

Grover's Search Algorithm

Introduction to quantum computing using QSim

Jothishwaran C A

Indian Institute of Technology Roorkee

email: jc_a@ece.iitr.ac.in

A particular kind of Boolean function

- Consider an n -variable Boolean function $F: \{0,1\}^n \rightarrow \{0,1\}$ that has the following definition:

$$F(x) = \begin{cases} 1 & ; x = x_0 \\ 0 & ; x \neq x_0 \end{cases}$$

$x_0 \in \{0,1\}^n$
—
n-bit string

- This function evaluates to 1 if and only if the input value is x_0 .
- The problem for this discussion is stated as follows:

"Given oracle access to such a Boolean function, how many times must the oracle of $F(x)$ be queried to find out the value of x_0 ?"

Oracle access : Only perform input / output queries.

Classical approach to the search problem

- Given that only oracle access to the function is available, the only option available to us is to check if the function evaluates to 1 at each point.
- This means one would have to query the function a maximum of 2^n times in order to deterministically find out the value of x_0 .
- The search problem has a classical computational complexity of $O(N)$, where $N = 2^n$.

↓
linear complexity

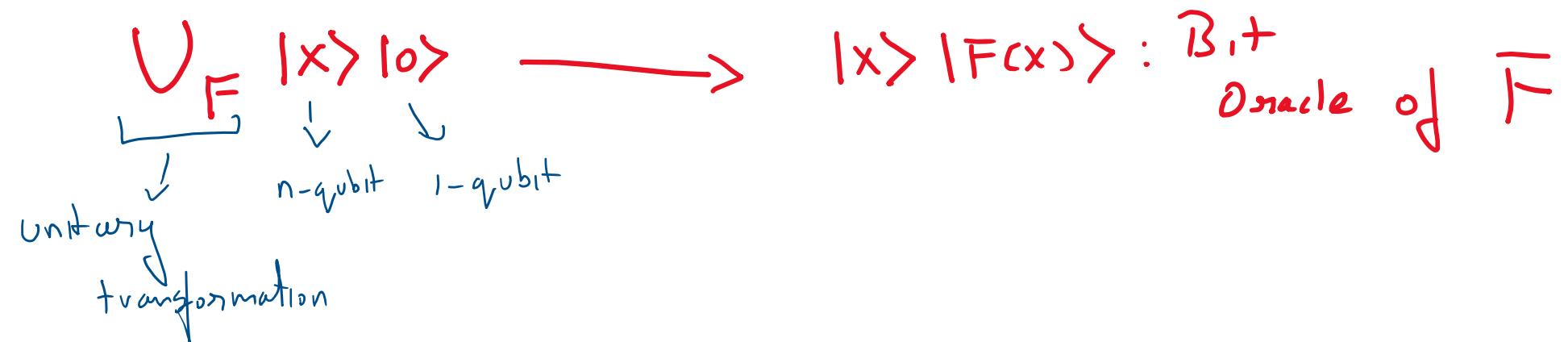
Can quantum computing do any better?

Grover's Search Algorithm

What do we have on a Quantum Computer?

→ Search is on a Boolean function

Therefore,



→ Quantum Parallelism : $U_F (|x_1\rangle + |x_2\rangle)|0\rangle \longrightarrow$

It is possible to evaluate F at multiple points using a single query.

$$|x_1\rangle|F(x_1)\rangle + |x_2\rangle|F(x_2)\rangle$$

Grover's Search Algorithm

→ Hadamard gate (H) : $\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$

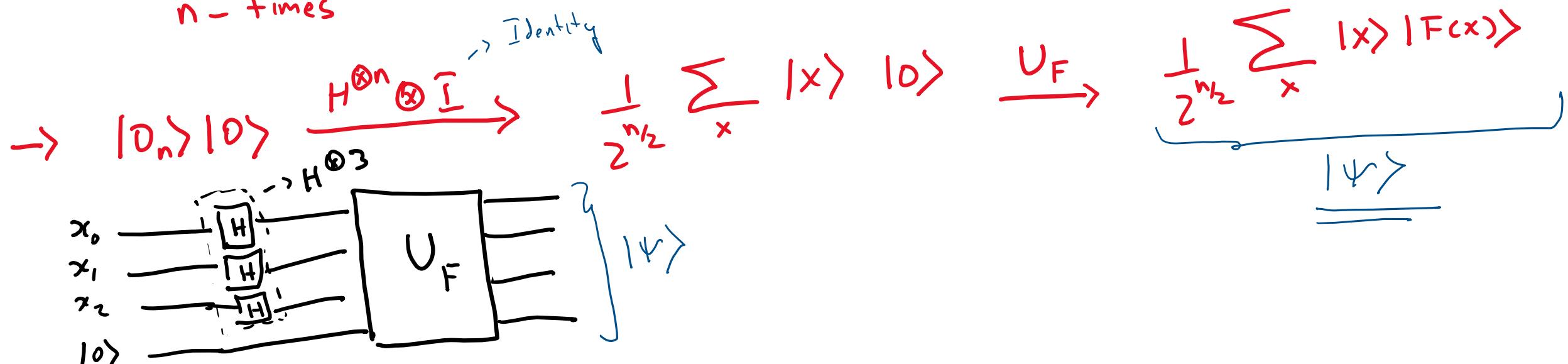
$$H|0\rangle \rightarrow |+\rangle$$

$$H|1\rangle \rightarrow |- \rangle$$

$$|0_n\rangle \xrightarrow{H^{\otimes n}} \frac{1}{\sqrt{2^{n/2}}} \sum_{x \in \{0,1\}^n} |x\rangle$$

: Equal weight superposition
of all possible values
of x

$$\underbrace{|0\rangle \otimes |0\rangle \otimes \dots \otimes |0\rangle}_{n\text{-times}} \Rightarrow |0_n\rangle \equiv |0\rangle^{\otimes n}$$



$|F\rangle$ contains the complete information about F

Since we know about F

$$|F\rangle = \frac{1}{2^{n/2}} \left[\sum_{x \neq x_0} |x\rangle |0\rangle + |x_0\rangle |1\rangle \right] \xrightarrow{\text{Measure}}$$

If the last is measured to be in $|1\rangle$ then, the first n qubits will measure to give the value of x_0 .

What is the probability of Success?

$$\text{Prob}\{\text{success}\} = \left| \frac{1}{2^{n/2}} \right|^2 = \frac{1}{2^n} \rightarrow \text{This is worse than the classical case.}$$

→ It would be preferable if the coefficient of $|x\rangle |i\rangle$ be increased somehow.

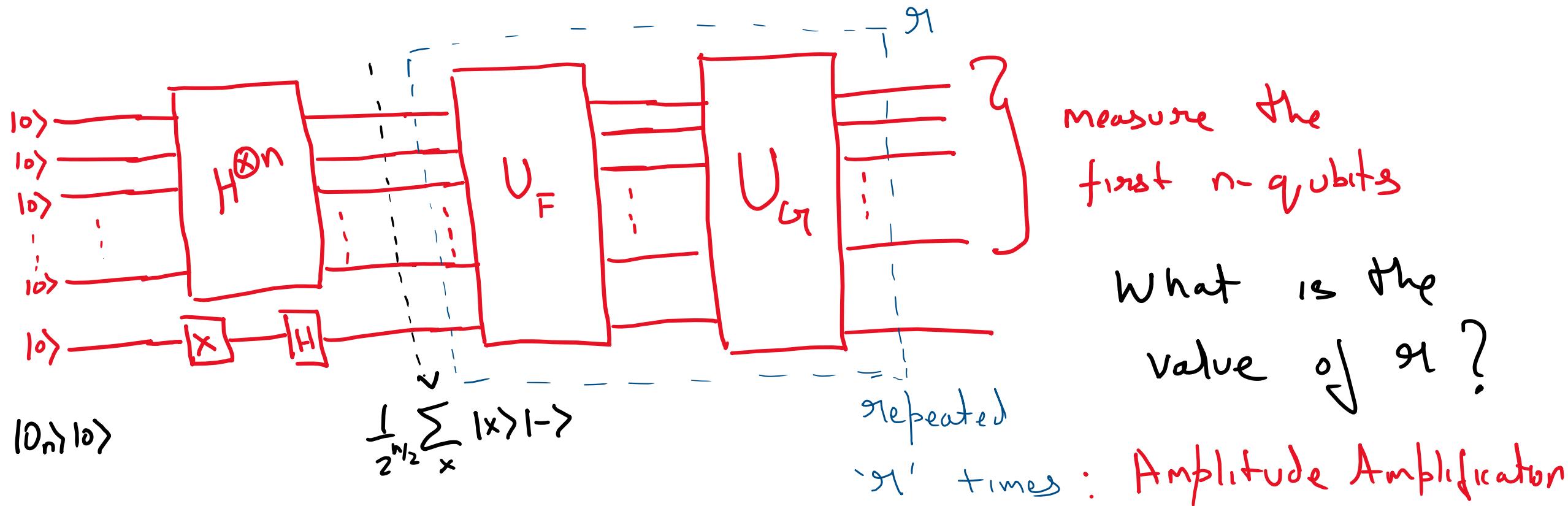
This is made possible using Grover's Rotation : U_G
{Grover Gate}

$$U_G = 2| \phi \rangle \langle \phi | - I_n : - n\text{-qubit Identity matrix } (2^n \times 2^n)$$

U_G - $2^n \times 2^n$ matrix : n-qubit transformation

$$|\phi\rangle = \frac{1}{2^{n/2}} \sum_{x \in \{0,1\}^n} |x\rangle$$

Outline of Grover's Algorithm [n+1 qubit version]



The measurement on the first n qubits will give the result of X_0 with a high probability

Grover's Algorithm: a brief analysis

- i) What is the value of α_1 ?
- ii) Why did we choose U_{CR} ?

$$H^{\otimes n} |0_n\rangle \longrightarrow |\phi\rangle \quad (\text{as previously defined})$$

$$\langle \phi | = (H^{\otimes n} |0_n\rangle)^+$$

$$= \langle 0_n | (H^+)^{\otimes n} \quad \because H^+ = H$$

$$= \langle 0_n | H^{\otimes n}$$

$$\begin{aligned} (|1\rangle\rangle)^+ &= \langle\psi| - \\ (|1\rangle\rangle \otimes |1\rangle\rangle)^+ &= (|1\rangle\rangle)^+ \otimes (|1\rangle\rangle)^+ \\ &= \langle\psi_1| \langle\psi_2| \end{aligned}$$

$$\begin{aligned} (A \otimes B)^+ &= A^+ \otimes B^+ \end{aligned}$$

$$\begin{aligned} (U|\psi\rangle)^+ &= \langle\psi| U^+ \end{aligned}$$

Above are some useful results

$$H^+ = H \quad ; \quad H \text{ is unitary} \quad \therefore H^+ H = I$$

final property of tensor products :

$$(A \otimes B) \cdot (C \otimes D) \quad ; \quad \bullet - \text{matrix multiplication}$$
$$= (A \cdot C) \otimes (B \cdot D)$$

$$(H^{(\otimes n)})^+ \cdot (H^{(\otimes n)}) = I_n$$

Using these we can redefine $U_{\alpha\gamma}$

Alternate version of U_{eff}

$$U_{\text{eff}} = 2|\phi\rangle\langle\phi| - I_n$$

$$= 2 \left[H^{\otimes n} |0_n\rangle\langle 0_n| H^{\otimes n} \right] - H^{\otimes n} \cdot I_n \cdot H^{\otimes n}$$

$$= H^{\otimes n} \cdot \left[2 |0_n\rangle\langle 0_n| - I_n \right] \cdot H^{\otimes n}$$

$$|0_n\rangle = \begin{pmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \Rightarrow |0_n\rangle \langle 0_n| = \begin{pmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & & \\ \vdots & & & & & \\ 0 & & & & & \end{pmatrix}_{2^n \times 2^n}$$

$$\begin{aligned} I_n &= \begin{pmatrix} 1 & & 0 \\ & \ddots & \\ 0 & \ddots & 1 \end{pmatrix}_{2^n \times 2^n} & \therefore 2|0_n\rangle \langle 0_n| - I_n \\ &= \begin{pmatrix} 1 & & 0 \\ -1 & -1 & \\ 0 & \ddots & -1 \end{pmatrix}_{2^n \times 2^n} & = M \end{aligned}$$

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

$|x\rangle$ - n-qubit state
 $x \in \{0, 1\}^n$

$$M|x\rangle = (-1)^{G(x)} |x\rangle$$

$G(x)$ is an n-variable Boolean function

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

$G(x)$ is the n-variable OR gate

$\therefore M$ is the phase oracle of the n-bit OR gate

The $n+1$ qubit version of M

The bit oracle representation of the OR gate $U_{OR} \rightarrow n+1$ qubit gate

$$U_{OR}|x\rangle|-\rangle = (-1)^{OR(x)}|x\rangle|-\rangle$$

Therefore the $n+1$ qubit version of $U_{OR} = (H^{\otimes n} \otimes I) \cdot U_{OR} \cdot (H^{\otimes n} \otimes I)$

here, it is understood the the last qubit is in $|-\rangle$ state

What is the value of α ?

When performing Grover's search over an ' n ' variable Boolean function

$$\alpha = \left[\frac{\pi}{4} \times \sqrt{2^n} \right] = \left[\frac{\pi}{4} \times 2^{n/2} \right] \quad [\text{presented without proof}]$$

$[m]$ → nearest integer to 'm'

$$\therefore \text{for } n=4 ; \alpha = \left[\frac{\pi}{4} \times \sqrt{2^4} \right] = [1] \\ = 3$$

A 4 variable Grover's Search

$$F(x) = x_3 \wedge x_2 \wedge \bar{x}_1 \wedge x_0$$

will evaluate to 1 only
at 1101

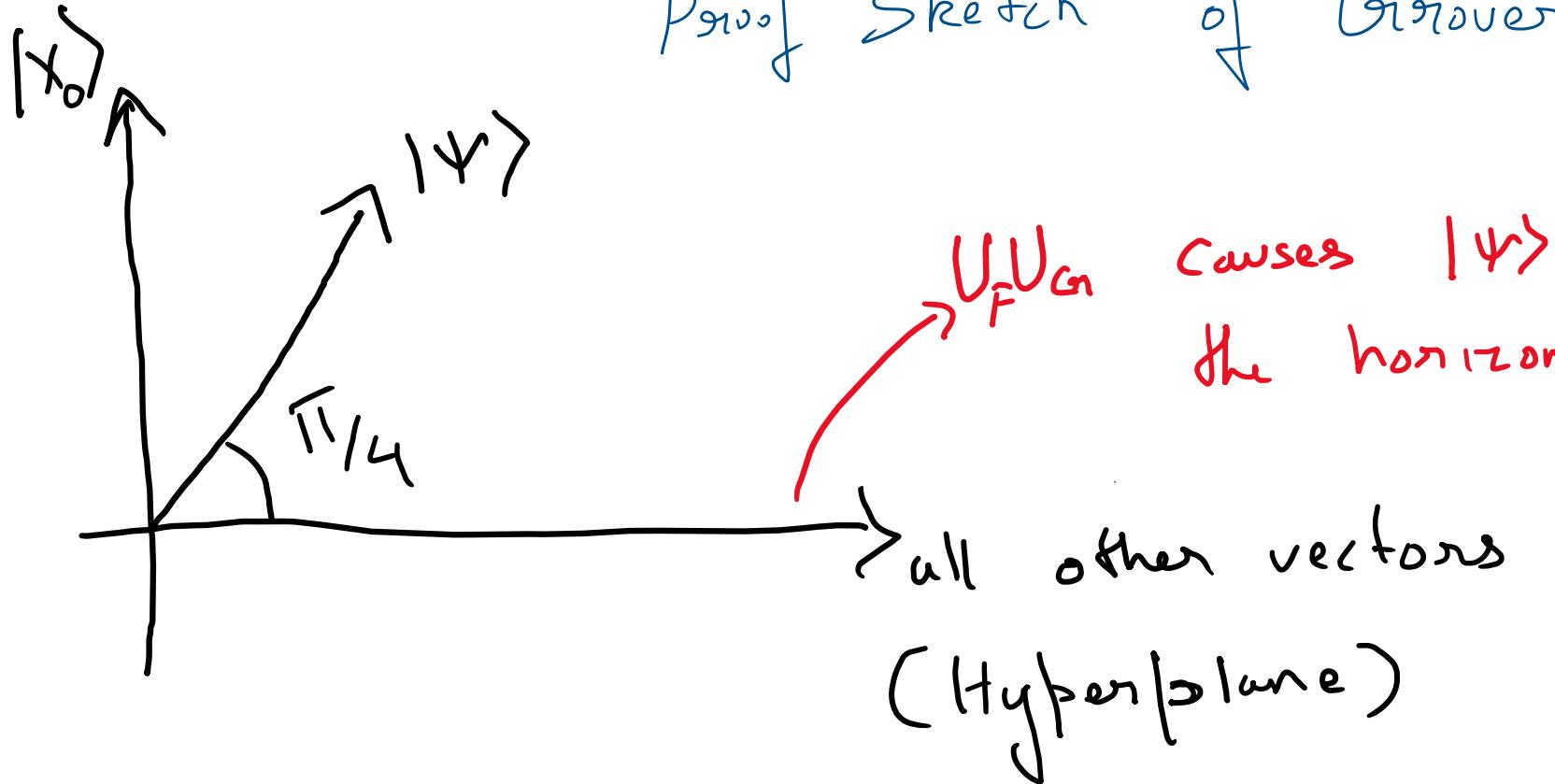
$$\bar{x}_1 = 1 + x_1$$

; + - XOR operation

$$F(x) = x_3 x_2 x_1 x_0 + x_3 x_2 x_0 \rightarrow \text{ANF of } F(x)$$

$$OR(x) = x_3 \vee x_2 \vee x_1 \vee x_0 = [\bar{x}_3 \bar{x}_2 \bar{x}_1 \bar{x}_0]'$$

Proof Sketch of Grover iterations:



Causes $|\psi\rangle$ to rotate about the horizontal line

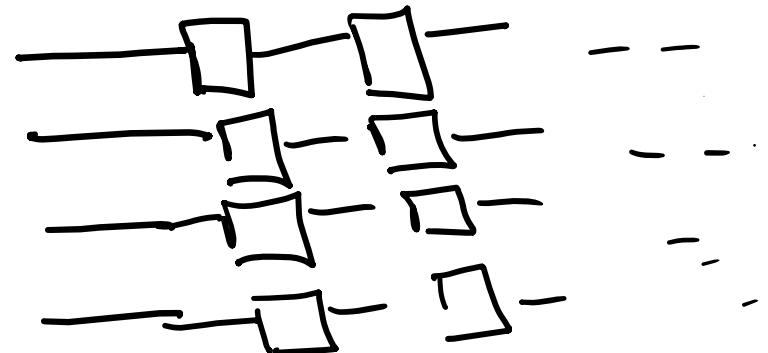
doing "n" rotations
will take angle closest to
 $\pi/2$

$$\pi = \left[\frac{\pi}{4} 2^{n/2} \right]$$

A brief note on Quantum Volume (QV)

QV is the size of the largest square circuit that can be run with low error on a Quantum computer.

QV - 16



largest known QV : 2^4

What is the required Quantum volume Grover's Algorithm

$$\text{required Q.V.} = O(n^2 2^{n/2}) ; \quad n=4 ; \quad 16 \times 4 = 64$$

$$n=16 ; \quad 16 \times 16 \times 2^8 = 2^{16} ; \quad N \approx 65,000$$