dots,n\}}} |\{x \in X \colon (x,y) \in R \text{ and } x_i \neq y_i\}|.
\end{aligned}$$

Then define $\mathrm{Amb}(f) := \max_{X,Y,R} \sqrt{\frac{mm'}{\ell\ell'}}$.

Prove that $\mathrm{Adv}(f) \geq \mathrm{Amb}(f)$, and hence that $Q_\epsilon(f) \geq \frac{1-2\sqrt{\epsilon(1-\epsilon)}}{2}\mathrm{Amb}(f)$.

Exercise: Applying the adversary method

1. Define $\textsc{parity}\colon \{0,1\}^n \to \{0,1\}$ by $\textsc{parity}(x) = x_1 \oplus \cdots \oplus x_n$. Show that $Q(\textsc{parity}) = \Omega(n)$.

2. Define $\textsc{nand}^2\colon \{0,1\}^{n^2} \to \{0,1\}$ by

$$\textsc{nand}^2(x) = \textsc{nand}\Big(\textsc{nand}(x_1, \ldots, x_n), \textsc{nand}(x_{n+1}, \ldots, x_{2n}), \\ \ldots, \textsc{nand}(x_{n^2-n+1}, \ldots, x_{n^2})\Big).$$

   Show that $Q(\textsc{nand}^2) = \Omega(n)$.

3. Let $x \in \{0,1\}^{\binom{n}{2}}$ specify the edges of a simple, undirected n-vertex graph, and define $\textsc{con}\colon \{0,1\}^{\binom{n}{2}} \to \{0,1\}$ by

$$\textsc{con}(x) = \begin{cases} 1 & \text{if the graph described by } x \text{ is connected} \\ 0 & \text{otherwise.} \end{cases}$$

   Show that $Q(\textsc{con}) = \Omega(n^{3/2})$.