Richard Cleve
David R. Cheriton School of Computer Science
Institute for Quantum Computing
University of Waterloo
Quantum states on $n$ qubits are $2^n$-dimensional unit vectors

The basic operations on them are:

- **unitary operations** (rotations)
- **measurements**, that project on to the basis states

Quantum algorithms so far:

- $f(0) \oplus f(1)$  [Deutsch]
- one-out-of-four search
one-out-of-$N$ search?

Natural question: what about search problems in spaces larger than four (and without uniqueness conditions)?

For spaces of size eight (say), the previous method breaks down—the state vectors will not be orthogonal.

Later on, we’ll see how to search a space of size $N$ with $O(\sqrt{N})$ queries ...
Contents of lecture 2

1. Preliminary remarks
2. Quantum states
3. Unitary operations & measurements
4. Subsystem structure & quantum circuit diagrams
5. Introductory remarks about quantum algorithms
6. Deutsch’s parity algorithm
7. One-out-of-four search algorithm
8. Shor’s period finding algorithm
9. Grover’s search algorithm
10. Concluding remarks
Contents of lecture 2

1. Preliminary remarks
2. Quantum states
3. Unitary operations & measurements
4. Subsystem structure & quantum circuit diagrams
5. Introductory remarks about quantum algorithms
6. Deutsch’s parity algorithm
7. One-out-of-four search algorithm
8. **Shor’s period finding algorithm**
9. Grover’s search algorithm
10. Concluding remarks
Period-finding

Given: \( f : \{0,1\}^n \rightarrow T \) such that \( f \) is (strictly) \( r \)-periodic, with unknown period \( r \)

Classically this is very hard in the general case—essentially it is as hard as finding a collision, which costs \( 2^{O(n)} \) queries

Yet Quantum algorithms can determine \( r \) very efficiently: with only \( O(1) \) queries to \( f \)

This is the basis of Shor’s factoring algorithm ...
Application of period-finding algorithm

Order-finding problem (≈ factoring)

Input: $m$ (an $n$-bit integer) and $a < m$ such that $\gcd(x, m) = 1$

Output: the minimum $r > 0$ such that $a^r = 1 \pmod{m}$

Example: let $a = 4$ and $m = 35$
(note that $\gcd(4, 35) = 1$)

Question: what is $r$ in this case?
Answer: $r = 6$

The sequence is cyclic because the set
$\mathbb{Z}_m^* = \{a \in \{1, 2, \ldots, m - 1\} : \gcd(x, m) = 1\}$
is a group under multiplication mod $m$
Application of period-finding algorithm

Order-finding problem \(\approx\) factoring

**Input:** \(m\) (an \(n\)-bit integer) and \(a < m\) such that \(\gcd(x, m) = 1\)

**Output:** the minimum \(r > 0\) such that \(a^r = 1 \pmod{m}\)

No classical polynomial-time algorithm is known for this problem—**in fact, the factoring problem reduces to it**

Order-finding reduces to finding the period of the function \(f(x) = a^x \pmod{m}\), which can be computed in polynomial time

A circuit computing the function \(f\) is substituted into the black-box:
Sketch of period-finding algorithm

Construct $\sum_x |x, f(x)\rangle$ and then measure second register to get the following state:

$$|k\rangle + |k+r\rangle + |k+2r\rangle + |k+3r\rangle + \ldots + |k+(s-1)r\rangle$$

Measuring this state yields just a uniformly random value.

More is needed to extract $r$, a global property of $f$...
Quantum Fourier transform

\[ F_N = \frac{1}{\sqrt{N}} \begin{bmatrix}
1 & 1 & 1 & 1 & \cdots & 1 \\
1 & \omega & \omega^2 & \omega^3 & \cdots & \omega^{N-1} \\
1 & \omega^2 & \omega^4 & \omega^6 & \cdots & \omega^{2(N-1)} \\
1 & \omega^3 & \omega^6 & \omega^9 & \cdots & \omega^{3(N-1)} \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
1 & \omega^{N-1} & \omega^{2(N-1)} & \omega^{3(N-1)} & \cdots & \omega^{(N-1)^2}
\end{bmatrix} \]

where \( \omega = e^{2\pi i/N} \)

It’s a unitary operation on \( n \) qubits (an \( N \times N \) matrix, where \( N = 2^n \))
Applying the quantum Fourier transform to the state
\[ |k\rangle + |k+r\rangle + |k+2r\rangle + |k+3r\rangle + \ldots + |k+(s-1)r\rangle \]
yields the state
\[ |0\rangle + \omega^k |s\rangle + \omega^{2k} |2s\rangle + \omega^{3k} |3s\rangle + \ldots + \omega^{(r-1)k} |(r-1)s\rangle \]

\[ \omega = e^{2\pi i/r} \]
\[ s = r^{-1} \]

**Note:** there is no longer an offset \( k \) (it’s now part of the “phase”)
Measure to get multiple of \( s \), from which \( r \) can be deduced
Computing the QFT

Quantum circuit for $F_{32}$:

Gates: $H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$

For $F_{2n}$ costs $O(n^2)$ gates
Quantum algorithm for order-finding

$|0\rangle, |0\rangle, |0\rangle, |0\rangle, |1\rangle$

$U_{a,m}|x,y\rangle = |x,a^xy \mod m\rangle$ (poly-size circuit)

Number of gates for a constant success probability is: $O(n^2 \log n \log\log n)$
Two-dimensional periodicity

Given:
\[ f : \{0,1\}^n \times \{0,1\}^n \rightarrow T \] with a two-dimensional repeating pattern

Goal: find a simple description of this periodic structure

Quantum algorithms can also solve this very efficiently, and this is the basis of Shor’s discrete logarithm algorithm [1994]
Hidden subgroup problem

Let $G$ be a known group and $H$ be an unknown subgroup of $G$

Let $f : G \to T$ have the property $f(x) = f(y)$ iff $xy^{-1} \in H$
(i.e., $x$ and $y$ are in the same right coset of $H$)

Problem: given a method for computing $f$, determine $H$

Example: $G = S_n$, the symmetric group (permutations of \{1,2,\ldots,n\})

Interesting fact: a fast algorithm for this leads to a fast algorithm for the graph isomorphism problem

... alas no efficient quantum has been found for this version of HSP, despite significant effort by many people
Contents of lecture 2

1. Preliminary remarks
2. Quantum states
3. Unitary operations & measurements
4. Subsystem structure & quantum circuit diagrams
5. Introductory remarks about quantum algorithms
6. Deutsch’s parity algorithm
7. One-out-of-four search algorithm
8. Shor’s period finding algorithm
9. Grover’s search algorithm
10. Concluding remarks
Is factoring an **NP**-hard problem?

If so, then *every* problem in **NP** is solvable by a poly-time quantum algorithm!

But factoring hasn’t been shown to be **NP**-hard

Moreover, there is “evidence” that it is not **NP**-hard:

factoring $\in \text{NP} \cap \text{co-NP}$

If factoring is **NP**-hard then **NP** = **co-NP
Quantum search problem

Given: a black box computing $f: \{0,1\}^n \rightarrow \{0,1\}$

Goal: find $x \in \{0,1\}^n$ such that $f(x) = 1$

Classically, using probabilistic procedures, order $2^n$ queries are necessary to succeed—even with probability $\frac{3}{4}$ (say)

Grover’s quantum algorithm makes only $O(\sqrt{2^n})$ queries

[Query diagram]

[Grover ‘96]
Applications of quantum search

The function $f$ could be realized as a 3-CNF formula:

$$f(x_1,\ldots,x_n) = (x_1 \lor \overline{x}_3 \lor x_4) \land (\overline{x}_2 \lor x_3 \lor \overline{x}_5) \land \cdots \land (\overline{x}_1 \lor x_5 \lor \overline{x}_n)$$

In fact, the search could be for a certificate for any problem in $\mathbf{NP}$

The resulting quantum algorithms appear to be quadratically more efficient than the best classical algorithms known
Prelude to Grover’s algorithm:

two reflections = a rotation

Consider two lines with intersection angle $\theta$:

Net effect: rotation by angle $2\theta$, \textit{regardless of starting vector}
Grover’s algorithm

Basic operations used:

Query: $|x_1\rangle$  $U_f$  $|x_1\rangle$  
$|x_n\rangle$  $|x_n\rangle$  
$|y\rangle$  $|y\oplus f(x_1,\ldots,x_n)\rangle$

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

Diffusion: 

$D = \begin{bmatrix}
-1 + 2/2^n & 2/2^n & \cdots & 2/2^n \\
2/2^n & -1 + 2/2^n & \cdots & 2/2^n \\
\vdots & \vdots & \ddots & \vdots \\
2/2^n & 2/2^n & \cdots & -1 + 2/2^n 
\end{bmatrix}$

Costs only $O(n)$ gates

Hadamard:

$H$

$H$

$H$
Grover’s algorithm II

1. construct state $H|0...0\rangle|\sim\rangle$
2. repeat $k$ times:
   - apply $D U_f$ to state
3. measure state, to get $x \in \{0,1\}^n$, and check if $f(x) = 1$
Grover’s algorithm III

Algorithm: $(DU_f)^k H |0...0\rangle$

\[ H|0...0\rangle = \sum_x |x\rangle \]

Since $DU_f$ is a composition of two reflections, it is a rotation by $2\theta$, where $\sin(\theta) \approx 1/\sqrt{N}$

We want $(2k+1)(1/\sqrt{N}) \approx \pi/2$, so set $k \approx (\pi/4)\sqrt{N}$
Contents of lecture 2

1. Preliminary remarks
2. Quantum states
3. Unitary operations & measurements
4. Subsystem structure & quantum circuit diagrams
5. Introductory remarks about quantum algorithms
6. Deutsch’s parity algorithm
7. One-out-of-four search algorithm
8. Shor’s period finding algorithm and factoring
9. Grover’s search algorithm
10. Concluding remarks
Conclusion

Extended Chruch-Turing Thesis: any polynomial-time algorithm can be simulated by a probabilistic polynomial-time Turing machine

Quantum computing *challenges* this thesis

Either:
- The Extended Chruch-Turing Thesis is false
- Quantum mechanics as we understand it is false
- There is a classical poly-time factoring algorithm

There are many curious properties of quantum information in the context of computation, communication, and cryptography

Waterloo has many people in the faculties of Mathematics, Science, and Engineering working in quantum computing (please see [www.iqc.ca](http://www.iqc.ca) more information)
Some possible project topics

1. Efficient “quantum proofs”: an example is the group non-membership problem

2. Nonlocal effects: apparently paradoxical tricks that can be performed with entangled states

3. Quantum error-correcting codes: important for physical implementations of quantum information processing

4. Alternate models of quantum computation: an example is the measurement-based model

5. Quantum walks: the quantum analogues of random walks, and their algorithmic applications

6. Quantum cryptography: information-theoretical security in a public-key setting
THE END