Introduction to **Quantum Algorithms**

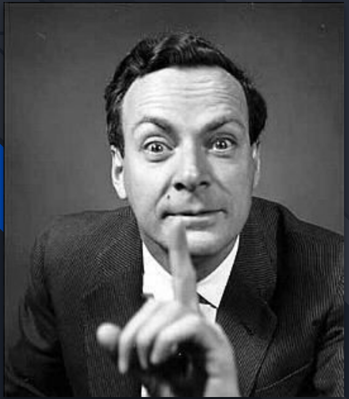
Paramjot Singh

Overview

- Introduction
- Data Encoding
- Quantum Circuits
- Simon’s Algorithm
- Grover’s Algorithm
- Future and Conclusion
- Useful Resources

What is **Q**uantum Computing ?

“I think I can safely say that nobody understands quantum mechanics”
- Richard Feynman



Quantum computer is a machine that performs operations based on the laws of quantum mechanics, such as superposition, entanglement, etc.

HOW'S YOUR QUANTUM COMPUTER PROTOTYPE COMING ALONG?
GREAT!
Dilbert.com DilbertCartoonist@gmail.com
THE PROJECT EXISTS IN A SIMULTANEOUS STATE OF BEING BOTH TOTALLY SUCCESSFUL AND NOT EVEN STARTED.
4-17-12 © 2012 Scott Adams, Inc./Dist. by Universal Uclick
CAN I OBSERVE IT?
THAT'S A TRICKY QUESTION.

Timeline of **Q**uantum computing

1980 **Yuri Manin** proposed an idea of quantum computing

1981 **Richard Feynman** proposed a basic model for a quantum computer

1982 **Paul Benioff** proposed the first theoretical framework for a quantum computer

1994 **Peter Shor** discovered quantum algorithm to factor large integers quickly.

1996 **Lov Grover**, at Bell Labs, invented quantum database search algorithm.

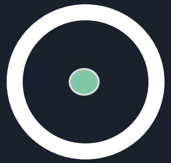
1998 First working **2-qubit NMR** computer demonstrated at UC Berkeley.

2001 First working **7-qubit NMR** computer demonstrated at IBM's Research Center and First execution of Shor's algorithm.

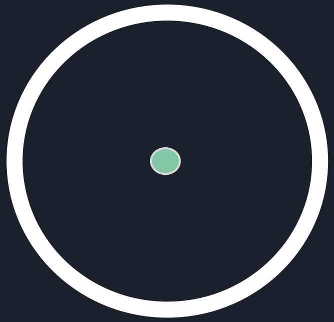
Data encoding & representation

Data Representation

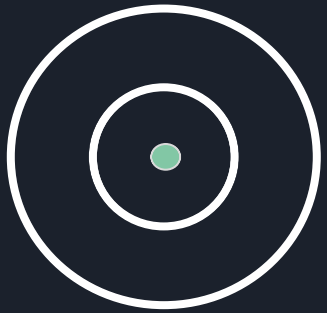
A bit of data is encoded by single electron of atom that is in one of the two states, which is denoted by |0> and |1>, representing the ground and excited states of atom respectively. It’s called qubit



Ground State |0>



Excited State |1>



Superposition α_0 |0> + α_1|1>

Super-position

The superposition principle states that the electron will be in a state (a1 |0> + a2 |1>), the coefficients a1 and a2 represent complex numbers such that $|a1|^2 + |a2|^2 = 1$. a1 and a2 represent the amplitude of the ground and excited states.

-This superposition (a1 |0> + a2 |1>) is the basic unit of encoded information in quantum computers and is called a qubit.

Superposition

The concept of superposition suggests that the electron cannot decide if it’s in the excited or ground state. It is tempting to think of the amplitudes a0 and a1 as probabilities but, we cannot since they can be imaginary. This is one of the aspects of quantum physics that extends beyond our intuitions of the physical world.

To get the state of the electron we need to take the measurement of the electron’s state, this gives us 0 or 1 depending on the state. The outcome of measurement is 0 for probability $|a1|^2$ and 1 for probability $|a2|^2$.

How to perform operations using **Q**ubits?

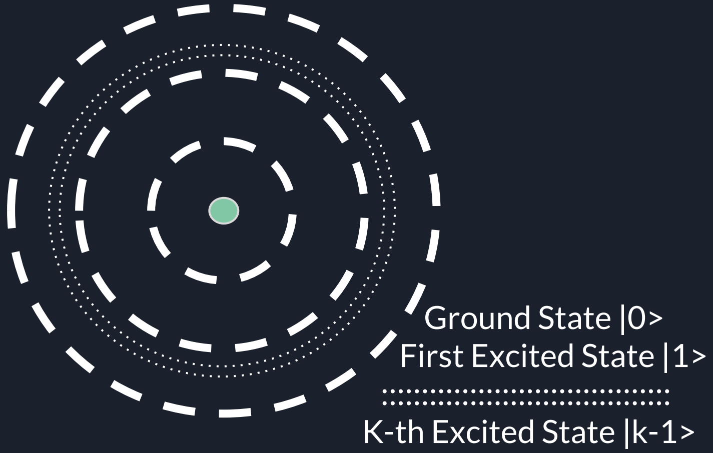
Quantum Circuits

Geometric Interpretation of K-level system

As Laksh explained, Superposition principle says:

$|\Psi> = \alpha_0|0> + \alpha_1|1> + ... + \alpha_{k-1}|k-1>$ and $\alpha_j \in C$

But, interpreting what this state means is not very easy, Since, it's hard to make sense when we say electron is in 1/2 + i/2.

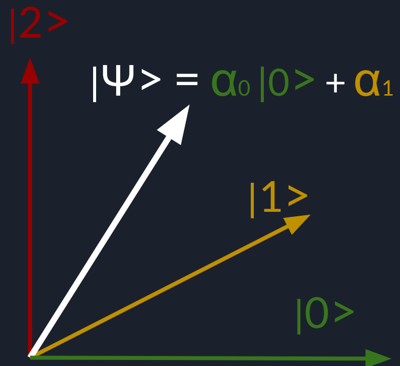


Geometric Interpretation of K-level system

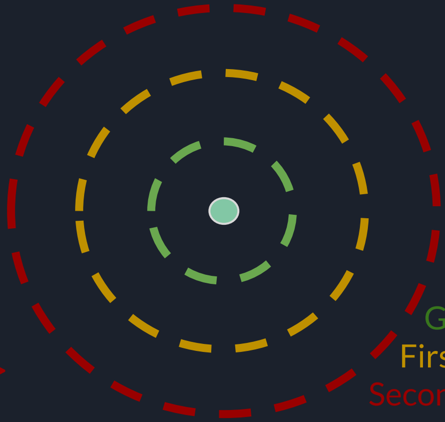
So geometrically superposition principle says,

$$|\Psi> = \alpha_0 |0> + \alpha_1 |1> + \ldots + \alpha_{k-1} |k-1> \quad (\text{Ket notation})$$

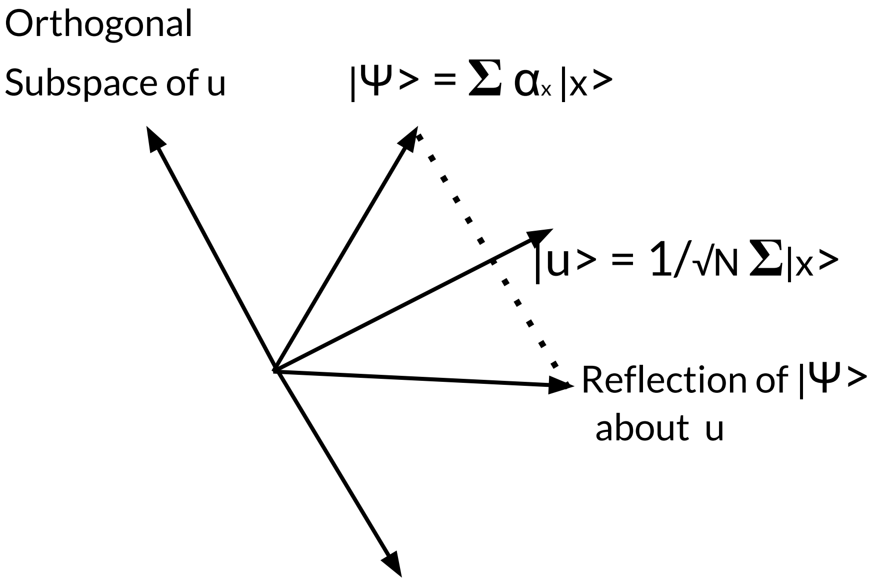
$$= \left| \begin{matrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_{k-1} \end{matrix} \right| \quad (\text{Vector notation})$$



$$|\Psi> = \alpha_0 |0> + \alpha_1 |1> + \alpha_2 |2>$$



Ground State |0>
First Excited State |1>
Second Excited State |2>



Orthogonal

Subspace of u

$|\Psi> = \Sigma\ \alpha_x\ |x>$

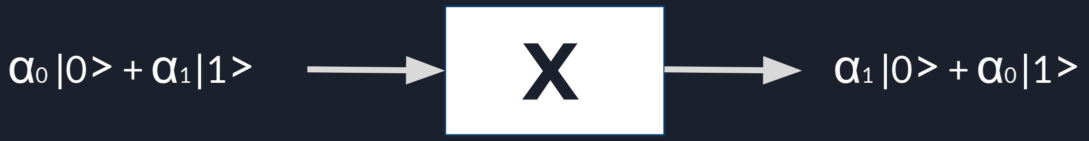
$|u> = 1/\sqrt{N}\ \Sigma |x>$

Reflection of $|\Psi>$

about u

Quantum Gates

- Bit Flip



$$\alpha_0\,|0> + \alpha_1|1> \longrightarrow \boxed{X} \longrightarrow \alpha_1\,|0> + \alpha_0|1>$$

Transformation matrix is

$$X = \begin{vmatrix} 0 & 1 \\ 1 & 0 \end{vmatrix}$$

Quantum Gates

- Phase Flip

$$\alpha_0 |0> + \alpha_1|1> \longrightarrow \boxed{\mathbf{Z}} \longrightarrow \alpha_0 |0> - \alpha_1|1>$$

Transformation matrix is

$$Z = \begin{vmatrix} 1 & 0 \\ 0 & -1 \end{vmatrix}$$

Quantum Gates

- Hadamard transform

$$\alpha_0\,|0> + \alpha_1|1> \longrightarrow \boxed{\mathbf{H}} \longrightarrow \beta_0\,|0> + \beta_1|1>$$

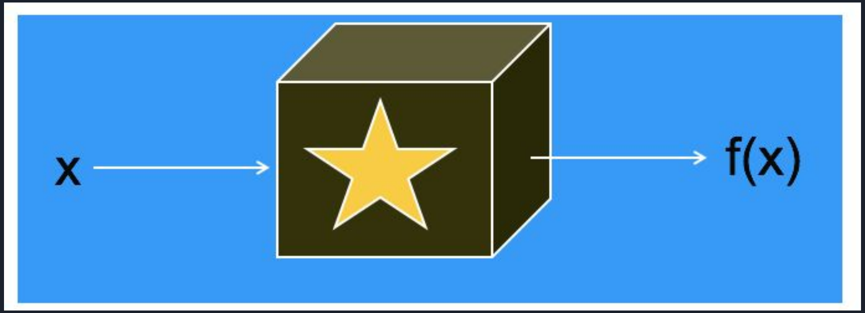
Transformation matrix for Hadamard gate

$$H = \begin{vmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{vmatrix}$$

Demonstration of Quantum Algorithms

Here, we will look into a quantum algorithm to solve the challenge of finding a secret string 's' in a function 'f' that maps n bit string to n bit string such that, $f(x) = f(x+s)$.

Imagine we are given a black box to compute f, until we see a collision or until we don't input two different values of 'x' such that f(x) is same for both, we don't get a structure for 'f'.



Demonstration of Quantum Algorithms

Here’s an outline of the quantum algorithm.

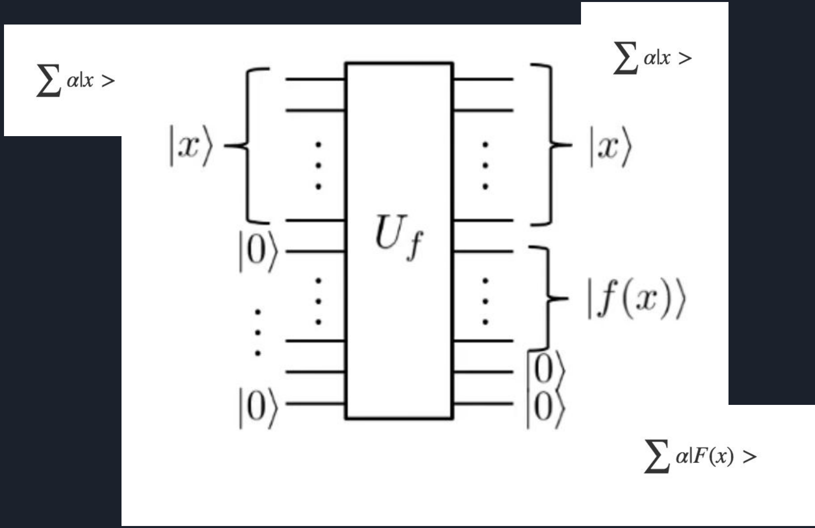
1. Set up random super-position
2. Fourier sampling to get y: y.s = 0(mod 2)
3. Repeat n-1 times to generate n-1 linear equations.
4. Solve for ‘s’

Step 1

-The first step is to produce a random superposition such as: $\frac{1}{\sqrt{2}}|r\rangle + \frac{1}{\sqrt{2}}|r \oplus s\rangle$

-The input to the quantum program is x and 0s and the output is

x, f(x) and 0’s.



-The whole point of quantum computing is to evaluate it over a

Super-position.

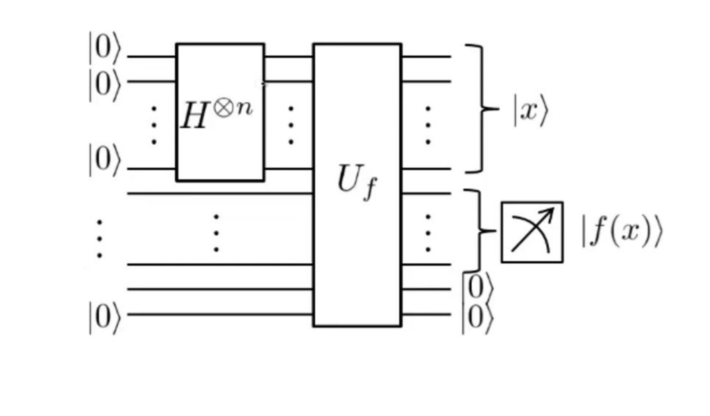
-Now we need to look at the type of superposition

we need to feed in.

Step 1

So here's the idea,

We start with an input of x and 0's and run them through the hadamard transform.

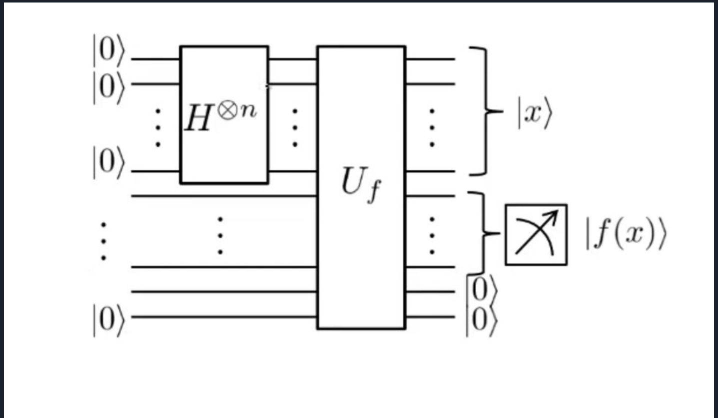


$$\sum \frac{1}{2^{n/2}} |x >$$

Now that we have a uniform superposition over n bit strings, we evaluate f() over this to get the output.

Step 1

Let us say we get some random string ‘a’ = f(r) from the measurement of ‘f’, we know that f is a 1->2 function, this means there is some value f(r) = a and a = f(r+s).

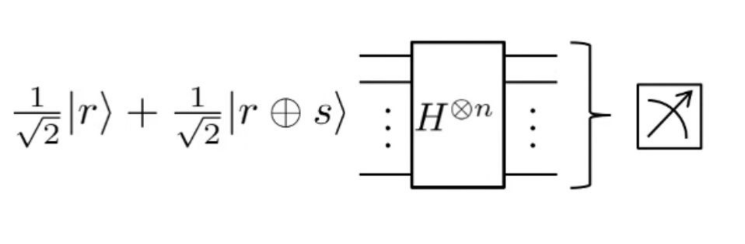


$$\frac{1}{\sqrt{2}}|r\rangle + \frac{1}{\sqrt{2}}|r \oplus s\rangle$$

When we measure the function f, the super-position collapses to the parts consistent with output ‘a’. We need to cross out the values in the super position of x that are no equal to ‘a’ so that we get the final super position as ...

Step 2

The next step is to apply fourier sampling to the super-position.


$\frac{1}{\sqrt{2}}|r\rangle + \frac{1}{\sqrt{2}}|r \oplus s\rangle$
$H^{\otimes n}$

$$\sum \beta|y >$$

The result of the fourier transform is in the form ... . . Next we take the measurement of the output and get some value ‘y’ such that y = 0 mod(2).

Step 3

Now we repeat the sampling n-1 times to get a set of linearly independent equations.

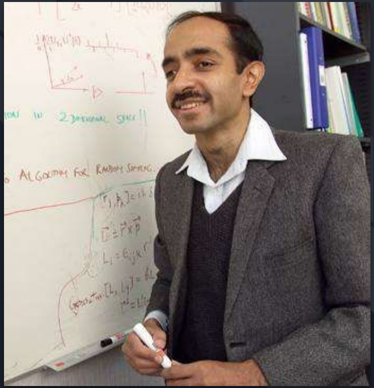
We can solve for x from these equations.

Conclusion

This algorithm was designed in 1994 to demonstrate the use of quantum computing and is called Simon’s algorithm.

A classical algorithm to solve this black box will typically be in O($2^{n/2}$), whereas, a Quantum algorithm can solve this in O(n), this provides an exponential speedup.

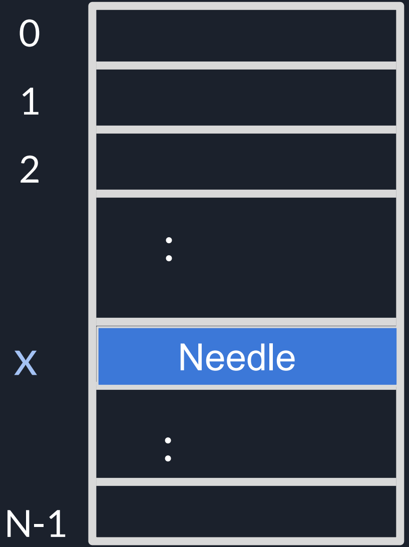
Quantum Search Algorithm



Lov Kumar Grover

Grover’s Algorithm

Digital Haystack



0

1

2

:

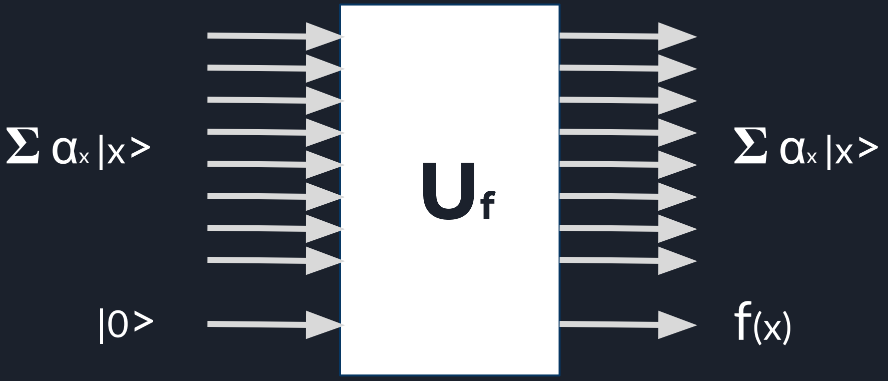
x Needle

:

N-1

Grover's Algorithm

Problem : Given boolean function f : { 0, 1, … , N-1 } ⇒ { 0, 1 }. Find x : f(x) = 1



$\Sigma\ \alpha_x\,|x>\ |0> \longrightarrow \Sigma\ \alpha_x\,|x>\ f(x)$

Grover’s Algorithm

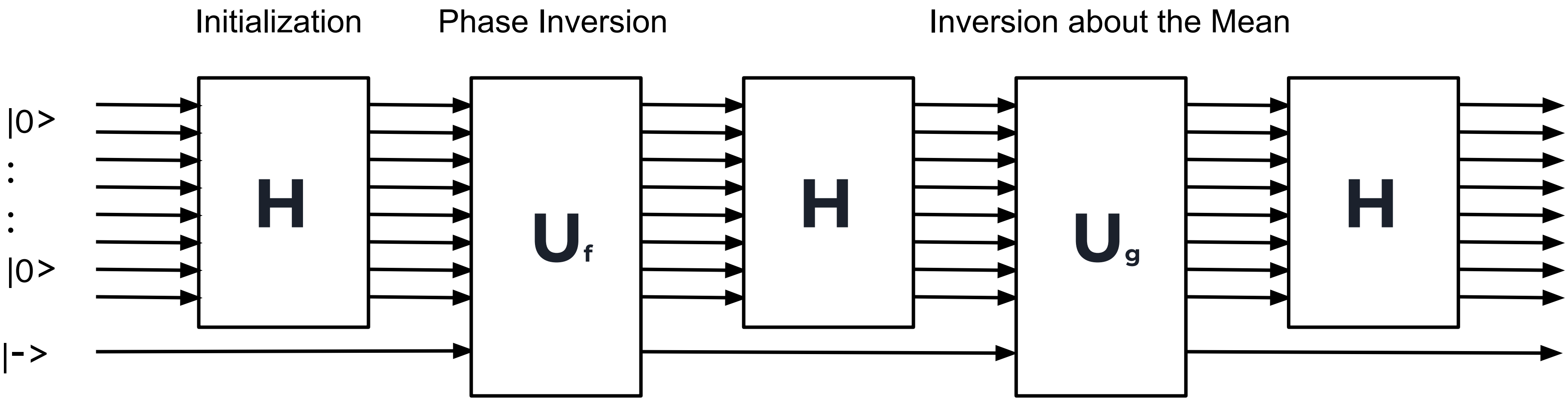
Grover’s Algorithm involves following two steps:

1. Phase Inversion

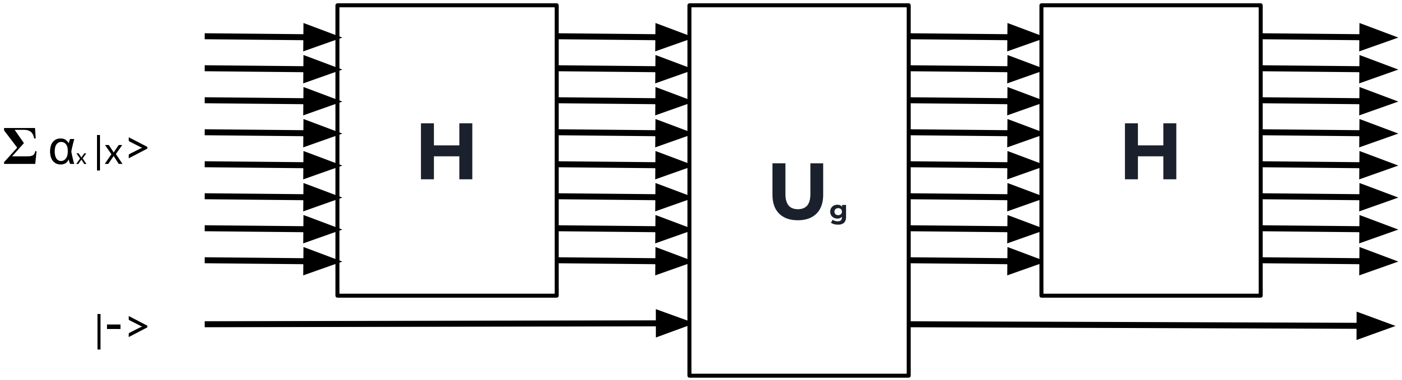
   $\Sigma\, \alpha_x\, |x> \Rightarrow\Rightarrow\Rightarrow \Sigma\, \alpha_x\, |x>$: x is not x^* - $\alpha_{x^*}\, |x^*>$

2. Inversion about the Mean(μ)

   $\Sigma\, \alpha_x\, |x> \Rightarrow\Rightarrow\Rightarrow \Sigma\, (2\mu - \alpha_x)\, |x>$ $\mu = \Sigma\, \alpha_x / N$



Repeating this √N iterations



$\Sigma\ \alpha_x\ |x>$

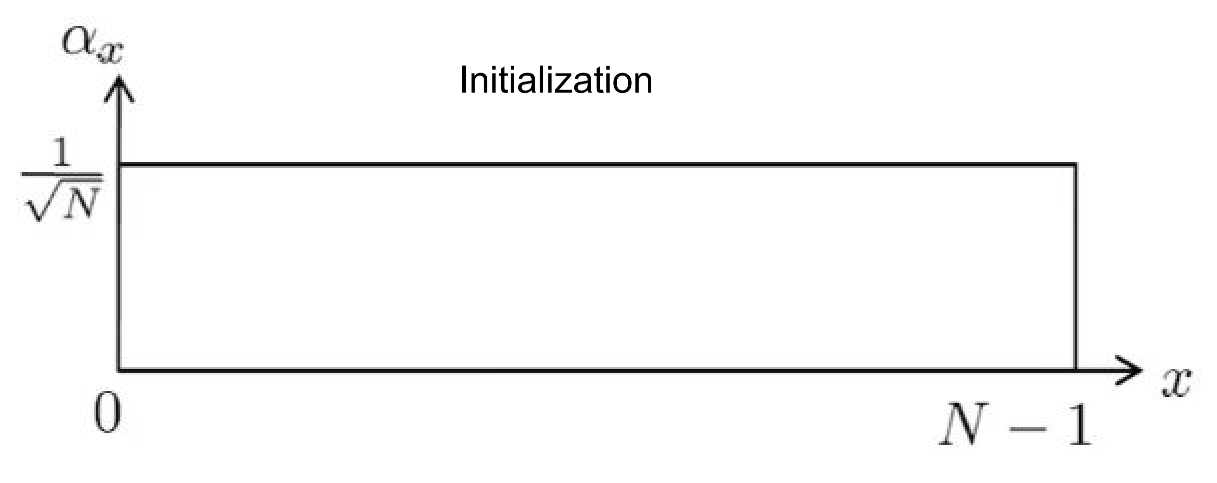
$|->$

H

U_g

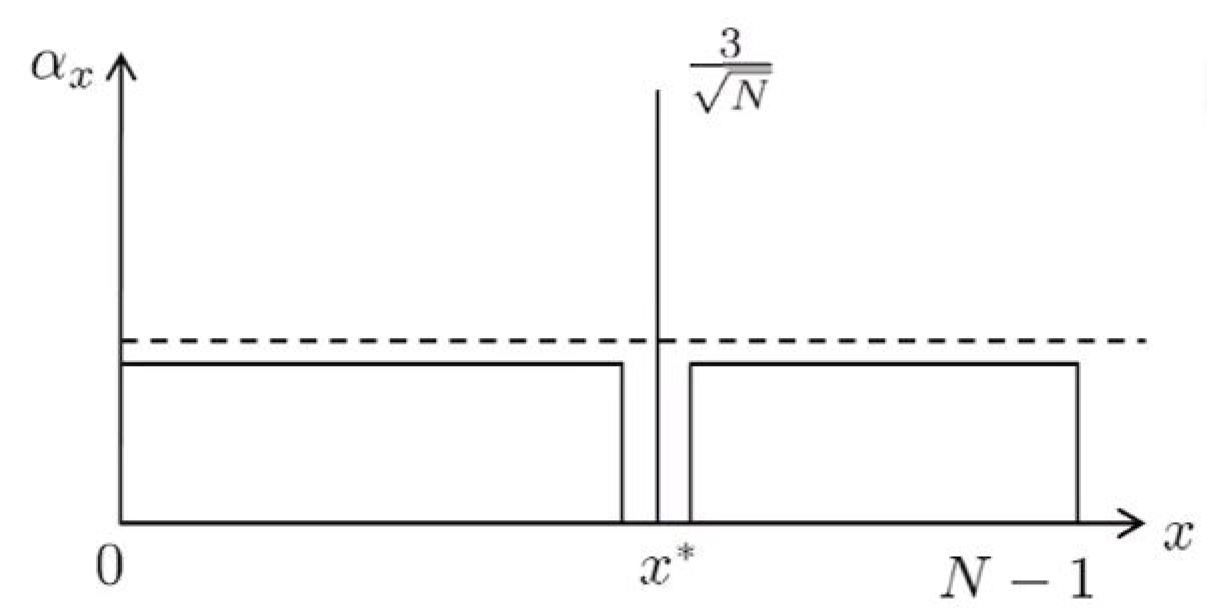
H

DELETE



Iterating √N times we get the amplitude of x* as 1/√2

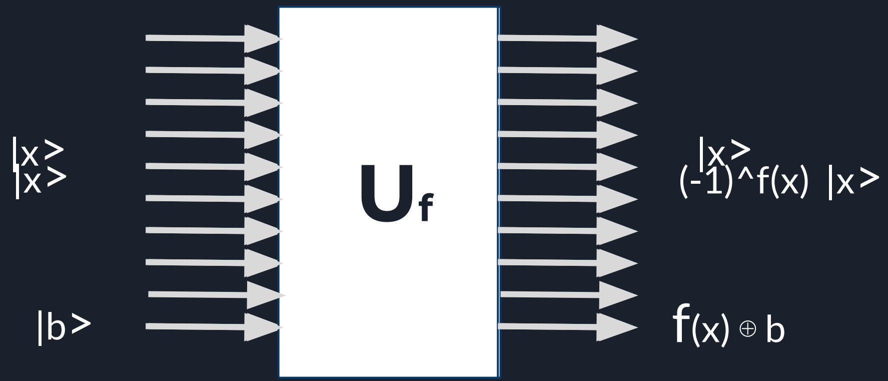
Grover’s Algorithm



Iterating √N times we get the amplitude of x* as 1/√2

Phase Inversion

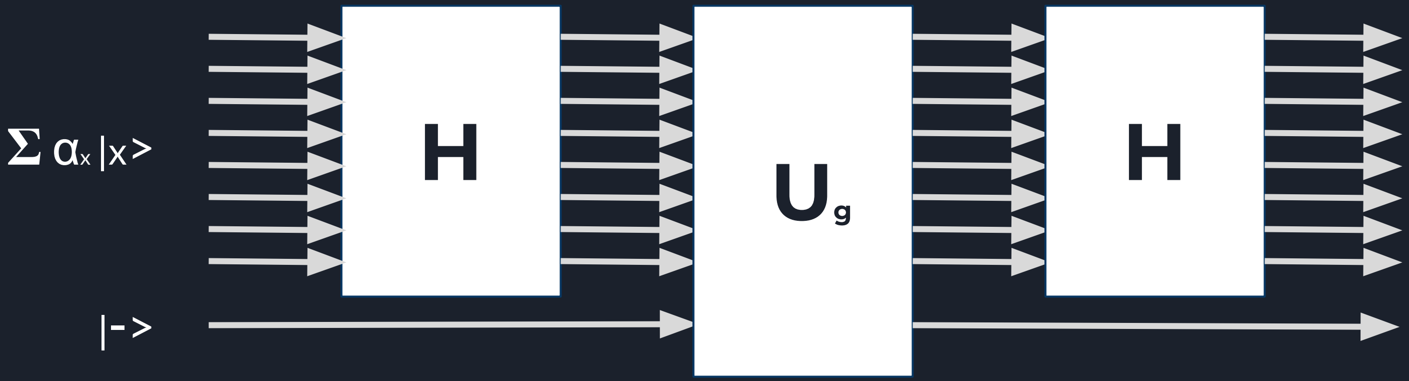
$$\Sigma\, \alpha_x |x> \;\Rightarrow\; \Sigma\, \alpha_x (-1)^{f(x)}\; |x>$$



b \ f(x)	|0>	|1>
|0>	**|0>**	**|1>**
|1>	**|1>**	**|0>**
|->	**|->**	**- |->**

Inversion about the Mean(μ)

$\Sigma\, \alpha_x |x> \;\Rightarrow\; \Sigma\, (2\mu - \alpha_x) |x>$

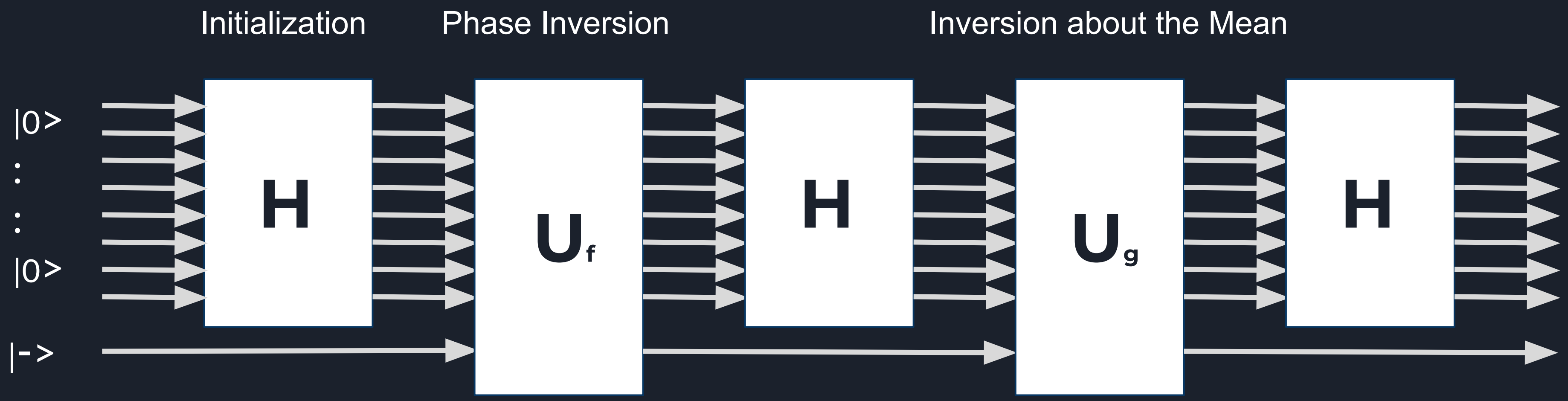


g: { 0, 1 }^n ⇒ { 0, 1 }

If g(0…...0) = 0

Else g(y) = 1 where y is not 0…….0

Grover's Algorithm Quantum Circuit



Repeating this √N iterations

Future and Conclusion

Future of Quantum Computing

Cyber security:

Pros: Quantum key distribution will protect us :) an ultra-secure communication method that requires a key to decipher a message. Thanks to the peculiar properties of quantum mechanics, if the message gets intercepted, no one else can read it.

Cons: Standard encryption like RSA will fail

Traffic Control:

Quantum computers can help to streamline traffic control. They will be able to quickly calculate the optimal routes concurrently which allows for efficient scheduling and would reduce traffic congestion.

Drug Design:
Quantum computers will be able to analyze multiple molecules, proteins and chemicals simultaneously, this will help chemists develop drugs faster than traditional methods.

Conclusion

- What algorithms will be discovered next?
- Can quantum computers solve NP Complete problems in polynomial time?
- Decoherence - the tendency of a quantum computer to decay from a given quantum state into an incoherent state as it interacts with the environment.

Resources

Useful Resources

- David Deutsch’s Lectures
  http://quiprocone.org/Protected/DD_lectures.htm
- Umesh Vajirani’s Lectures on Quantum Mechanics and Quantum Computing
  https://www.youtube.com/watch?v=OkkAwRgcnWA&list=PL2fCZiDqOYYWR3UR9zkKrmW03igCZSL7P
- Timeline of Quantum Computing
  https://en.wikipedia.org/wiki/Timeline_of_quantum_computing
- Quantum Frontiers (CalTech)
  https://quantumfrontiers.com/2013/08/22/the-most-awesome-animation-about-quantum-computers-you-will-ever-see/
- Unboxing a Quantum Computer - DWave
  https://www.youtube.com/watch?v=60OkanvToFI
- Algorithms implemented by 1QBit
  https://1qbit.com/technology/
- IBM QuBit Simulator
  https://quantumexperience.ng.bluemix.net/qx/editor

Questions ?