Quantum Computing
Lecture 6

Quantum Search

Maris Ozols

Grover's search problem

One of the two most important algorithms in quantum computing is Grover's search algorithm (invented by Lov Grover in 1996) for searching for a particular value in an unstructured / unsorted search space.

Example: Searching in a sorted vs unsorted database:

- find a name in a telephone directory
- find a phone number in a telephone directory

Given a black box that for each of N different input strings answers either **yes** or **no**, and there is a unique string with answer **yes**, Grover's algorithm finds this string with $O(\sqrt{N})$ questions (with high probability).

This is a quantum alternative to brute-force search.

Oracle function

Suppose the search space consists of $N = 2^n$ elements which we identify with n-bit strings. Let $f : \{0,1\}^n \to \{0,1\}$ be a function telling which of these elements are marked:

$x \in \{0,1\}^n$ is marked if $f(x) = 1$ and unmarked otherwise.

Important: The oracle can recognize a solution but may not know what the solution is. Even when given the "source code" of f, we may still not be able to easily find x such that $f(x) = 1$.

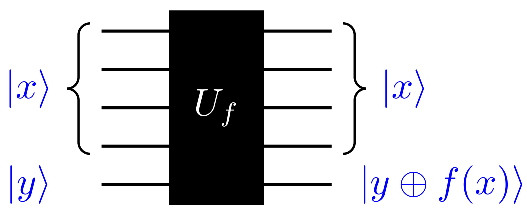
Example: Assume f is hiding a password in one of two ways:

- $f(x) = 1$ iff $x = \texttt{password}$ (knows the password)
- $f(x) = 1$ iff $h(x) = \texttt{c9b93f3f0682250b6cf8331b7ee68fd8}$ (recognizes a correct password but does not know it since inverting a hash function $h : \{0,1\}^n \to \{\texttt{0},\texttt{1},\dots,\texttt{f}\}^m$ in general is very hard)

Grover's black box

Recall from Lecture 4 that any Boolean function $f : \{0,1\}^n \to \{0,1\}$ can be implemented reversibly as follows, where $x \in \{0,1\}^n$, $y \in \{0,1\}$:

$$U_f|x\rangle|y\rangle = |x\rangle|y \oplus f(x)\rangle$$



We refer to U_f as the black box or oracle for computing f.

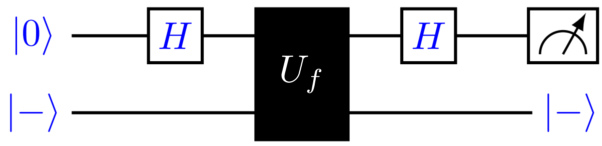
Suppose there is a unique $a \in \{0,1\}^n$ that yields value 1. Let

$$f_a(x) = \begin{cases} 1 & \text{if } x = a \\ 0 & \text{otherwise} \end{cases}$$

Grover's algorithm can determine the value of a with $O(\sqrt{N})$ calls to the black box U_{f_a} where $N = |\{0,1\}^n| = 2^n$.

Deutsch's algorithm revisited

Deutsch's algorithm determines $f(0) \oplus f(1)$ with a single call to the oracle U_f for function $f : \{0,1\} \to \{0,1\}$:

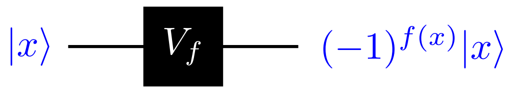


Recall from $U_f|x\rangle|y\rangle = |x\rangle|y \oplus f(x)\rangle$ the phase kick-back trick:

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

Since the last qubit remains unaffected, we effectively have a single-qubit phase oracle V_f that acts as follows:

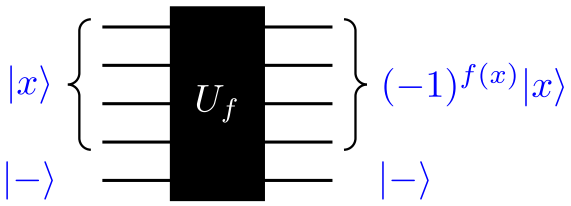
$$V_f|x\rangle = (-1)^{f(x)}|x\rangle$$

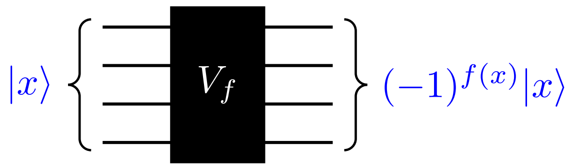


Note that V_f is a diagonal matrix with a ± 1 version of the truth table of f on its diagonal.

The action of V_f

For any $f : \{0,1\}^n \to \{0,1\}$ and $x \in \{0,1\}^n$ the two oracles are





For simplicity, we will assume from now on that we are directly given the n-qubit phase oracle V_f rather than the $(n+1)$-qubit oracle U_f.

Recall that $f_a(x) = 1$ when $x = a$ and $f_a(x) = 0$ otherwise. Hence

$$\begin{aligned} V_{f_a}|a\rangle &= -|a\rangle \\ V_{f_a}|x\rangle &= +|x\rangle \qquad \text{for any } x \neq a \end{aligned}$$

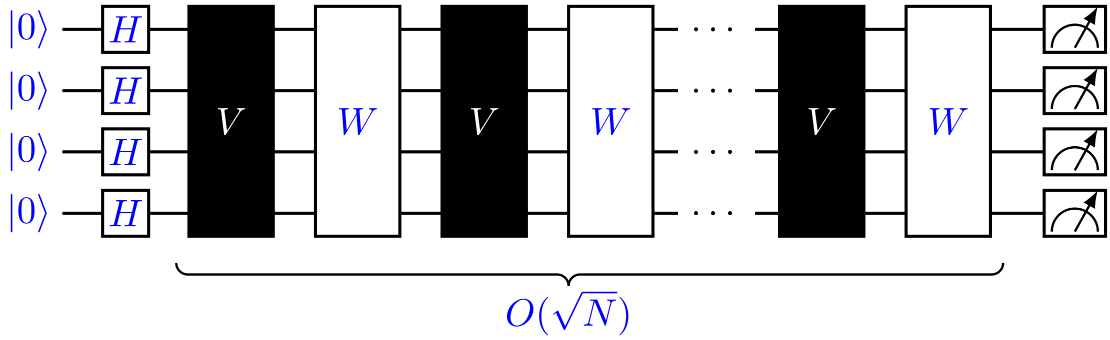
Equivalently, the phase oracle for f_a is

$$V_{f_a} = I - 2|a\rangle\langle a|$$

This is known as the reflection with respect to $|a\rangle$.

Circuit for Grover's algorithm

Let $N = 2^n$ and V be the phase oracle of some n-argument Boolean function f_a. Then Grover's algorithm looks as follows:



Here $W = -(I - 2|\Psi\rangle\langle\Psi|)$ is another reflection, with respect to

$$|\Psi\rangle = \underbrace{|+\rangle \otimes \cdots \otimes |+\rangle}_{n} = |+\rangle^{\otimes n} = \frac{1}{\sqrt{2^n}} \sum_{s \in \{0,1\}^n} |s\rangle$$

The operator $G = WV$ is known as the Grover iterate.

The final state before measurement is

$$G^{\sqrt{N}} \cdot H^{\otimes n} \cdot |0\rangle^{\otimes n} = G^{\sqrt{N}} \cdot |+\rangle^{\otimes n} = G^{\sqrt{N}} \cdot |\Psi\rangle$$

The Grover iterate

Recall that $G = WV$ where

$$W = 2|\Psi\rangle\langle\Psi| - I \qquad\qquad V = I - 2|a\rangle\langle a|$$

Since $|\Psi\rangle$ is the uniform superposition over $N = 2^n$ strings and a is one of them, $\langle\Psi|a\rangle = \langle a|\Psi\rangle = 1/\sqrt{N}$.

Consider the actions of W and V on the two states $|\Psi\rangle$ and $|a\rangle$:

$$W|\Psi\rangle = |\Psi\rangle \qquad\qquad V|\Psi\rangle = |\Psi\rangle - \frac{2}{\sqrt{N}}|a\rangle$$

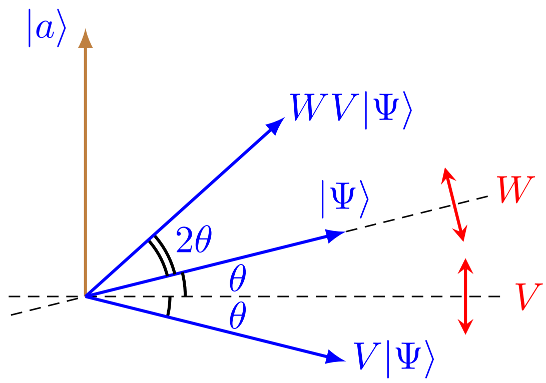
$$W|a\rangle = \frac{2}{\sqrt{N}}|\Psi\rangle - |a\rangle \qquad\qquad V|a\rangle = -|a\rangle$$

Starting from the state $|\Psi\rangle$, repeated applications of V and W will always give a real linear combination of $|a\rangle$ and $|\Psi\rangle$. Thus, the state remains in a 2-dimensional subspace throughout the algorithm!

Geometric view of Grover's algorithm

We can picture the action of V and W in the two-dimensional real plane spanned by the vectors $|a\rangle$ and $|\Psi\rangle$. They are both reflections:

- V reflects about the line perpendicular to $|a\rangle$, since $V|a\rangle = -|a\rangle$
- W reflects about $|\Psi\rangle$, since $W|\Psi\rangle = |\Psi\rangle$



Fact: The composition of two reflections is a rotation. If the angle between the reflection axes is θ then the angle of rotation is 2θ.

The Grover rotation

The Grover iterate $G = WV$ is a rotation through an angle 2θ in the direction from $|\Psi\rangle$ to $|a\rangle$, where the angle between $|\Psi\rangle$ and $|a\rangle$ is $\frac{\pi}{2} - \theta$:

$$\sin\theta = \cos(\tfrac{\pi}{2} - \theta) = \langle a|\Psi\rangle = \frac{1}{\sqrt{N}} = \frac{1}{2^{n/2}}$$

If N is large, $|\Psi\rangle$ and $|a\rangle$ are nearly orthogonal so θ is small:

$$\theta \sim \frac{1}{\sqrt{N}} = \frac{1}{2^{n/2}}$$

Number of iterations

After $t = \frac{\pi/2}{2\theta} \sim \frac{\pi}{4}\sqrt{N}$ iterations of $G = WV$, the state of the system

$$G^t|\Psi\rangle$$

is within an angle θ of $|a\rangle$.

At this point, a measurement in the computational basis yields the state $|a\rangle$ with probability

$$|\langle a|G^t|\Psi\rangle|^2 \geq (\cos\theta)^2 = 1 - (\sin\theta)^2 = \frac{N-1}{N}$$

which is close to 1 when N is large.

Note: Further iterations beyond t will reduce the probability of finding $|a\rangle$.

Multiple solutions

Grover's algorithm works even if the solution $a \in \{0,1\}^n$ is not unique.

Suppose there is a set of solutions $A \subseteq \{0,1\}^n$ and let $M = |A|$ be the number of solutions and $N = 2^n$ be the total number of strings.

Grover iterate is then a rotation in the space spanned by the following two vectors:

$$|\Psi\rangle = \frac{1}{\sqrt{N}} \sum_{s \in \{0,1\}^n} |s\rangle \qquad\qquad |A\rangle = \frac{1}{\sqrt{M}} \sum_{a \in A} |a\rangle$$

As the angle between these is smaller, the number of iterations drops, but so does the probability of success.

The total number of iterations in this case is $O(\sqrt{N/M})$.

Implementing W

How do we implement the second reflection $W = 2|\Psi\rangle\langle\Psi| - I$ using only CNOT and single-qubit unitaries (see Lecture 4)?

Recall that $|\Psi\rangle = |+\rangle^{\otimes n}$ is the uniform superposition over all n-bit strings.

Note that for any n-qubit unitary U,

$$UWU^\dagger = U(2|\Psi\rangle\langle\Psi| - I)U^\dagger = 2U|\Psi\rangle\langle\Psi|U^\dagger - I$$

so $UWU^\dagger$ is a reflection around $U|\Psi\rangle$. If we take $U = H^{\otimes n}$ then

$$H^{\otimes n} W H^{\otimes n} = 2|0^n\rangle\langle 0^n| - I$$

since $H|+\rangle = |0\rangle$, where $|0^n\rangle \equiv |0\rangle^{\otimes n}$. Further notice that

$$X^{\otimes n}(H^{\otimes n} W H^{\otimes n})X^{\otimes n} = 2|1^n\rangle\langle 1^n| - I$$

Doing everything in reverse, we can express W as follows:

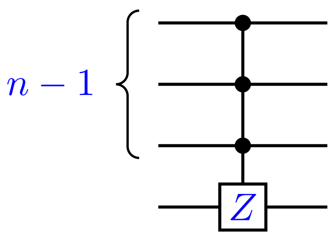
$$W = -H^{\otimes n}\big(X^{\otimes n}\mathrm{C}_{n-1}(Z)X^{\otimes n}\big)H^{\otimes n}$$

Implementing multiple-controlled Z

What remains is to implement the $(n-1)$-fold controlled Z operation

$$\mathrm{C}_{n-1}(Z) = I - 2|11\ldots1\rangle\langle 11\ldots1|$$

that reflects around the final standard basis vector.



$$\mathrm{C}_{n-1}(Z) = \begin{pmatrix} 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \\ 0 & 0 & \cdots & 0 & -1 \end{pmatrix}$$

Note that $\mathrm{C}_1(X) = \mathrm{CNOT}$ is the controlled NOT while $\mathrm{C}_1(Z)$ is the controlled Z gate. Also note that $HZH = X$, so we can implement $\mathrm{C}_1(Z)$ using H and CNOT:

$$\mathrm{C}_1(Z) = (I \otimes H)\mathrm{CNOT}(I \otimes H)$$

$\mathrm{C}_{n-1}(Z)$ can be implemented using $O(n)$ $\mathrm{Toffoli}$ and $\mathrm{C}_1(Z)$ gates, using some extra workspace qubits.

Quantum speed-up

For classical algorithms, searching an unstructured space of size N requires at least $\Omega(N)$ calls to the black box function f to identify the unique solution.

Grover's algorithm demonstrates that for certain problems a quantum algorithm can beat any classical algorithm.

It is possible to show an $\Omega(\sqrt{N})$ lower bound for the number of calls to U_f (or V_f) by any quantum algorithm that identifies a unique solution.

Grover's algorithm does not allow quantum computers to solve NP-complete problems in polynomial time. It can only provide a polynomial speed-up!

Summary

- **Grover's problem:** given access to f, find x such that $f(x) = 1$ (equivalent to brute-force search)
- **Black box function:** $U_f|x\rangle|y\rangle = |x\rangle|y \oplus f(x)\rangle$, it can recognize a solution but may not know it
- **Phase oracle:** $V_f|x\rangle = (-1)^{f(x)}|x\rangle$; you get it from U_f by putting $|-\rangle$ in the last register; $V_{f_a} = I - 2|a\rangle\langle a|$ when $f_a(x) = 1$ iff $x = a$
- **Reflection:** $I - 2|v\rangle\langle v|$ is a reflection around vector $|v\rangle$
- **Grover iterate:** $G = WV$ where $W = 2|\Psi\rangle\langle\Psi| - I$ and $|\Psi\rangle = |+\rangle^{\otimes n}$; V is the phase oracle for f_a for some unknown a
- **Grover's algorithm:** $G^{\sqrt{N}}|+\rangle^{\otimes n}$ where $N = 2^n$
- **Grover's rotation:** two reflections make a rotation!
- **Complexity:** $O(\sqrt{N})$ iterations suffice to find the unique solution with probability $1 - 1/N$; for M solutions, $O(\sqrt{N/M})$ iterations suffice to find a random solution with probability $1 - M/N$
- **Implementation:** $W = -H^{\otimes n}(X^{\otimes n}\mathrm{C}_{n-1}(Z)X^{\otimes n})H^{\otimes n}$ where $\mathrm{C}_{n-1}(Z)$ is the $(n-1)$-fold controlled Z gate