Plusieurs qubits

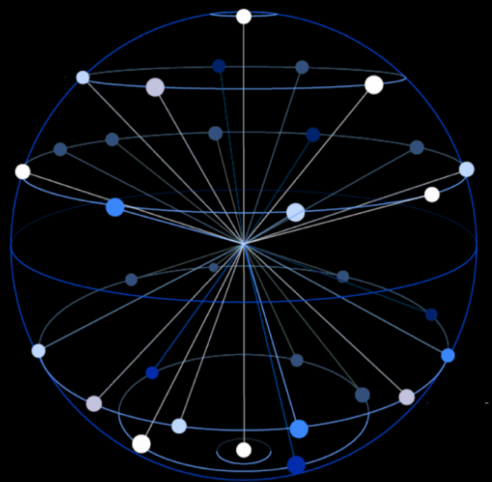
IBM Client Center Montpellier

JM Torres | torresjm@fr.ibm.com | 2-4 décembre 2019

© 2019 IBM Corporation

Multiple qubits

Universal quantum computers leverage quantum mechanical properties of superposition and entanglement to create states that scale exponentially with number of qubits, or quantum bits.

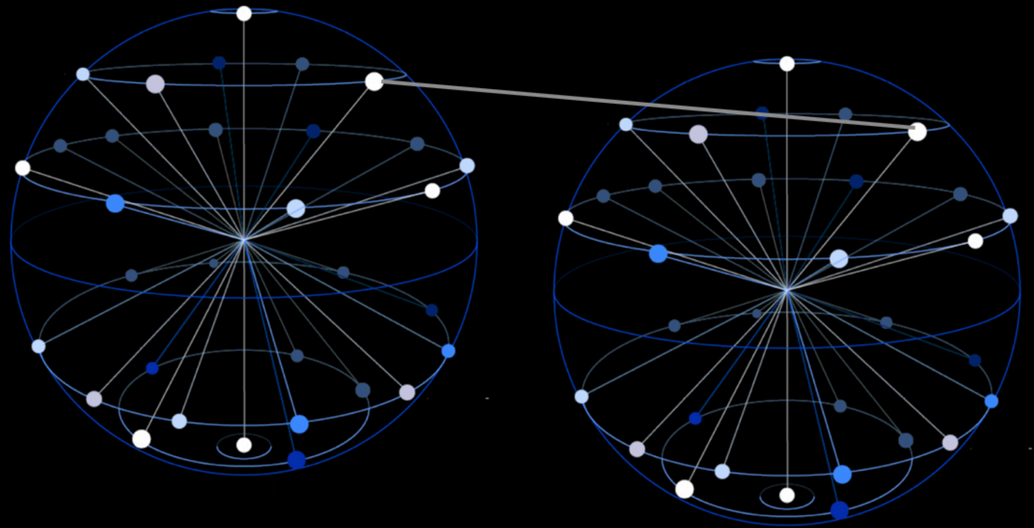


Superposition

A single quantum bit can exist in a superposition of 0 and 1:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

and N qubits allow for a superposition of all possible 2^N combinations.



Entanglement

The states of entangled qubits cannot be described independently of each other.

© 2019 IBM Corporation

Système de deux qubits :

le qubit A est dans son espace d'états $|\boldsymbol{\psi}_A\rangle = \boldsymbol{\alpha}|0_A\rangle + \boldsymbol{\beta}|1_A\rangle$
le qubit B est dans son espace d'états $|\boldsymbol{\psi}_B\rangle = \boldsymbol{\gamma}|0_B\rangle + \boldsymbol{\delta}|1_B\rangle$

Le système des deux qubits, A et B, est décrit par le produit des deux états, il suffit de multiplier :

$$|\boldsymbol{\Psi}_{AB}\rangle = |\boldsymbol{\psi}_A\rangle\ |\ \boldsymbol{\psi}_B\rangle = (\boldsymbol{\alpha}|0_A\rangle + \boldsymbol{\beta}|1_A\rangle) \times (\boldsymbol{\gamma}|0_B\rangle + \boldsymbol{\delta}|1_B\rangle)$$

$$|\boldsymbol{\Psi}\rangle = \boldsymbol{\alpha\gamma}|0_A\rangle|0_B\rangle + \boldsymbol{\alpha\delta}|0_A\rangle|1_B\rangle + \boldsymbol{\beta\gamma}|1_A\rangle\ |0_B\rangle + \boldsymbol{\beta\delta}|1_A\rangle|1_B\rangle$$

© 2019 IBM Corporation

Système de deux qubits :

$$|\boldsymbol{\Psi}\rangle = \boldsymbol{\alpha\gamma}|0_A\rangle\,|0_B\rangle + \boldsymbol{\alpha\delta}|0_A\rangle\,|1_B\rangle + \boldsymbol{\beta\gamma}|1_A\rangle\,|0_B\rangle + \boldsymbol{\beta\delta}|1_A\rangle|1_B\rangle$$

En faisant attention à « l'ordre » dans le ket, on peut simplifier la notation sans ambiguïté:

$$|\boldsymbol{\Psi}\rangle = \boldsymbol{\alpha\gamma}|0_A0_B\rangle + \boldsymbol{\alpha\delta}|0_A1_B\rangle + \boldsymbol{\beta\gamma}|1_A0_B\rangle + \boldsymbol{\beta\delta}|1_A1_B\rangle$$

$$|\boldsymbol{\Psi}\rangle = \boldsymbol{\alpha\gamma}|00\rangle + \boldsymbol{\alpha\delta}|01\rangle + \boldsymbol{\beta\gamma}|10\rangle + \boldsymbol{\beta\delta}|11\rangle$$

© 2019 IBM Corporation

Système de deux qubits :

$$|\boldsymbol{\Psi}\rangle = \boldsymbol{\alpha\gamma}|00\rangle + \boldsymbol{\alpha\delta}|01\rangle + \boldsymbol{\beta\gamma}|10\rangle + \boldsymbol{\beta\delta}|11\rangle$$

Ensemble, les deux qubits sont donc décrits dans un espace de dimension quatre dont les vecteurs de base sont : $|00\rangle$, $|01\rangle$, $|10\rangle$, $|11\rangle$

L'état général s'écrit :

$|\psi\rangle = a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle$ et $|a|^2+|b|^2+|c|^2+|d|^2 = 1$

Et pour 3 qubits :

$|\psi\rangle = a|000\rangle + b|001\rangle + c|010\rangle + d|011\rangle + e|100\rangle + f|101\rangle + g|110\rangle + h|111\rangle$

(et la somme des carrés des coefficients vaut 1)

Ainsi de suite pour 4 qubits…, n qubits

© 2019 IBM Corporation

Two qubit states calculations (tensor product)

$$\begin{pmatrix}\alpha_1\\ \beta_1\end{pmatrix} \otimes \begin{pmatrix}\alpha_2\\ \beta_2\end{pmatrix} = \begin{pmatrix}\alpha_1\alpha_2\\ \alpha_1\beta_2\\ \beta_1\alpha_2\\ \beta_1\beta_2\end{pmatrix}$$

$$|0> \otimes |0> = |00> = \begin{pmatrix}1\\ 0\end{pmatrix} \otimes \begin{pmatrix}1\\ 0\end{pmatrix} = \begin{pmatrix}1\\ 0\\ 0\\ 0\end{pmatrix}$$

$$|0> \otimes |1> = |01> = \begin{pmatrix}1\\ 0\end{pmatrix} \otimes \begin{pmatrix}0\\ 1\end{pmatrix} = \begin{pmatrix}0\\ 1\\ 0\\ 0\end{pmatrix}$$

© 2019 IBM Corporation

Two qubit states calculations (tensor product)

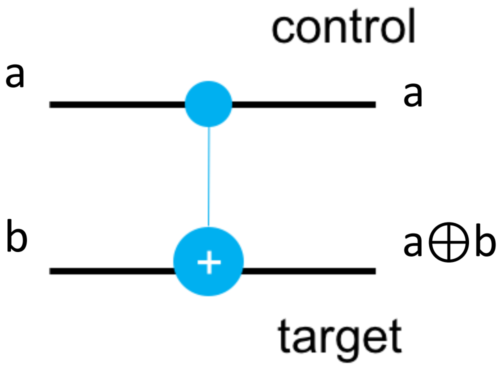
$$|1> \otimes |0> = |10> = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

$$|1> \otimes |1> = |11> = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

And so on : $\begin{pmatrix} \alpha_1 \\ \beta_1 \end{pmatrix} \otimes \begin{pmatrix} \alpha_2 \\ \beta_2 \end{pmatrix} \otimes \begin{pmatrix} \alpha_3 \\ \beta_3 \end{pmatrix} = \begin{pmatrix} \alpha_1\alpha_2\alpha_3 \\ \alpha_1\alpha_2\beta_3 \\ \alpha_1\beta_2\alpha_3 \\ \alpha_1\beta_2\beta_3 \\ \beta_1\alpha_2\alpha_3 \\ \beta_1\alpha_2\beta_3 \\ \beta_1\beta_2\alpha_3 \\ \beta_1\beta_2\beta_3 \end{pmatrix}$, par exemple : $|101> = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$

© 2019 IBM Corporation

Multi–qubits Gates : CNOT gate



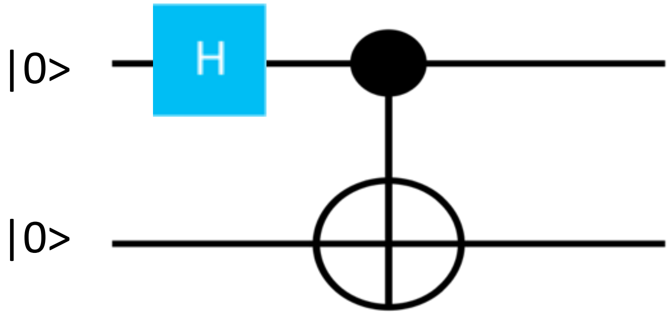
- CNOT = Controlled-NOT Gate
- Inverts a target qubit according to the state of the control qubit

CNOT		
a	b	a⊕b
0	0	0
0	1	1
1	0	1
1	1	0

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

© 2019 IBM Corporation

Example Bell-State



$$|00\rangle = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\frac{1}{\sqrt{2}}(|00\rangle + |10\rangle) = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

$$\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

© 2018 IBM Corporation

intrication

$$|Bell\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

Si $|Bell\rangle$ est le produit de deux états quantiques :

$$|\psi\rangle = a\,|0\rangle + (b + ic)|1\rangle \quad et \quad |\varphi\rangle = d|0\rangle + (e + if)|1\rangle$$

Alors :

$$|\psi\rangle|\varphi\rangle = ad\,|00\rangle + ae + iaf|01\rangle + (db + idc\,)|10\rangle + (b + ic)(e + if)|11\rangle$$

on peut identifier les coefficients sur les vecteurs de base :

$$(1)\ ad = \frac{1}{\sqrt{2}}$$

$$(2)\ \ ae + iaf\ = 0$$

$$(3)\ \ db + idc = 0$$

$$(4)\ \ be\ - cf\ + i(bf + ce) = \frac{1}{\sqrt{2}}$$

© 2019 IBM Corporation

intrication

$$(1)\ ad = \frac{1}{\sqrt{2}}$$

$$(2)\ ae + iaf = 0$$

$$(3)\ db + idc = 0$$

$$(4)\ be - cf + i(bf + ce) = \frac{1}{\sqrt{2}}$$

D’après (1) : AD n’est pas nul donc a et d sont non nuls, dans (2) et (3) on déduit $e = b = 0\ et\ f = c = 0$

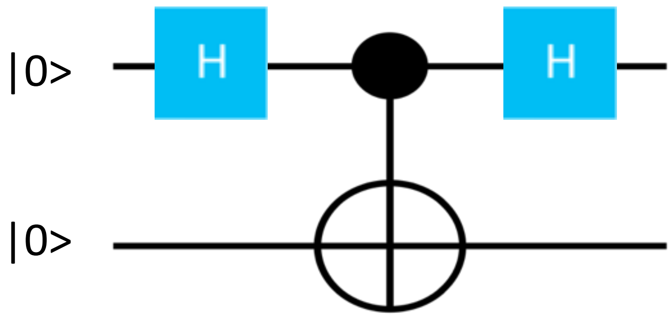
Donc be – $cf = 0$ ce qui contredit (4)

L’état de Bell ne peut pas être le produit de deux états quantiques.

Ce qui veut dire qu’on ne peut rien connaitre de chacun des deux composants du système intriqué, on ne peut considérer que le système comme un tout.

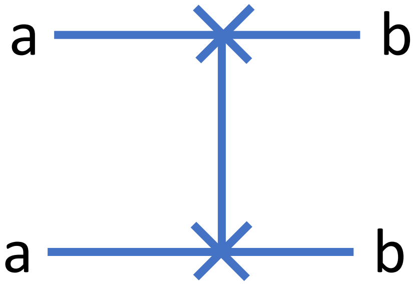
© 2019 IBM Corporation

allons un peu plus loin



© 2018 IBM Corporation

Multi–qubits Gates : SWAP gate



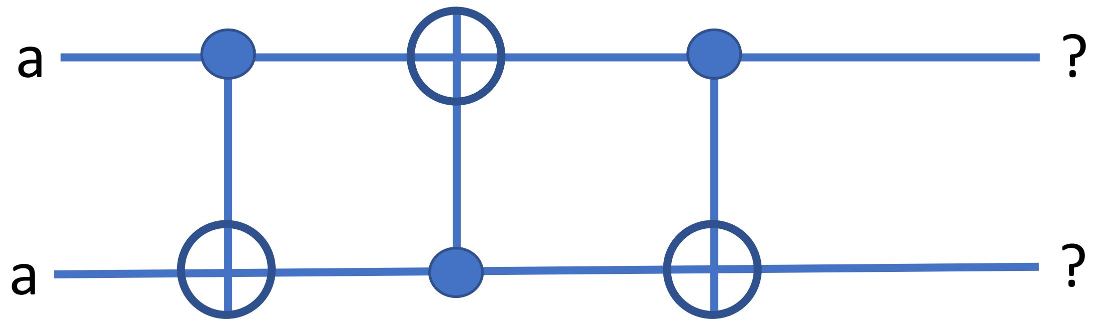
- Swappe les états des qubits d'entrée

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

SWAP			
a	b	a	b
0	0	0	0
0	1	1	0
1	0	0	1
1	1	1	1

© 2018 IBM Corporation

Que fait ce circuit ?



© 2018 IBM Corporation

SWAPPER n’est pas CLONER

Théorème de non clonage : on ne peut pas copier un état quantique (Wooters & Zurech, 1982)

La démonstration se fait par l’absurde, en supposant que nous disposons d’un opérateur U, tel que :

$$(a)\ U(|\Psi\rangle \otimes |\Phi\rangle) = |\Psi\rangle \otimes |\Psi\rangle \quad \text{pour tout état } \Psi$$

qui permettrait de copier $|\Psi\rangle$ sur $|\Phi\rangle$. Posons $|\Omega\rangle = \alpha|\Psi\rangle$ on réécrit (a) avec $|\Omega\rangle$:

$$U(|\Omega\rangle \otimes |\Phi\rangle) = |\Omega\rangle \otimes |\Omega\rangle$$

$$U(|\Omega\rangle \otimes |\Phi\rangle) = \alpha|\Psi\rangle \otimes \alpha|\Psi\rangle = \alpha^2|\Psi\rangle \otimes |\Psi\rangle$$

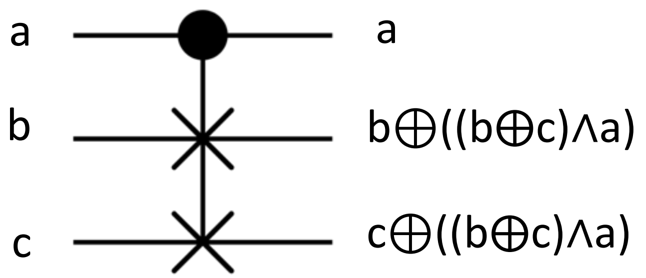
D’autre part : $U(|\Omega\rangle \otimes |\Phi\rangle) = U(\alpha|\Psi\rangle \otimes |\Phi\rangle)$

et par linéarité de U : $U(|\Omega\rangle \otimes |\Phi\rangle) = \alpha U(|\Psi\rangle \otimes |\Phi\rangle) = \alpha|\Psi\rangle \otimes |\Psi\rangle$

on trouve que l’on force $\alpha^2 = \alpha$, ce qui n’est pas vrai en général

© 2019 IBM Corporation

Multi–qubits Gates : CSWAP (Fredkin gate)



- CSWAP = Controlled-Swap Gate
- Swaps two target qubits according to the state of the control qubit

CSWAP					
a	b	c	(b⊕c)∧a	b⊕((b⊕c)∧a)	c⊕((b⊕c)∧a)
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	0	0	0
1	0	1	1	1	0
1	1	0	1	0	1
1	1	1	0	1	1

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

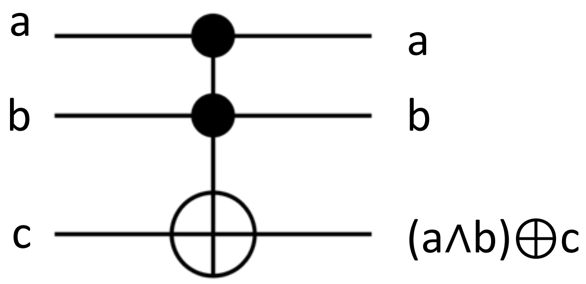
Basis: |000>,|001>,|010>,|011>,|100>,|101>,|110>,|111>

Note:

- If c==0 : c⊕((b⊕c)∧a) = a ∧ b
- If c==1: b⊕((b⊕c)∧a) = a ∨ b
- If b==0 & c ==1 : c⊕((b⊕c)∧a) = ¬a

© 2019 IBM Corporation

Multi–qubits Gates : CCNOT (Toffoli gate)



- CCNOT = Control-Control-NOT Gate
- Inverts a target qubit according to the state of the two control qubits

CCNOT			
a	b	c	(a∧b)⊕c
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Basis: |000>,|001>,|010>,|011>,|100>,|101>,|110>,|111>

© 2018 IBM Corporation

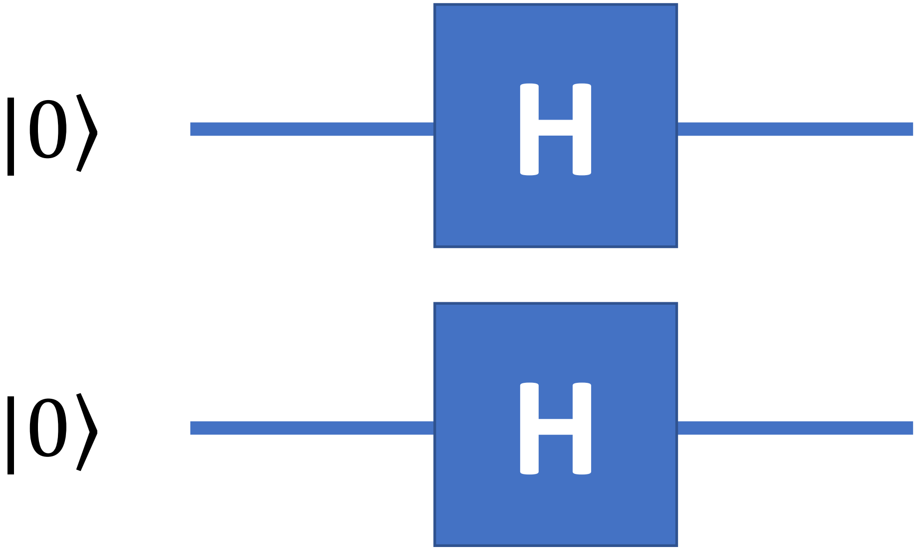
$H^{\otimes n}$

$|0\rangle$

$$\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

© 2018 IBM Corporation

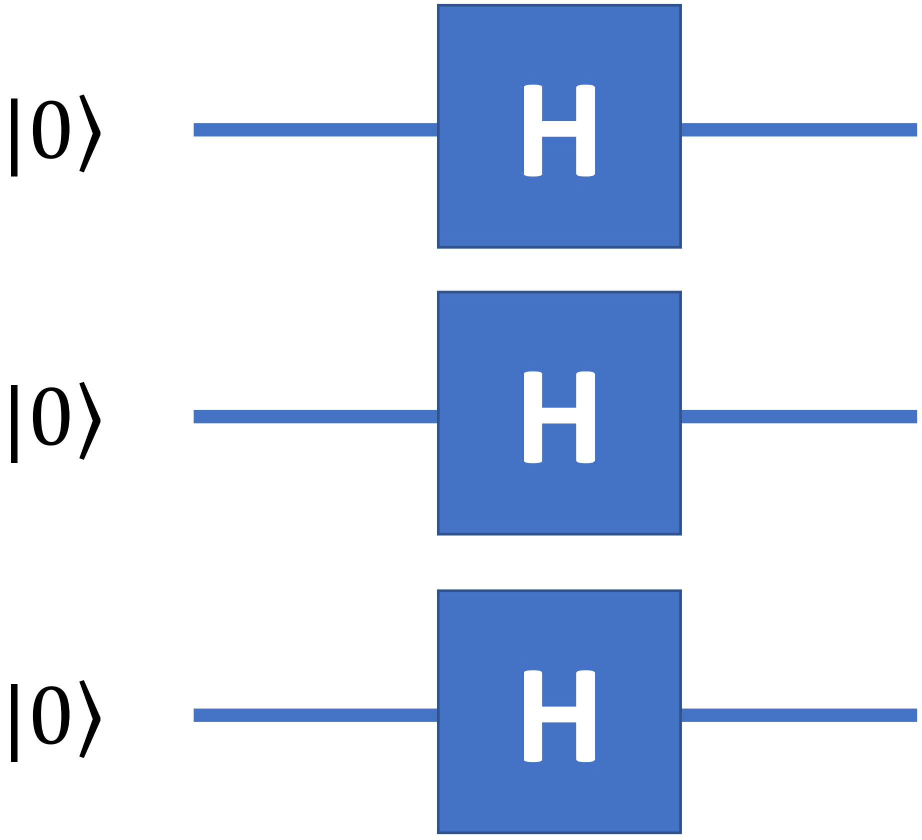
$H^{\otimes n}$



$$\frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)$$

© 2018 IBM Corporation

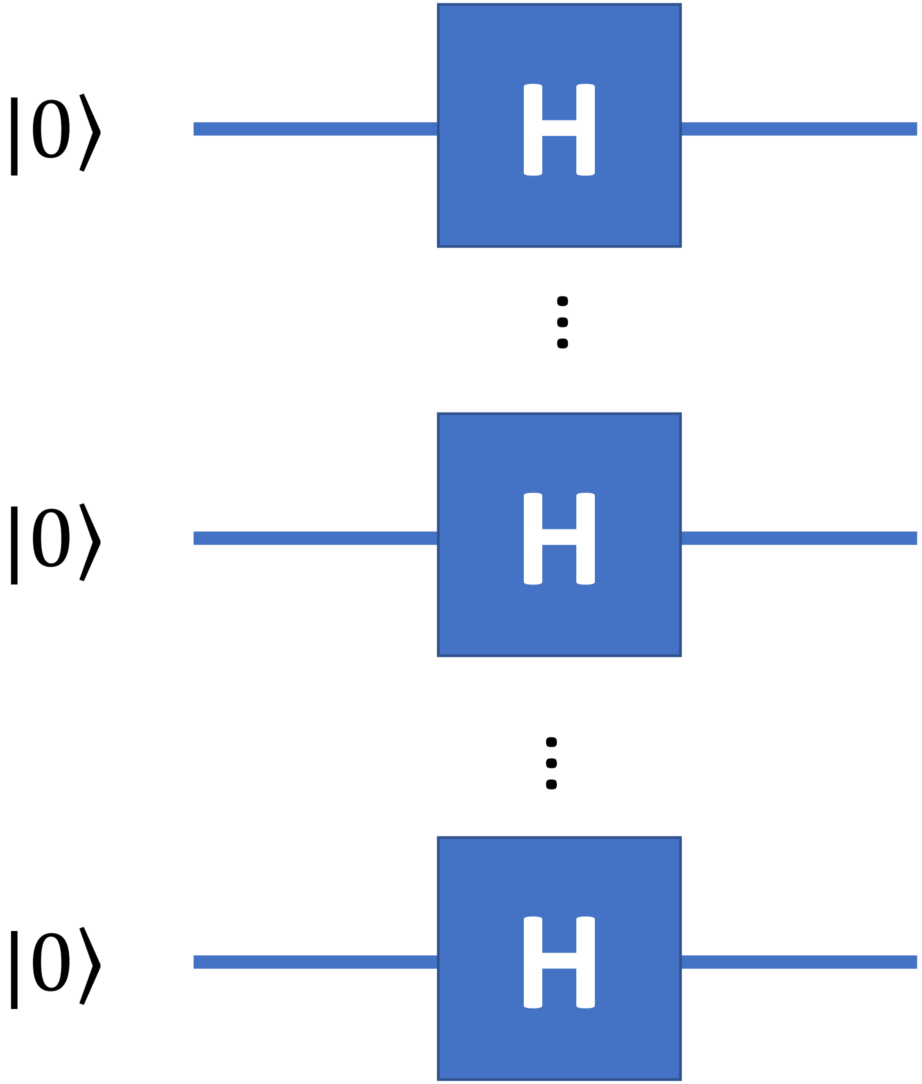
$H^{\otimes n}$



$$\frac{1}{2\sqrt{2}}(|000\rangle + |001\rangle + |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle)$$

© 2019 IBM Corporation

H^n

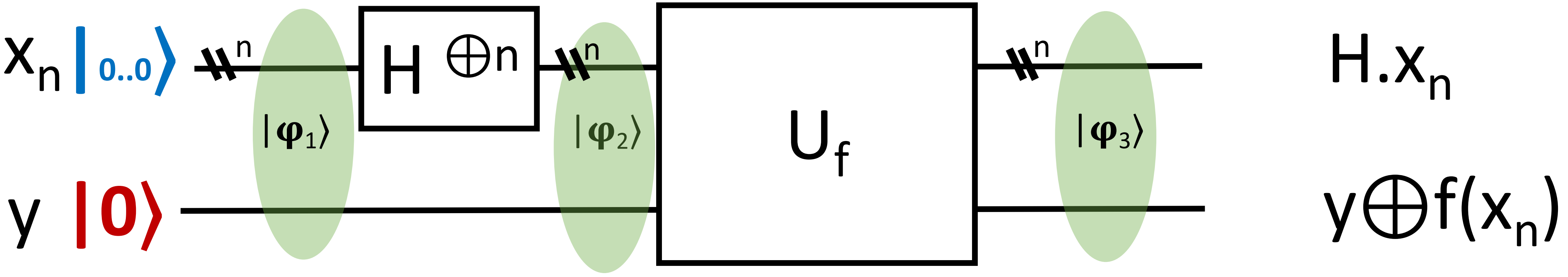


$$\frac{1}{\sqrt{2^n}}(|0\ldots0\rangle + |0\ldots1\rangle + \cdots$$

$$+|1\ldots10\rangle + |1\ldots11\rangle)$$

© 2018 IBM Corporation

“Parallelism”



$$|\boldsymbol{\varphi}_1\rangle = |0..0,0\rangle \qquad |\boldsymbol{\varphi}_2\rangle = \frac{1}{\sqrt[n]{2}}(|0..0,0\rangle + \ldots + |1\ldots1,0\rangle)$$

$$|\boldsymbol{\varphi}_3\rangle = \frac{1}{\sqrt[n]{2}}(|f(0..0),0)\rangle + \ldots + |f(1..1),0\rangle)$$

L’état de sortie $|\boldsymbol{\varphi}_3\rangle$ contient les résultats du calcul de f appliqué à toutes les valeurs possibles de x_n, c’est-à-dire f(0..0), f(0..1), et f(1..1), alors que U_f a été appliqué une seule fois.

Faire de cela quelque chose d’utile est l’enjeu de l’algorithmique quantique.

© 2019 IBM Corporation

FIN?

© 2018 IBM Corporation

Deustch si on a le temps, ce sera gagné pour le lendemain

© 2018 IBM Corporation

“Deutsch problem, Deutsch Algorithm”

Let f be a function from {0,1} to {0,1} : $f : x \in \{0,1\} \longmapsto \{0,1\}$

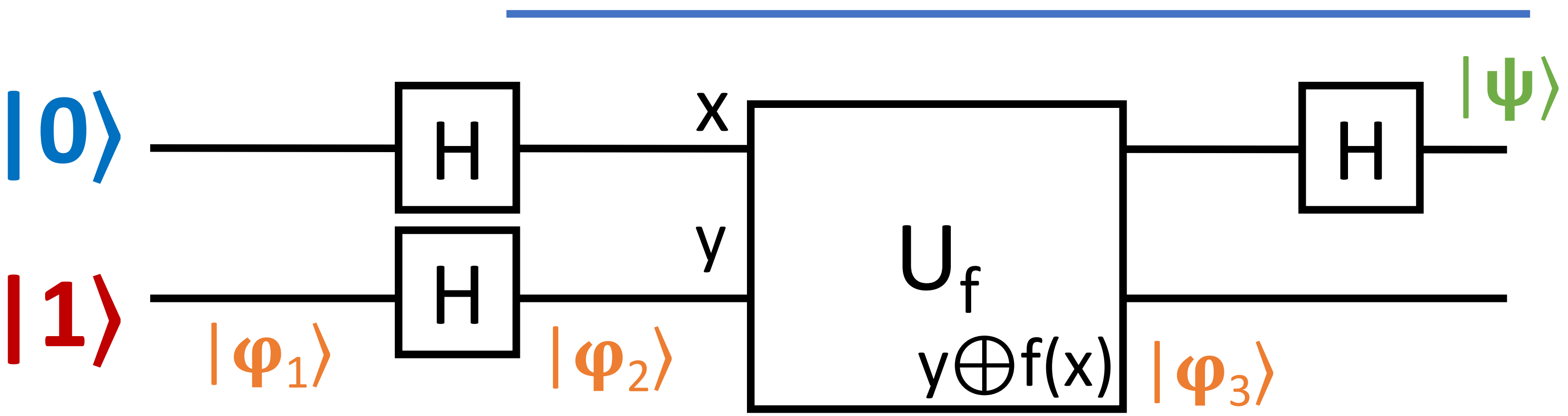
x	$f_0(x)$	$f_1(x)$	$f_2(x)$	$f_3(x)$
0	0	0	1	1
1	0	1	0	1

f_0 and f_3 are **constant** functions ($\forall x : f_3(x) = 1$)
f_1 and f_2 are **balanced** ($Card(f_1^{-1}(0)= Card(f_1^{-1}(1)$)

If we want to determine if such a function is constant or balanced, we need to calculate $f(0)$ and $f(1)$ … which requires two evaluations of f .
David Deutsch has demonstrated, in 1985 that a quantum algorithm can determine if f is constant or balanced in just one pass.

© 2019 IBM Corporation

Deutsch algorithm



$|\varphi_1\rangle = |0\rangle \otimes |1\rangle$

$|\varphi_2\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$

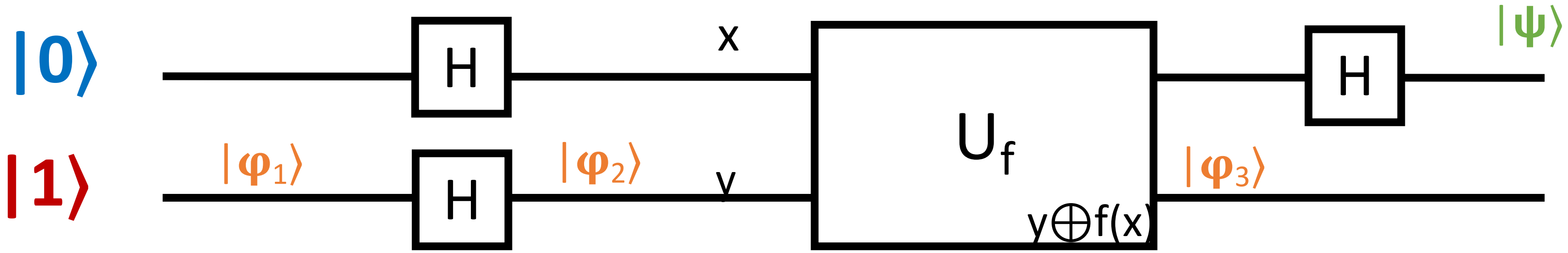
$|\varphi_2\rangle = \frac{1}{2}(\ |0\rangle(|0\rangle - |1\rangle) + |1\rangle(|0\rangle - |1\rangle)\ )$ (develop)

$|\varphi_2\rangle = \frac{1}{2}\sum_{x=0}^{x=1} |x\rangle\ (|0\rangle - |1\rangle)$ (factor $(|0\rangle - |1\rangle)$)

$|\varphi_2\rangle = \frac{1}{2}\sum_{x=0}^{x=1} (|x\rangle|0\rangle - |x\rangle|1\rangle)$ … now let's see what $|\varphi_3\rangle$ looks like

© 2019 IBM Corporation

Deutsch algorithm



$|\varphi_2\rangle = \frac{1}{2}\sum_{x=0}^{x=1} (|x\rangle|0\rangle - |x\rangle|1\rangle)$

$|\varphi_3\rangle = \frac{1}{2}\sum_{x=0}^{x=1} (|x\rangle|0\oplus f(x)\rangle - |x\rangle|1\oplus f(x)\rangle)$

Nota :

if f(x) == 0 then $0\oplus f(x) = 0$ and $1\oplus f(x) = 1$

if f(x) == 1 then $0\oplus f(x) = 1$ and $1\oplus f(x) = 0$

So:

if f(x) == 0 then $|x\rangle\ |0\oplus f(x)\rangle - |x\rangle|1\oplus f(x)\rangle = |x\rangle\ |0\rangle - |x\rangle|1\rangle$

if f(x) == 1 then $|x\rangle\ |0\oplus f(x)\rangle - |x\rangle|1\oplus f(x)\rangle = |x\rangle\ |1\rangle - |x\rangle|0\rangle$

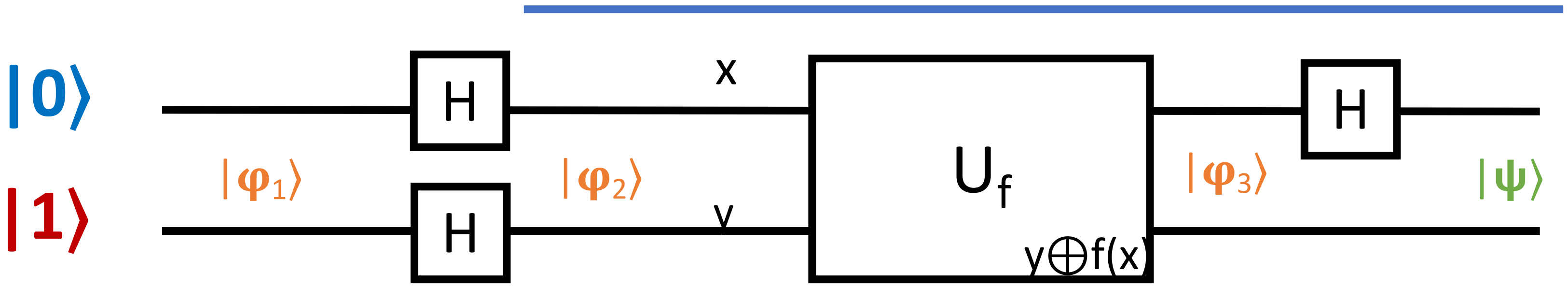
Then:

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

$|\varphi_3\rangle = \frac{1}{2}\sum_{x=0}^{x=1}[(-1)^{f(x)}](|x\rangle\ |0\rangle - |x\rangle|1\rangle) = \frac{1}{2}\sum_{x=0}^{x=1}[(-1)^{f(x)}]|x\rangle\ (|0\rangle - |1\rangle)$ and we are done

© 2019 IBM Corporation

Deutsch algorithm



Let’s have our first qubit go through H:

$|\varphi_3\rangle = \frac{1}{2}\sum_{x=0}^{x=1}\left[(-1)^{f(x)}\right]|x\rangle\,(|0\rangle - |1\rangle) = \frac{1}{2}\left[(-1)^{f(0)}|0\rangle + (-1)^{f(1)}|1\rangle\right](|0\rangle - |1\rangle)$

$|\psi\rangle = = \frac{1}{2}\left[(-1)^{f(0)}\left(\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\right) + (-1)^{f(1)}\left(\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)\right)\right](|0\rangle - |1\rangle)$

$|\psi\rangle = = \frac{1}{2\sqrt{2}}\left[\left((-1)^{f(0)} + (-1)^{f(1)}\right)|0\rangle + \left((-1)^{f(0)} - (-1)^{f(1)}\right)|1\rangle\right](|0\rangle - |1\rangle)$

If *f(0) == f(1)* a measurement on the first qubit will result in $|0\rangle$ because $\left((-1)^{f(0)} - (-1)^{f(1)}\right) = 0$

If *f(0) ≠ f(1)* a measurement on the first qubit will result in $|1\rangle$, because $\left((-1)^{f(0)} + (-1)^{f(1)}\right) = 0$

In 1992, Deutsch & Josza came with the same algorithm… for n qubits x_n

© 2018 IBM Corporation