Grover's Algorithm

Poll

Search Problem

Let $f : \{0,1\}^n \to \{0,1\}$ and say we want to find an x such that $f(x) = 1$, or conclude no such x exists.

Example: Determining if a formula is satisfiable is an instance of such a problem

$$(x_1 \vee \neg x_2 \vee x_3) \wedge (\neg x_1 \vee \neg x_2 \vee x_3) \wedge (x_1 \vee \neg x_2 \vee \neg x_3) \wedge (\neg x_1 \vee x_2 \vee \neg x_3)$$

Function f evaluates to truth value of an assignment $x = x_1 x_2 x_3$.

Search Problem

Let $f : \{0,1\}^n \to \{0,1\}$ and say we want to find an x such that $f(x) = 1$, or conclude no such x exists.

Example: Determining if a formula is satisfiable is an instance of such a problem

$$(x_1 \vee \neg x_2 \vee x_3) \wedge (\neg x_1 \vee \neg x_2 \vee x_3) \wedge (x_1 \vee \neg x_2 \vee \neg x_3) \wedge (\neg x_1 \vee x_2 \vee \neg x_3)$$

This formula is satisfied by $x_1 = 0, x_2 = 0, x_3 = 0$.

Given such a formula it is a hard problem in general to tell if it is satisfiable!

Search Problem

Let $f : \{0,1\}^n \to \{0,1\}$ and say we want to find an x such that $f(x) = 1$, or conclude no such x exists.

Classically a randomized algorithm requires $\Omega(2^n)$ evaluations of f to solve this problem.

In 1996 Grover gave a quantum algorithm that can solve this problem with $O(\sqrt{2^n})$ evaluations of f .

This is "only" a quadratic speedup, but for a problem with tons of applications.

We are first going to study this problem with the promise that there is exactly one x such that $f(x) = 1$.

Example: Let's start with the case $f : \{0,1\}^2 \to \{0,1\}$.

input	output
00	1
01	0
10	0
11	0

input	output
00	0
01	0
10	1
11	0

Possible truth tables for f.

input	output
00	0
01	1
10	0
11	0

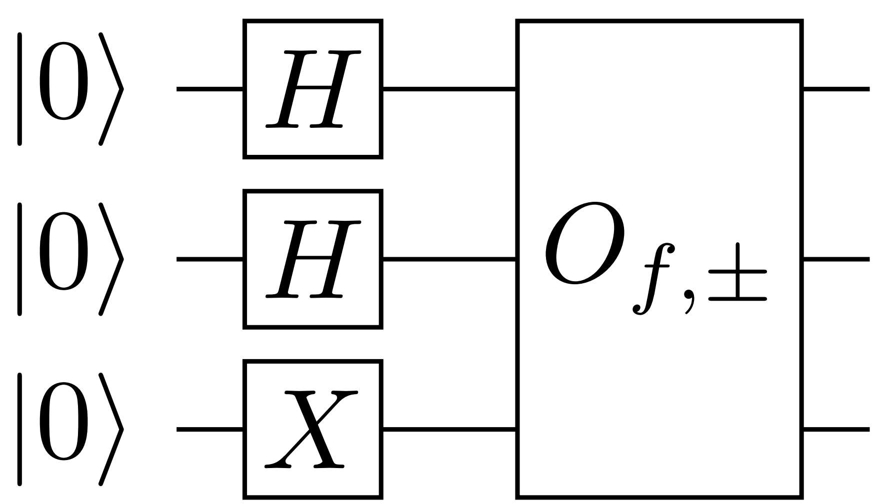
input	output
00	0
01	0
10	0
11	1

First Algorithm

Let's say we have a phase oracle $O_{f,\pm}$ for $f : \{0,1\}^2 \to \{0,1\}$.

$$O_{f,\pm}|x\rangle|b\rangle = (-1)^{b \cdot f(x)}|x\rangle|b\rangle$$

We begin in the familiar way:



$$\sum_{x \in \{0,1\}^2} (-1)^{f(x)}|x\rangle|1\rangle$$

First Algorithm

$$\sum_{x \in \{0,1\}^2} (-1)^{f(x)} |x\rangle |1\rangle$$

Depending on what f is, we are now in one of the 4 states (ignoring normalization and last register).

$$\begin{bmatrix} -1 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ -1 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ -1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \\ -1 \end{bmatrix}$$

These states are orthogonal! We can distinguish them with certainty by a measurement.

Concretely, let's apply the unitary

$$R = \frac{1}{2}\begin{bmatrix} -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \end{bmatrix}$$

$$R \sum_{x \in \{0,1\}^2} (-1)^{f(x)} |x\rangle = \begin{cases} |00\rangle & \text{if } f(00) = 1 \\ |01\rangle & \text{if } f(01) = 1 \\ |10\rangle & \text{if } f(10) = 1 \\ |11\rangle & \text{if } f(11) = 1 \end{cases}$$

Thus measuring in the computational basis we identify f with certainty after a single query.

Implementing R

$$\frac{1}{2}\begin{bmatrix} -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \end{bmatrix}$$

We see that $R = 2|u\rangle\langle u| - \mathbb{I}$ where $|u\rangle$ is the uniform superposition.

As $R = H^{\otimes 2}(2|00\rangle\langle 00| - \mathbb{I})H^{\otimes 2}$ it suffices to implement

$$2|00\rangle\langle 00| - \mathbb{I}$$

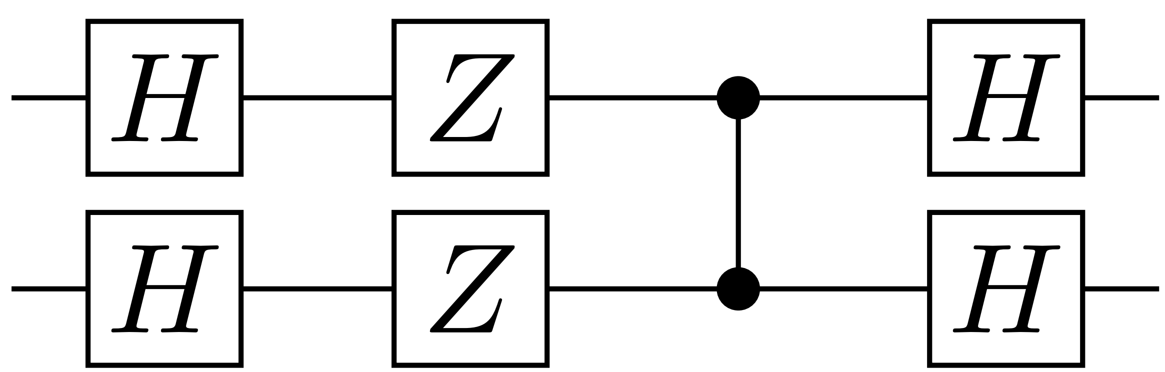
Implementing R

$$\frac{1}{2}\begin{bmatrix} -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \end{bmatrix}$$

As $R = H^{\otimes 2}(2|00\rangle\langle 00| - \mathbb{I})H^{\otimes 2}$ it suffices to implement

$$2|00\rangle\langle 00| - \mathbb{I}$$

We want to multiply every comp. basis vector by -1 except for $|00\rangle$.



Grover

Now let's consider $f : \{0,1\}^n \to \{0,1\}$ with the promise there is a unique x with $f(x) = 1$. Let $N = 2^n$.

We will call this x the "marked element".

When x is the marked element we denote the phase oracle by O_x .

We will also drop the last (control) register of the oracle.

$$O_x|y\rangle = \begin{cases} -|y\rangle & \text{if } y = x \\ |y\rangle & \text{otherwise} \end{cases}$$

Two important states

$$O_x|y\rangle = \begin{cases} -|y\rangle & \text{if } y = x \\ |y\rangle & \text{otherwise} \end{cases}$$

The oracle $O_x = \mathbb{I} - 2|x\rangle\langle x|$ is reflection about the plane orthogonal to $|x\rangle$.

The other important state for the alg. is the uniform superposition $|u\rangle = H^{\otimes n}|0^n\rangle$.

We begin in $|u\rangle$ and want to get close to $|x\rangle$.

The whole alg. takes place in this 2D plane.

Grover Iterate

The Grover iterate is the product of the two reflections

$$G = (2|u\rangle\langle u| - \mathbb{I})(\mathbb{I} - 2|x\rangle\langle x|)$$

The Grover iterate can be implemented with one query and $O(n)$ many other gates.

The algorithm is $G^k H^{\otimes n}|0^n\rangle$ for a smartly chosen k .

Geometric Picture

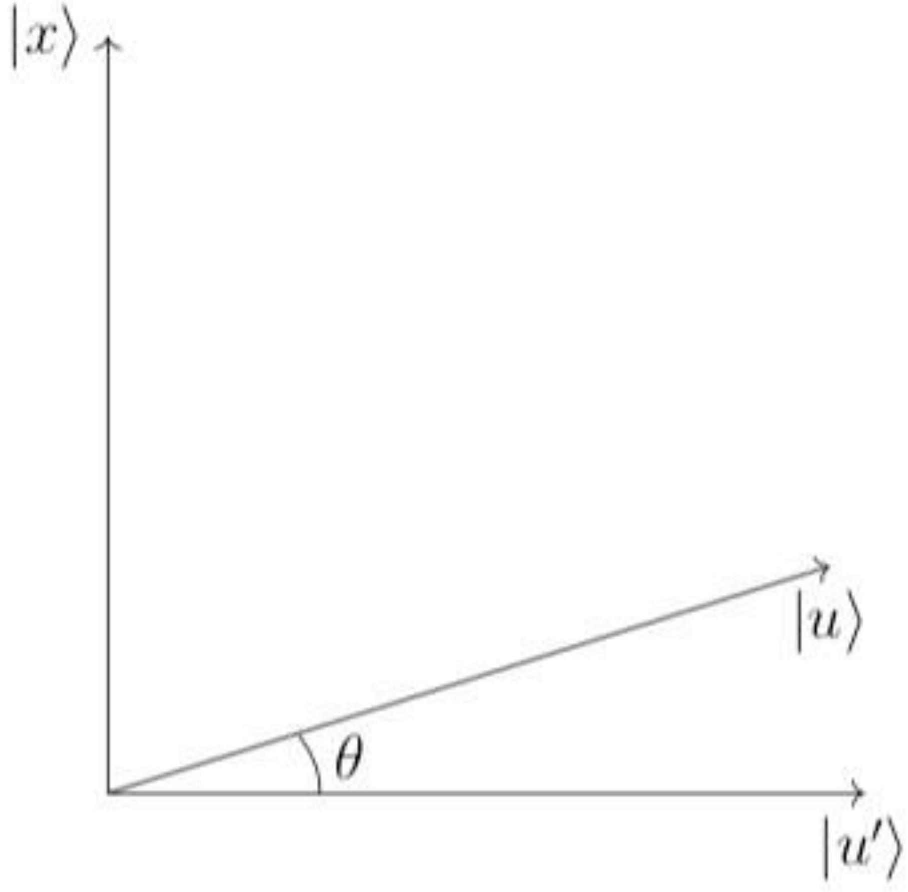
Let $|u'\rangle$ be the uniform superposition over the unmarked strings

$$|u'\rangle = \frac{1}{\sqrt{N-1}} \sum_{\substack{y\in\{0,1\}^n \\ y\neq x}} |y\rangle$$

We have $\langle x|u'\rangle = 0$ and

$$|u\rangle = \frac{1}{\sqrt{N}}(\sqrt{N-1}|u'\rangle + |x\rangle)$$

Geometric Picture



$$|u\rangle = \frac{1}{\sqrt{N}}(\sqrt{N-1}|u'\rangle + |x\rangle)$$

What is the angle θ ?

$$\langle u|u'\rangle = \sqrt{1 - 1/N} = \cos(\theta)$$

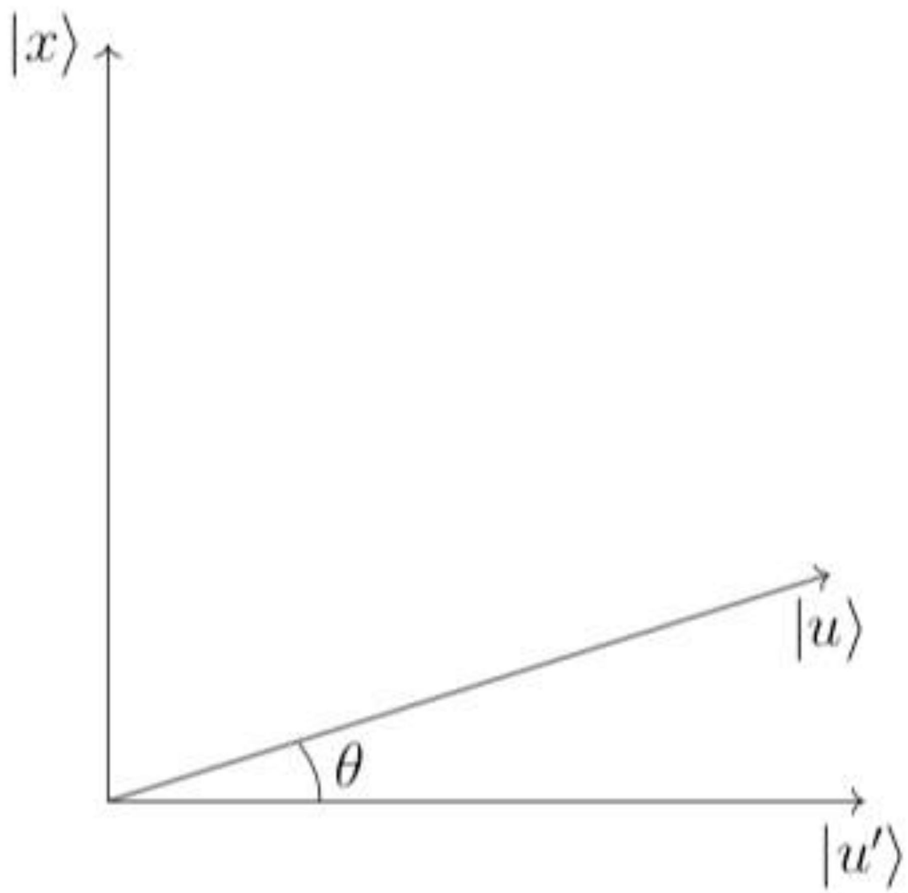
$$\sin(\theta) = \sqrt{1/N}$$

$$\theta = \arcsin(\sqrt{1/N})$$

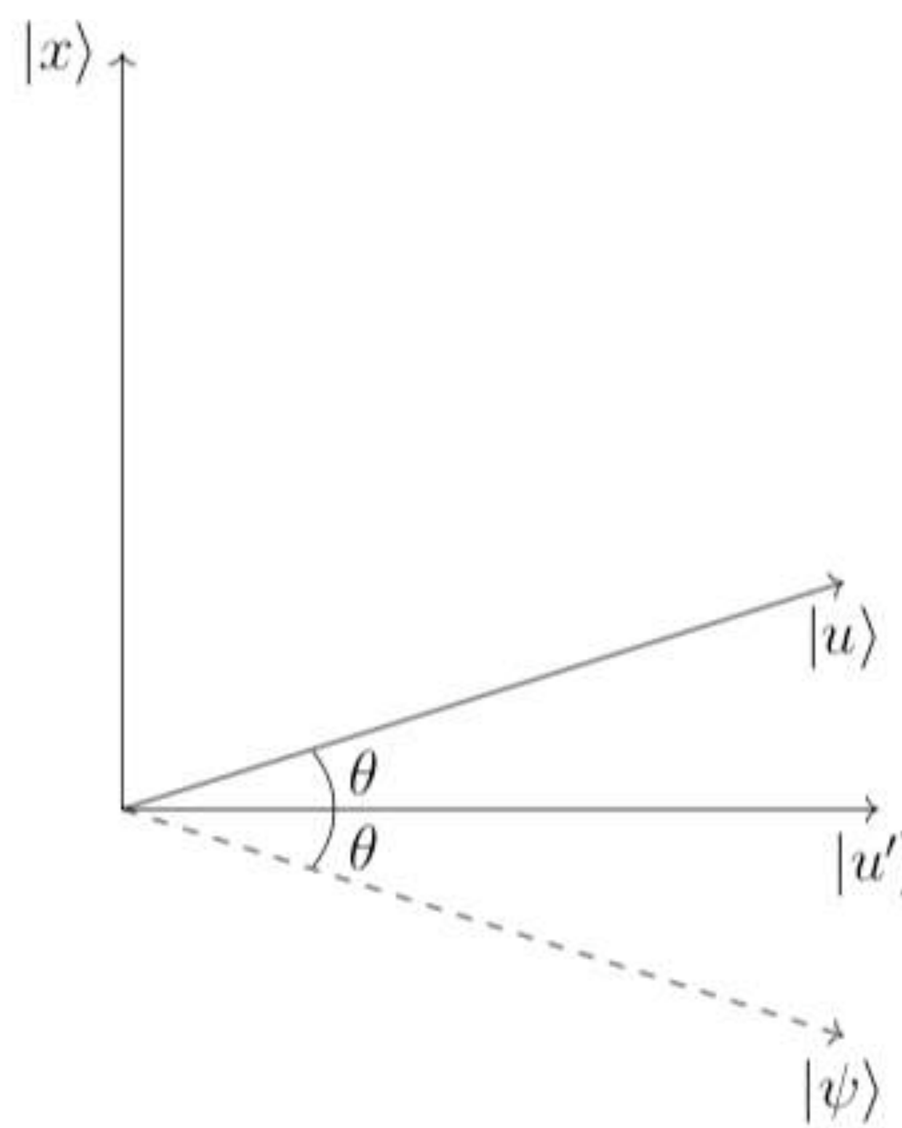
Geometric Picture

$$G = (2|u\rangle\langle u| - \mathbb{I})(\mathbb{I} - 2|x\rangle\langle x|)$$

What does the Grover iterate do to $|u\rangle$?



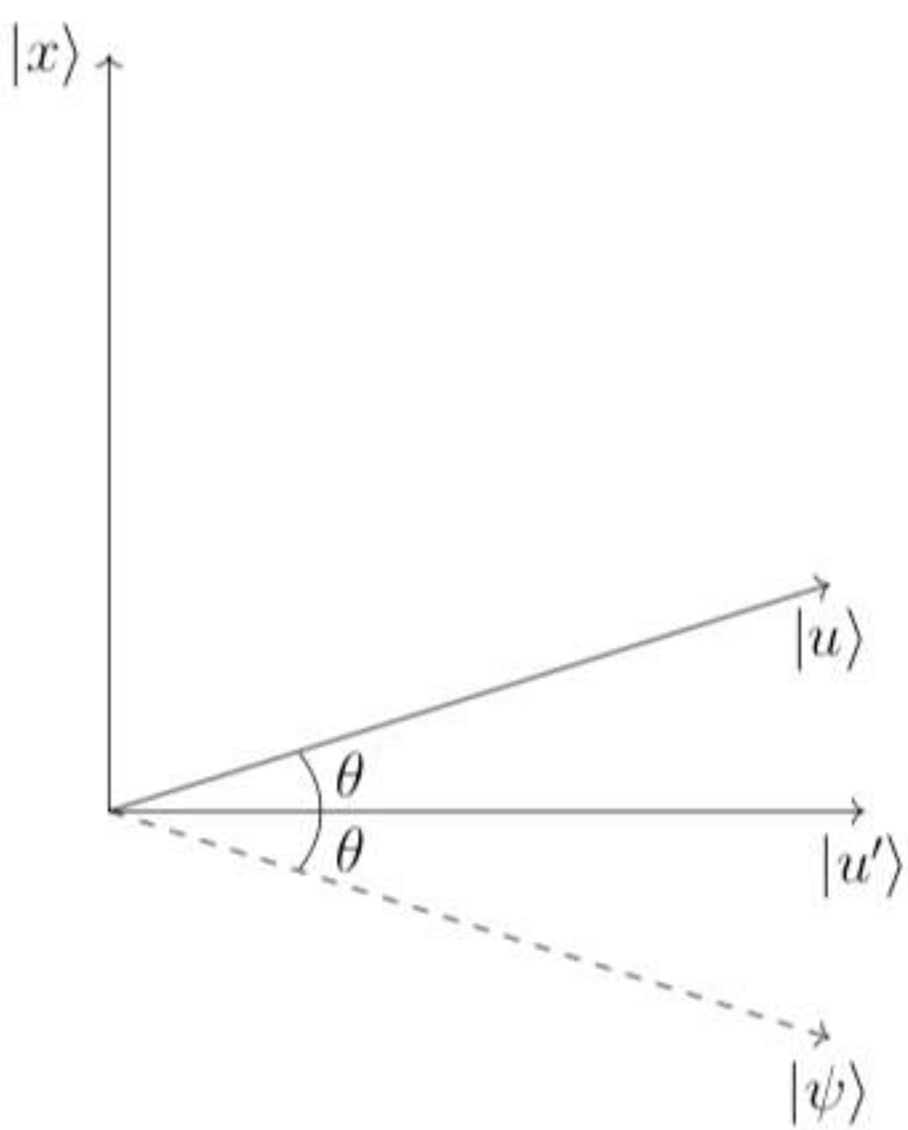
$$\mathbb{I} - 2|x\rangle\langle x|$$



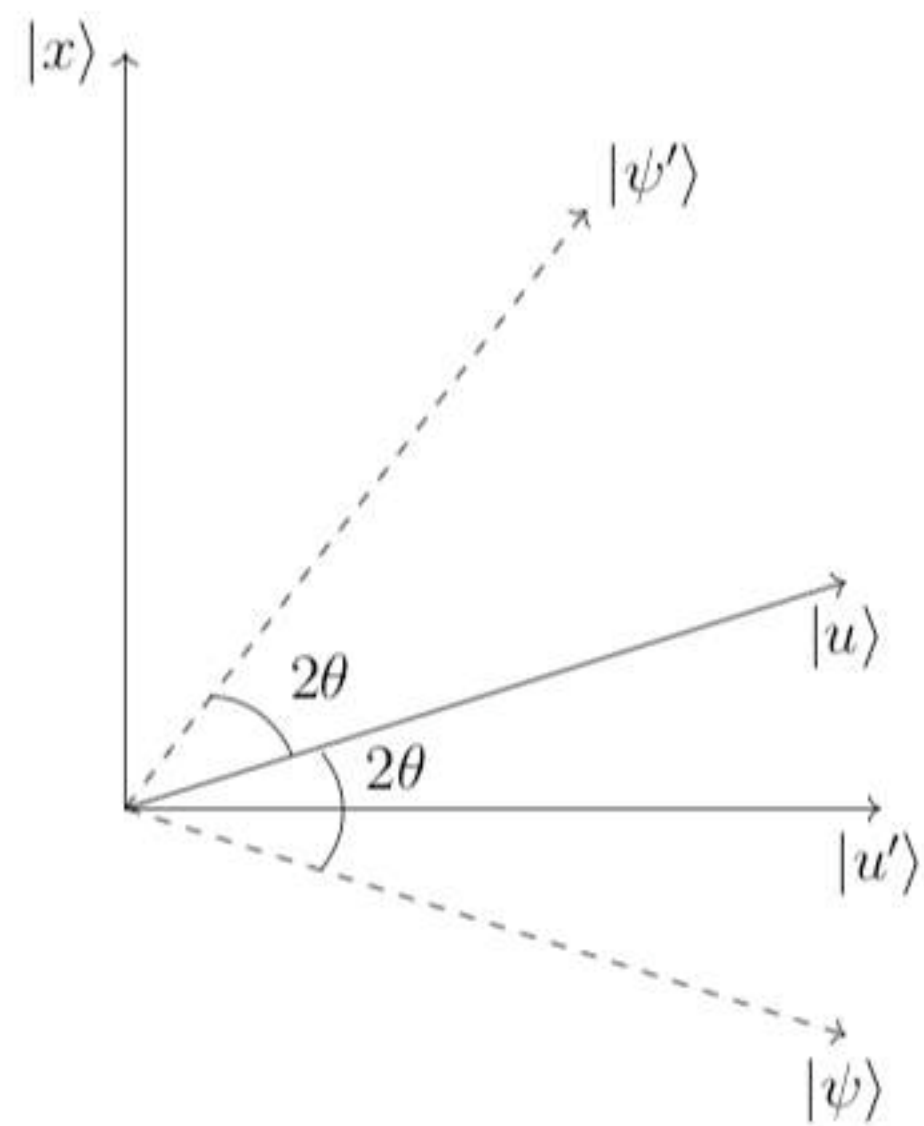
Geometric Picture

$$G = (2|u\rangle\langle u| - \mathbb{I})(\mathbb{I} - 2|x\rangle\langle x|)$$

What does the Grover iterate do to $|u\rangle$?



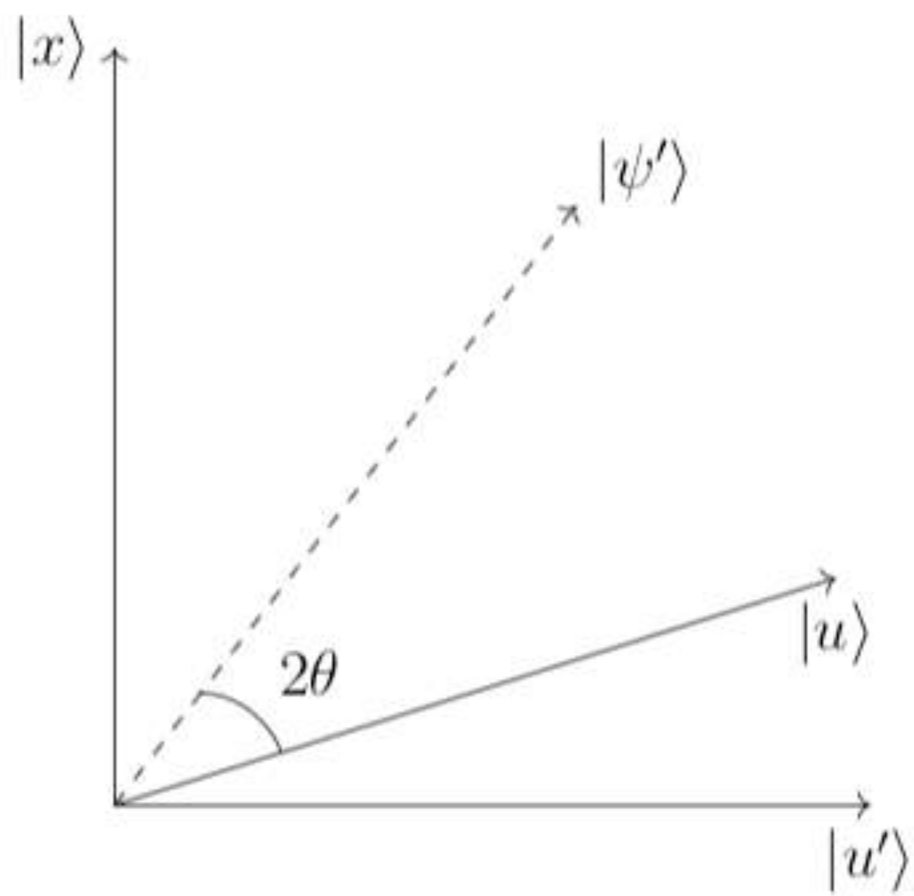
$$2|u\rangle\langle u| - \mathbb{I}$$



Geometric Picture

$$G = (2|u\rangle\langle u| - \mathbb{I})(\mathbb{I} - 2|x\rangle\langle x|)$$

What does the Grover iterate do to $|u\rangle$?



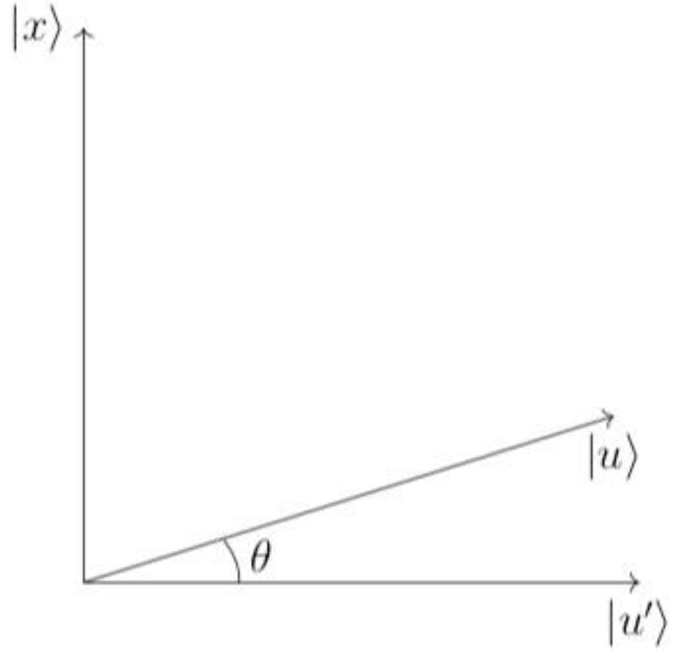
G rotates $|u\rangle$ by an angle of 2θ .

Geometric Picture

Let's look at the action of G in the 2D plane spanned by $|u'\rangle$ and $|x\rangle$.

$$G = (2|u\rangle\langle u| - \mathbb{I})(\mathbb{I} - 2|x\rangle\langle x|)$$

$$\begin{bmatrix}\cos(\theta) & -\sin(\theta)\\ \sin(\theta) & \cos(\theta)\end{bmatrix}\begin{bmatrix}1 & 0\\ 0 & -1\end{bmatrix}\begin{bmatrix}\cos(\theta) & \sin(\theta)\\ -\sin(\theta) & \cos(\theta)\end{bmatrix} \qquad \begin{bmatrix}1 & 0\\ 0 & -1\end{bmatrix}$$



Multiplying these out + trig identities gives

$$\begin{bmatrix}\cos(2\theta) & -\sin(2\theta)\\ \sin(2\theta) & \cos(2\theta)\end{bmatrix}$$

Success Probability

$$G^k|u\rangle = \cos((2k+1)\theta)|u'\rangle + \sin((2k+1)\theta)|x\rangle$$

The success probability measuring after k iterations is

$$\sin^2((2k+1)\theta)$$

The ideal choice is $\bar{m} = \dfrac{\pi}{4\theta} - \dfrac{1}{2}$ but this may not be an integer.

Take $m = \left\lfloor \dfrac{\pi}{4\theta} \right\rfloor$. Then $|m - \bar{m}| \leq \dfrac{1}{2}$.

Poll

Success Probability

The ideal choice is $\bar{m} = \dfrac{\pi}{4\theta} - \dfrac{1}{2}$ but this may not be an integer.

Take $m = \left\lfloor \dfrac{\pi}{4\theta} \right\rfloor$. Then $|m - \bar{m}| \leq \dfrac{1}{2}$.

$$\begin{aligned} \sin((2m+1)\theta) &\geq \sin(\pi/2 - \theta) \\ &= \cos(\theta) = \sqrt{1 - 1/N} \end{aligned}$$

With this choice the success probability is $1 - 1/N$.

Complexity

$$\sin(\theta) = \frac{1}{\sqrt{N}} \leq \theta$$

We do $m = \left\lfloor \frac{\pi}{4\theta} \right\rfloor \leq \frac{\pi}{4}\sqrt{N}$ applications of the Grover iterate.

$$G = (2|u\rangle\langle u| - \mathbb{I})(\mathbb{I} - 2|x\rangle\langle x|)$$

Each iteration can be performed with one query and $O(n)$ other gates.

We can achieve success probability $1 - 1/N$ after $O(\sqrt{N})$ queries and a quantum circuit of size $O(\sqrt{N}\log(N))$.

Notes

We only assumed N is a power of 2 in order to create the uniform superposition using Hadamards.

For arbitrary N one can use an approximate Fourier transform F_N to create the uniform superposition $F_N|0\rangle$.

Everything else goes through as before.

I think of Grover's algorithm searching a "database" represented by $z \in \{0,1\}^N$.

Activity

Think about the following two questions

1) How can you construct a quantum circuit to implement $2|u\rangle\langle u| - \mathbb{I}$ with $O(n)$ gates?

2) The algorithm we have given assumes there is a unique marked element. Use this algorithm as a black box to solve the general problem with $O(\sqrt{N})$ many quantum queries.

Removing Uniqueness

More marked elements

Again suppose we have phase oracle access to a function $f : \{0,1\}^n \to \{0,1\}$ and want to find a "marked element" x such that $f(x) = 1$.

$f^{-1}(1)$

Suppose that $T = |\{x : f(x) = 1\}|$ and that T is known.

A fairly straightforward generalization of Grover's algorithm now finds a marked element after $O(\sqrt{2^n/T})$ many queries.

Setup

Let $N = 2^n$. Now we take as our basis vectors

$$|\psi_{\mathrm{good}}\rangle = \frac{1}{\sqrt{T}} \sum_{x \in f^{-1}(1)} |x\rangle \qquad |\psi_{\mathrm{bad}}\rangle = \frac{1}{\sqrt{N-T}} \sum_{x \in f^{-1}(0)} |x\rangle$$

The phase oracle is $O_f = \mathbb{I} - 2|\psi_{\mathrm{good}}\rangle\langle\psi_{\mathrm{good}}|$.

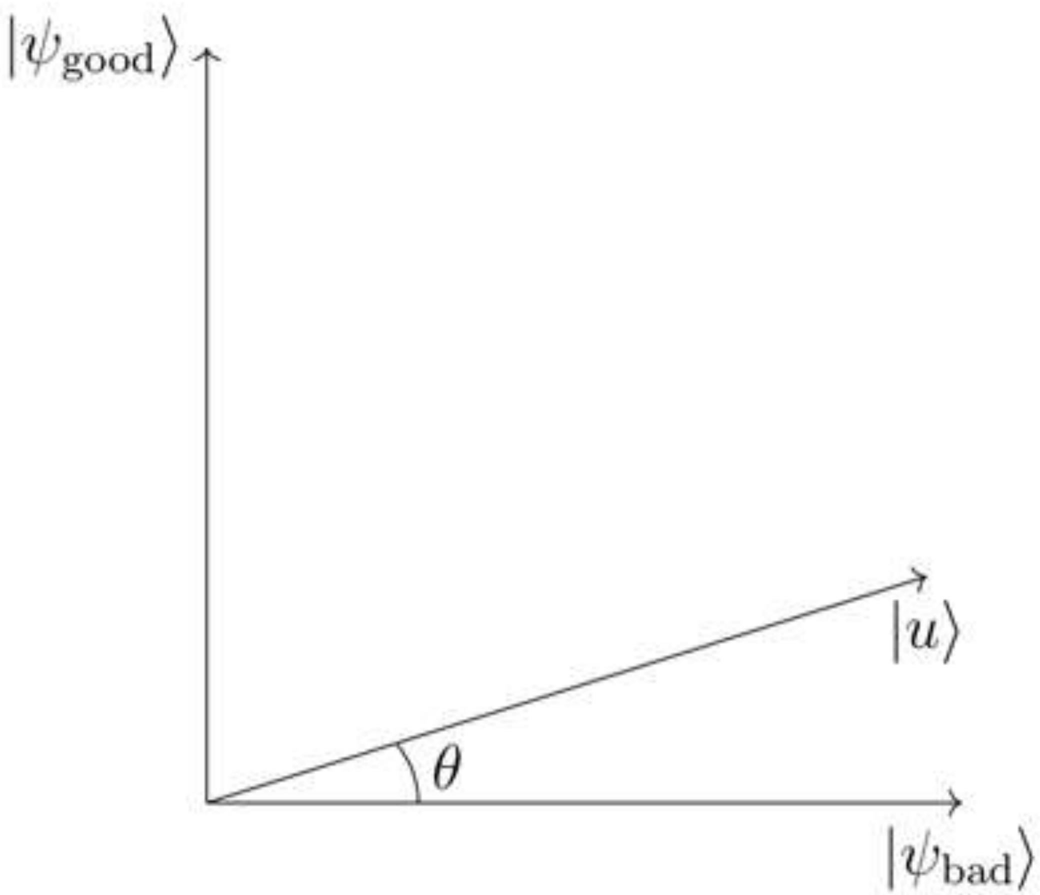
The Grover iterate is as before

$$G = (2|u\rangle\langle u| - \mathbb{I})(\mathbb{I} - 2|\psi_{\mathrm{good}}\rangle\langle\psi_{\mathrm{good}}|)$$

where $|u\rangle = H^{\otimes n}|0^n\rangle$.

Algorithm

The algorithm is again $G^k H^{\otimes n}|0^n\rangle$ for an appropriate value of k.



G is rotation by 2θ

where $\cos(\theta) = \langle u|\psi_{\text{bad}}\rangle$
$$= \sqrt{1 - T/N}$$

To maximize $\sin^2((2k+1)\theta)$

take $k = \left\lfloor \frac{\pi}{4\theta} \right\rfloor \leq \frac{\pi}{4}\sqrt{N/T}$

Conclusion

Let $f:\{0,1\}^n \to \{0,1\}$ with $|f^{-1}(1)| = T$ and assume that T is known.

There is a quantum algorithm to find a marked element after $O(\sqrt{N/T})$ many queries to f and circuit size $O(\sqrt{N/T}\log(N))$.

Application: Collision

Let $f : \{0,1\}^n \to \{0,1\}^n$ with the promise that f is either 2-to-1 or 1-to-1. Let $N = 2^n$.

Determine which one is the case.

2-to-1: for every $z \in \mathrm{range}(f)$ there are exactly two $x, y \in \{0,1\}^n$ with $f(x) = f(y) = z$.

Where have we seen a 2-to-1 function before?

Application: Collision

Where have we seen a 2-to-1 function before?

Simon's problem! This could be solved with only $O(n)$ queries but the function had a lot of additional structure.

Can you use Grover's algorithm to solve the collision problem with $O(\sqrt{N})$ queries?

Activity

We can actually do better!

Think about how to use Grover's algorithm to solve the collision problem with $O(N^{1/3})$ queries to f .