

Algorithme de Bernstein-Vazirani

1 Problème à résoudre

Soient x et s tels que $x, s \in \{0, 1\}^n$.

On pose une fonction f définie par :

$$\begin{aligned} f : x &\rightarrow y = s \cdot x \pmod{2} = x_1 s_1 + x_2 s_2 + \cdots + x_n s_n \\ f : \{0, 1\}^n &\rightarrow \{0, 1\}, \end{aligned}$$

Exemple 1. Soit s le mot booléen suivant : $s = 10$. La fonction f a donc la table de vérité suivante :

(x_1, x_2)	s	$f(x_1, x_2)$
$(0, 0)$	10	0
$(0, 1)$	10	0
$(1, 0)$	10	1
$(1, 1)$	10	1

On observe que le résultat est de 1 pour les entrées (x_1, x_2) où l'emplacement des 1 correspond à ceux de s .

Problème 1 (Bernstein-Vazirani). Etant donné un mot s secret, et la fonction f implémentant l'opération décrite précédemment, comment peut-on retrouver s en le moins d'évaluations de f possibles ?

1.1 Solution classique

Dans le cas classique, on va devoir évaluer au pire toutes les valeurs possibles de s pour trouver sa valeur, soit n évaluations de f . C'est un algorithme de complexité $\mathcal{O}(n)$

1.2 Solution quantique

Dans le cas quantique, ce problème se résout en une seule évaluation quantique de f . L'algorithme reprends celui de Deutsch-Jozsa en changeant la fonction appliquée dans l'oracle quantique.

1.2.1 Initialisation

On commence avec : $|u_0\rangle = (|0\rangle^{\otimes n})$: n-qubits à $|0\rangle$

1.2.2 Etape 1

On applique une porte de Hadamard à $|u_0\rangle$ pour avoir un état équiprobable : $|u_1\rangle = H|u_0\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle$

1.2.3 Etape 2

On applique l'oracle quantique suivant à $|u_1\rangle$:

$$o : |x\rangle|y\rangle \mapsto |x\rangle|y \oplus (s \cdot x \pmod{2})\rangle.$$

En suivant exactement le même raisonnement que pour Deutsch-Jozsa, on arrive à l'expression suivante :

$$|u_2\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} (-1)^{s \cdot x \pmod{2}} |x\rangle \quad (1)$$

1.2.4 Etape 3

De la même façon à Deutsch-Jozsa, on applique une porte Hadamard à chaque qubit sortant, ce qui donne :

$$|u_3\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} (-1)^{s \cdot x \pmod{2}} \left(\frac{1}{\sqrt{2^n}} \sum_{y=0}^{2^n-1} (-1)^{x \cdot y} |y\rangle \right)$$

$$|u_3\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} (-1)^{s \cdot x \pmod{2}} \left(\frac{1}{\sqrt{2^n}} \sum_{y=0}^{2^n-1} (-1)^{x \cdot y} |y\rangle \right) \quad (2)$$

$$= \frac{1}{2^n} \sum_{x=0}^{2^n-1} \sum_{y=0}^{2^n-1} (-1)^{(s \cdot x \pmod{2}) + x \cdot y} |y\rangle \quad (3)$$

Et on peut prouver que $\frac{1}{2^n} \sum_{x=0}^{2^n-1} \sum_{y=0}^{2^n-1} (-1)^{(s \cdot x \pmod{2}) + x \cdot y} |y\rangle$ est égal à $|s\rangle$ (à faire ...)

1.3 Exemple

Prenons par exemple $s = (10)_2 = 2_{10}$, soit $f(x) = 2 \cdot x \pmod{2}$

Etape 1 : porte de Hadamard

On commence avec $|u_0\rangle = |00\rangle$. La première étape est l'application de la porte d'hadamard à $|u_0\rangle$:

$$|u_1\rangle = H|u_0\rangle = H|0\rangle \otimes H|0\rangle \quad (4)$$

$$= \frac{1}{2} ((|0\rangle + |1\rangle) \otimes (|0\rangle + |1\rangle)) \quad (5)$$

$$= \frac{1}{2} \{|00\rangle + |01\rangle + |10\rangle + |11\rangle\} \quad (6)$$

Etape 2 : oracle quantique

On applique à $|u_1\rangle$ l'oracle quantique $|x\rangle|y\rangle \rightarrow |x\rangle|y \oplus (s \cdot x \pmod{2})\rangle =:$

$$\begin{aligned} |u_2\rangle &= \frac{1}{2} ((-1)^{10 \cdot 00 \pmod{2}} |00\rangle + (-1)^{10 \cdot 01 \pmod{2}} |01\rangle + (-1)^{10 \cdot 10 \pmod{2}} |10\rangle + (-1)^{10 \cdot 11 \pmod{2}} |11\rangle) \\ &= \frac{1}{2} ((-1)^0 |00\rangle + (-1)^0 |01\rangle + (-1)^1 |10\rangle + (-1)^1 |11\rangle) \\ &= \frac{1}{2} (|00\rangle + |01\rangle - |10\rangle - |11\rangle) \end{aligned}$$

Etape 3 : porte de Hadamard

On applique donc une porte de hadamard à $|u_2\rangle$:

$$|u_3\rangle = \frac{1}{2} H (|00\rangle + |01\rangle - |10\rangle - |11\rangle) \quad (7)$$

Nous sommes sur une porte de hadamard pour 2 qubits, ce qui donne la relation matricielle suivante pour $|u_3\rangle$:

$$|u_3\rangle = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ -1 \\ -1 \end{bmatrix}, \quad (8)$$

$$= \frac{1}{4} \begin{bmatrix} 0 \\ 0 \\ 4 \\ 0 \end{bmatrix}. \quad (9)$$

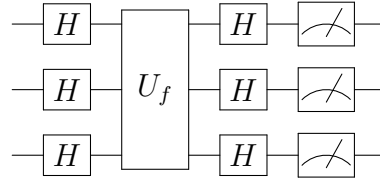
Lors de la mesure, on va obtenir l'état $|10\rangle$ avec une probabilité de 1, qui était bien notre mot binaire s de départ.

On peut observer que, lors de l'application de la porte de Hadamard à $|u_2\rangle$, on obtient la superposition d'état suivante : $|00\rangle + |01\rangle - |10\rangle - |11\rangle$. Cela correspond à la troisième ligne de la matrice de Hadamard, correspondant au $|s\rangle$ voulu. Dans tout les cas, peu importe le s choisi, on obtiendra une superposition d'état correspondant à une des lignes de la matrice, forçant à 0 les probabilités de tout les états sauf de celui indiqué.

1.4 Implémentation du circuit

Circuit global

L'implémentation du circuit quantique pour cet algorithme est très similaire à celui de Deutsch-Jozsa, à la différence qu'on a un qubit de moins :



Implémentation de l'oracle

Prenons le cas où $n = 2$. La matrice correspondant à la porte U_f va avoir 4 possibilité pour obtenir, comme on l'a dit lors de l'exemple, une des 4 lignes de la matrice de Hadamard :

$$U_{f_{00}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, U_{f_{01}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}, U_{f_{10}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}, U_{f_{11}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

On remarque que ces quatres matrices sont en fait des produits tensoriels de deux matrices correspondant à des portes à 1 qubit :

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

Pour $n = 2$, on a $s \in \{00, 01, 10, 11\}$. En reprenant les matrices correspondantes, on obtient les produits tensoriels suivant :

$$U_{f_{00}} = I \otimes I, U_{f_{01}} = I \otimes Z, U_{f_{10}} = Z \otimes I, U_{f_{11}} = Z \otimes Z$$

On peut généraliser sur l'implémentation en disant :

$$U_f = \bigotimes_{i=0}^n U_i, \quad U_i = \begin{cases} I & \text{si } s_i = 0 \\ Z & \text{si } s_i = 1 \end{cases} \quad (10)$$

Un exemple d'implémentation complète serait alors (pour $s = 101$) :

