Algorithme de Bernstein-Vazirani

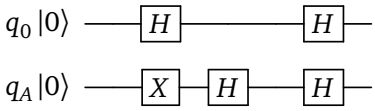
Il est maintenant possible de mettre en application ce que nous avons vu jusqu'ici pour étudier les algorithmes "historiques". Il s'agit de démonstrations de ce qu'il est possible de réaliser avec les qubits et leurs propriétés particulières et en quoi de nouvelles solutions peuvent être ainsi apportées. En l'occurrence l'utilité immédiate n'est pas le premier critère recherché. Ces algorithmes émergent dans les vingt dernières années du XX^{ieme} siècle, les ordinateurs quantiques n'existent pas encore, il s'agit de trouver des exemples pour montrer les possibilités théoriques du calcul quantique.

1. Phase Kickback - Retour de phase

L'algorithme de Bernstein-Vazirani (découvert en 1997) exploite un aspect de l'intrication quantique, obtenu par une porte $CNOT$, voyons cela en détail.
restes

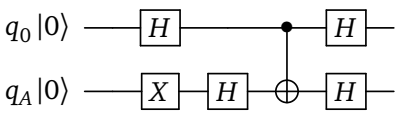
Voici deux configurations de circuits à 2 qubits, identiques à une porte C_X près :

1. le qubit 0 n'interfère pas avec le qubit A :



   Dans ce cas, l'état de q_0 en sortie du circuit est bien évidemment $|0\rangle$, car $H^2 = I$

2. l'état du qubit 0 contrôle le qubit A :



La raison pour laquelle le second qubit est noté q_A est que ce qubit est utilisé pour mener à bien les transformations sur q_0 mais son état final ne nous intéresse pas, il s'agit d'un qubit « auxilliaire ».

Dans cette configuration le qubit q_0 est toujours mesuré à l'état $|1\rangle$

L'état du qubit q_0 vaut $|0\rangle$ en sortie si il n'y a pas de porte CNOT), et $|1\rangle$ si il y a une porte CNOT.

Dans ce second cas, on pourrait s'attendre à ce que l'état de q_0 soit à l'état $|0\rangle$ en sortie du circuit, mais regardons ce qu'il se passe à chaque étape : initialement l'état du système des deux qubits vaut $|00\rangle$.
Avant la porte $CNOT$, l'état du système est $H \otimes HX\,|00\rangle$ (H est appliqué à q_0 d'une part, X puis H est appliqué à q_A d'autre part, on combine les deux opérateurs par leur produit tensoriel $\otimes$ pour obtenir l'opérateur sur les deux qubits, que l'on applique à l'état initial $|00\rangle$)

$$HX = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix} \times \begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}$$

$$HX = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ -1 & 1\end{pmatrix}$$

$$H \otimes HX = \frac{1}{2}\begin{pmatrix}1 & 1 & 1 & 1\\ -1 & 1 & -1 & 1\\ 1 & 1 & -1 & -1\\ -1 & 1 & 1 & -1\end{pmatrix}$$

et donc :

$$H \otimes HX\,|00\rangle = \frac{1}{2}\begin{pmatrix}1 & 1 & 1 & 1\\ -1 & 1 & -1 & 1\\ 1 & 1 & -1 & -1\\ -1 & 1 & 1 & -1\end{pmatrix} \times \begin{pmatrix}1\\ 0\\ 0\\ 0\end{pmatrix} = \frac{1}{2}\begin{pmatrix}1\\ -1\\ 1\\ -1\end{pmatrix}$$

On pouvait aussi dire que l'on a l'état $|+\rangle \otimes |-\rangle$ avant d'appliquer la porte $CNOT$ soit :

$$|+\rangle \otimes |-\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = \frac{1}{2}(|00\rangle - |01\rangle + |10\rangle - |11\rangle)$$

Ce qui représente bien le même vecteur d'état.

On applique à présent $CNOT$ à cet état :

$$CNOT\begin{pmatrix}1\\ -1\\ 1\\ -1\end{pmatrix} = \begin{pmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 1 & 0\end{pmatrix} \times \begin{pmatrix}1\\ -1\\ 1\\ -1\end{pmatrix} = \begin{pmatrix}1\\ -1\\ -1\\ 1\end{pmatrix}$$

On pouvait aussi utiliser l'expression $\frac{1}{2}(|00\rangle - |01\rangle + |10\rangle - |11\rangle)$, et voir comment chacune des quatre composantes de ce vecteur est transformé par la porte $CNOT$.
Selon la règle de la porte $CNOT$:

- la composante sur $|00\rangle$ est inchangée,
- la composante sur $|01\rangle$ est inchangée,
- la composante sur $|10\rangle$ est projetée sur $|11\rangle$,
- la composante sur $|11\rangle$ est projetée sur $|10\rangle$,

On trouve donc : $\frac{1}{2}(|00\rangle - |01\rangle + |11\rangle - |10\rangle)$, une fois que les vecteurs de base sont remis dans l'ordre : $\frac{1}{2}(|00\rangle - |01\rangle - |10\rangle + |11\rangle)$
Il reste à passer le qubit q_0 par la porte H, c'est à dire qu'on applique $H \otimes I$ au vecteur précédent représentant l'état des deux qubits :

$$(H \otimes I)\frac{1}{2}\begin{pmatrix}1\\ -1\\ -1\\ 1\end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 0 & 1 & 0\\ 0 & 1 & 0 & 1\\ 1 & 0 & -1 & 0\\ 0 & 1 & 0 & -1\end{pmatrix} \times \frac{1}{2}\begin{pmatrix}1\\ -1\\ -1\\ 1\end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix}0\\ 0\\ 1\\ -1\end{pmatrix} = \frac{1}{\sqrt{2}}(|10\rangle - |11\rangle)$$

On voit que la mesure sur le qubit 0 donnera 1.

Dans le premier cas, de manière évidente, le qubit 0 sort à l'état $|0\rangle$. Donc le fait de contrôler ou pas, un qubit à l'état $|-\rangle$ renvoie le qubit 0 à l'état $|1\rangle$ ou $|0\rangle$. C'est ce qu'on appelle le « retour de phase », ou « phase kickback », et c'est ce phénomène qui est exploité dans l'algorithme de Bernstein-Vazirani.

Nota : après le « phase kickback », le qubit auxiliaire, qui était dans l'état $|-\rangle$ avant d'être contrôlé par un qubit à l'état $|+\rangle$ se retrouve dans l'état $-|-\rangle$, regardons alors l'effet de cet état sur le prochain qubit qui viendrait à faire une $CNOT$ sur ce même qubit auxiliaire.

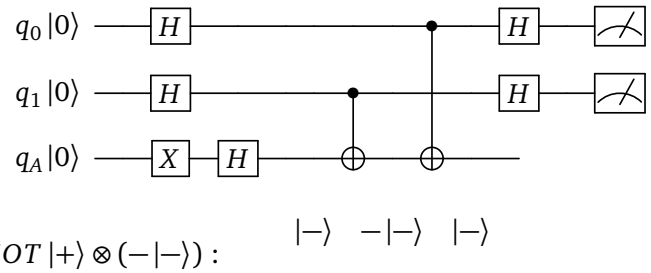
D'abord, notons que :

$$X|-\rangle = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \times \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} -1 \\ 1 \end{pmatrix} = -|-\rangle$$

et

$$X(-|-\rangle) = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \times \frac{1}{\sqrt{2}} \begin{pmatrix} -1 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix} = |-\rangle$$

Ensuite vérifions que ce phénomène persiste pour plusieurs utilisations de la porte CNOT depuis un qubit vers q_A. C'est à dire que le qubit auxiliaire est à l'état $-|-\rangle$, il s'agit de voir l'état du qubit q_A sur ce schéma :



Il s'agit ce calculer $CNOT|+\rangle \otimes (-|-\rangle)$:
on a d'abord :

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

$$|+\rangle \otimes (-|-\rangle) = \frac{1}{2}(-|00\rangle + |01\rangle) - |10\rangle + |11\rangle) = \frac{1}{2} \begin{pmatrix} -1 \\ 1 \\ -1 \\ 1 \end{pmatrix}$$

On applique CNOT :

$$CNOT \frac{1}{2} \begin{pmatrix} -1 \\ 1 \\ -1 \\ 1 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} -1 \\ 1 \\ -1 \\ 1 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} -1 \\ 1 \\ 1 \\ -1 \end{pmatrix}$$

Puis, on obtient l'état final en appliquant $H \otimes I$:

$$H \otimes I \frac{1}{2} \begin{pmatrix} -1 \\ 1 \\ 1 \\ -1 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{pmatrix} \times \frac{1}{2} \begin{pmatrix} 0 \\ 0 \\ -2 \\ 2 \end{pmatrix} = \frac{1}{\sqrt{2}}(|11\rangle - |10\rangle$$

A nouveau, la probabilité de mesurer $|1\rangle$ sur le qubit sortant (ici on s'intéresse à q_0) est 1.

2. Algorithme de Bernstein-Vazirani

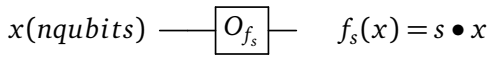
Voici à présent la description de ce que l'on appelle l'algorithme de Bernstein-Vazirani.
Soit f_s une fonction de $\{0,1\}^n$ (l'ensemble des bitstrings de longueur n) vers $\{0,1\}$, dont on sait qu'elle retourne le *ou exclusif* (ou l'addition modulo 2) des produits (and) bit à bit entre l'entrée x et un bit-string inconnu s (de longueur n) :

$$f_s(x) = s_0.x_0 \oplus s_1.x_1 \oplus s_2.x_2 \oplus \quad ... \quad \oplus s_{n-1}.x_{n-1}$$

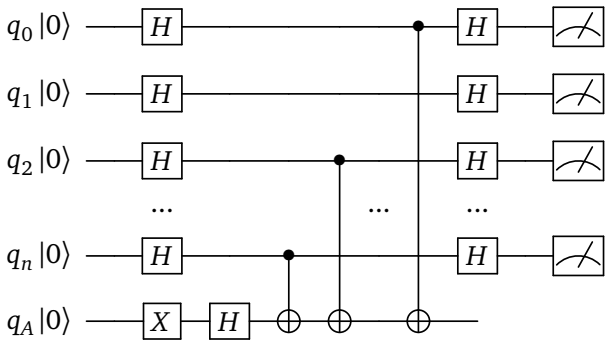
Cette fonction, considérée comme « une boite noire », cachant le secret s est appelée « Oracle » (au sens où l'on présente une entrée x et l'on reçoit un résultat : 0 ou 1 dans notre cas).
Le but est de découvrir les caractéristiques de l'oracle (la valeur de s) en utilisant le plus petit nombre d'appels possibles à l'oracle.
Dans le cadre du calcul classique, il faut n évaluations de $f_s(x)$ pour identifier s par exemple en procédant ainsi :

$$\begin{gathered} f_s(1000...0) = s_0 \\ f_s(0100...0) = s_1 \\ f_s(0010...0) = s_2 \\ ... \\ f_s(0000...1) = s_{n-1} \end{gathered}$$

L'algorithme de Bernstein-Vazirani y parvient en une seule fois (en utilisant $O(n)$ portes)



On utilise ce que l'on a vu ci-dessus pour construire l'oracle correspondant, en plaçant une porte $CNOT$ pour le qubit q_i si s_i vaut 1 (sinon pas de porte).



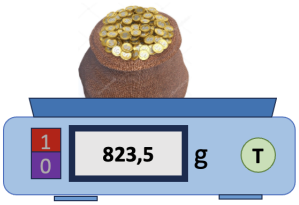
Ce circuit permet d'obtenir en une seule fois, dans le registre de sortie, la valeur de s. On aura alors un avantage d'ordre n par rapport à l'algorithme classique.
Aussi étrange qu'il puisse paraître (de résoudre un problème pour lequel on code la solution dans l'oracle), il faut prendre ce résultat au pied de la lettre : dans les conditions de la fonction f_s telles que décrites plus haut, et avec les règles indiquées : il faut au minimum n interrogations à l'oracle pour trouver s (n étant le nombre de bits de s), et l'algorithme de Bernstein-Vazirani ne nécessite qu'une seule interrogation de l'oracle.
L'utilité n'est pas un critère ici. Cet exemple illustre bien le fait que l'on cherche des manières radicalement nouvelles d'utiliser les qubits pour construire les bases d'un "calcul". D'autres exemples suivent.

Identifier les fausses pièces à l'aide d'une balance quantique

Une manière de décrire l'algorithme BV proposée par Simon Perdrix, Inria,CNRS, Université de Lorraine.

1. La pesée des pièces

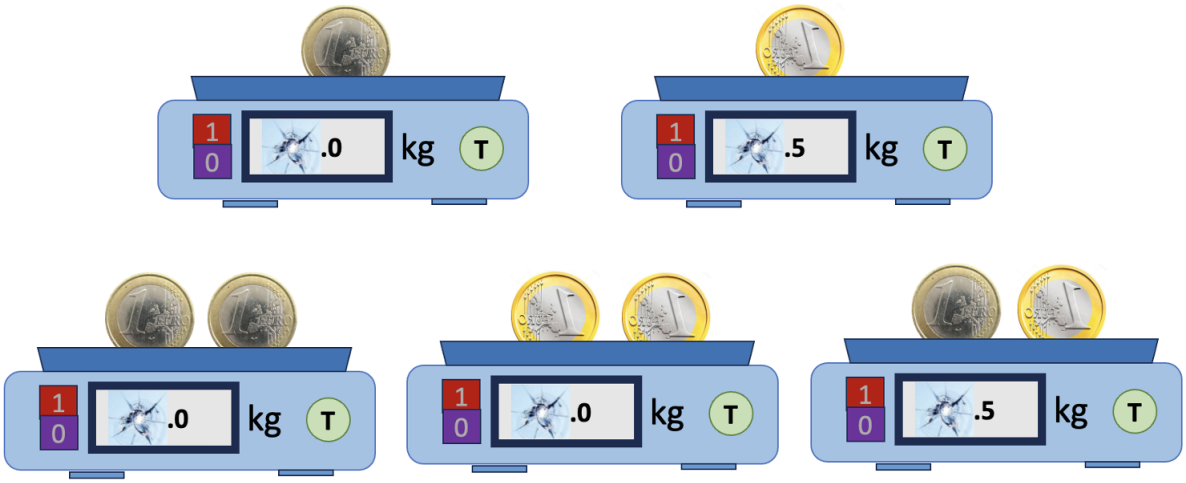
On suppose que l'on a un sac de pièces et on souhaite savoir si il y en a qui sont fausses : combien et lesquelles. Elles sont identiques sauf que les pièces authentiques pèsent 8g alors que les fausses pièces pèsent 7,5g.



Malheureusement l'affichage de la balance est défectueux et il n'affiche que les décimales :

Convention visuelle :

Avec une ou deux pièces, on obtient les informations suivantes :



Avec n pièces :

- si il y a une quantité paire de fausses pièces la balance affiche : ■.0
- si il y a une quantité impaire de fausses pièces la balance affiche : ■.5

1.1. Modélisation

Représentons l'ensemble des pièces sous la forme d'un mot binaire : les fausses pièces sont représentées par un 1, les vraies par un 0.

Soit a le mot binaire qui représente le sac de pièces étudié : une opération de pesée applique la fonction f_a : $x \rightarrow f_a(x)$ ($x \in \{0,1\}^n$), cette fonction renvoie :

$$f_a(x) = \sum_n x_i.a_i(mod2) = x \bullet a$$

On cherche la valeur de a (les a_i indiquent si la pièce i est fausse).

Dans le cas classique : la meilleure stratégie est de peser les pièces une par une, car on a besoin de connaître la valeur de n bits et chaque pesée fournit un seul bit d'information.

La balance possède une tare que l'on peut mettre à la valeur .0 ou .5, ce qui produit :

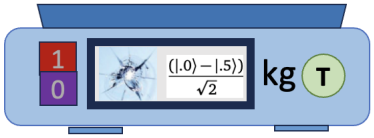
Affichage obtenu	si la tare vaut .0	si la tare vaut .5
Nombre pair de fausses pièces	.0	.5
Nombre impair de fausses pièces	.5	.0

C'est à dire : si le nombre de fausses pièces est impair, l'affichage change, si le nombre de fausses pièces est pair l'affichage ne change pas lors de la pesée.

2. avec une balance quantique

- La fonction f_a devient un unitaire (U_{f_a}) qui s'applique à des états $|x\rangle$ sur n qubits, x est un mot binaire qui représente un sac de pièce.
- Il est possible de mettre la valeur de la tare dans un état superposé des 2 valeurs possibles et on choisit :

$$|T\rangle = \frac{(|.0\rangle - |.5\rangle)}{\sqrt{2}}$$



- si la parité de x est 0 (nombre pair de fausses pièces), l'état du système {sac de pièces, affichage de la balance} avant la pesée est :

$$|x\rangle \otimes \frac{(|.0\rangle - |.5\rangle)}{\sqrt{2}}$$

Après la pesée, l'état devient :

$$|x\rangle \otimes \frac{(|.0\rangle - |.5\rangle)}{\sqrt{2}}$$

Car l'affichage ne change pas.

- Mais si le nombre de fausses pièces est impair, l'état avant la pesée est toujours :

$$|x\rangle \otimes \frac{(|.0\rangle - |.5\rangle)}{\sqrt{2}}$$

Ensuite :

— la composante : $\frac{1}{\sqrt{2}}|x\rangle \otimes |.0\rangle$ devient : $\frac{1}{\sqrt{2}}|x\rangle \otimes |.5\rangle$
— et la composante : $-\frac{1}{\sqrt{2}}|x\rangle \otimes |.5\rangle$ devient : $-\frac{1}{\sqrt{2}}|x\rangle \otimes |.0\rangle$

et au total l'état devient :

$$\frac{1}{\sqrt{2}}|x\rangle \otimes |.5\rangle - \frac{1}{\sqrt{2}}|x\rangle \otimes |.0\rangle = |x\rangle \frac{1}{\sqrt{2}}(|.5\rangle - |.0\rangle) = -|x\rangle \frac{1}{\sqrt{2}}(|.0\rangle - |.5\rangle)$$

— En résumé :

— Cas pair :

$$|x\rangle \otimes \frac{(|.0\rangle - |.5\rangle)}{\sqrt{2}} \rightarrow |x\rangle \otimes \frac{(|.0\rangle - |.5\rangle)}{\sqrt{2}}$$

— Cas impair :

$$|x\rangle \otimes \frac{(|.0\rangle - |.5\rangle)}{\sqrt{2}} \rightarrow -|x\rangle \otimes \frac{(|.0\rangle - |.5\rangle)}{\sqrt{2}}$$

C'est à dire que la pesée change le signe de $|x\rangle$ selon la parité du mot binaire x.
Donc :

$$|x\rangle \overset{\text{pesée}}{\rightarrow} U_{f_a}|x\rangle$$

$$|x\rangle \overset{\text{pesée}}{\rightarrow} (-1)^{f_a(x)}|x\rangle$$

$$|x\rangle \overset{\text{pesée}}{\rightarrow} (-1)^{x \bullet a} |x\rangle$$

3. Procédure Bernstein-Vazirani

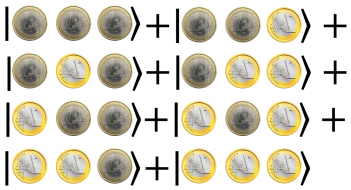
On rappelle les trois formules suivantes :

$$H^{\otimes n} |0\rangle^{\otimes n} = |s\rangle = \frac{1}{\sqrt{2^n}} \sum_x |x\rangle$$

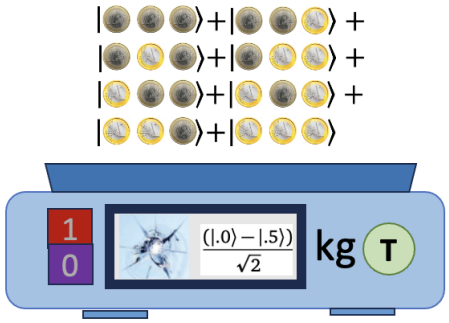
$$H^{\otimes n} |x\rangle = \frac{1}{\sqrt{2^n}} \sum_j (-1)^{x \bullet j} |j\rangle$$

$$H^{\otimes n} . H^{\otimes n} = I^{\otimes n} = I_n$$

Pour commencer on part de l'état $|0\rangle^{\otimes n}$ et on produit la superposition de tous les sacs de pièces possibles (ici avec 3 pièces) : $H^{\otimes n} |0\rangle^{\otimes n} = |s\rangle = \frac{1}{\sqrt{2^n}} \sum_x |x\rangle$



On pose ceci sur la balance :



Il reste à réaliser la pesée :

$$\frac{1}{\sqrt{2^n}} \sum_x |x\rangle \overset{\text{pesée}}{\mapsto} \frac{1}{\sqrt{2^n}} \sum_x U_{f_a} |x\rangle$$

$$\frac{1}{\sqrt{2^n}} \sum_x |x\rangle \overset{\text{pesée}}{\mapsto} \frac{1}{\sqrt{2^n}} \sum_x (-1)^{x \bullet a} |x\rangle$$

(et on a vu que $\frac{1}{\sqrt{2^n}} \sum_j (-1)^{j \bullet a} |j\rangle = H^{\otimes n} |a\rangle$)

et donc :

$$\frac{1}{\sqrt{2^n}} \sum_x |x\rangle \overset{\text{pesée}}{\mapsto} H^{\otimes n} |a\rangle$$

On applique $H^{\otimes n}$ à ce résultat de pesée, ce qui fait $H^{\otimes n} H^{\otimes n} |a\rangle = |a\rangle$ La mesure produira a, en résumé : on génère l'état $|s\rangle$, on effectue la pesée, on applique des portes H à tous les qubits du registre, on mesure pour trouver a.

■ c.q.f.d.

■ jm. Torrès (torresjm@fr.ibm.com)