Master Systèmes Dynamiques et Signaux

Rapport Bibliographique

Informatique Quantique

Auteur :
M. Pierre Engelstein

Encadrants :
Dr. Nicolas Delanoue
Pr. François Chapeau-Blondeau

Jury :
Pr. Laurent Hardouin
Dr. Nicolas Delanoue
Pr. François Chapeau-Blondeau
Pr. Sébastien Lahaye
Dr. Mehdi Lhommeau
Dr. Remy Guyonneau

Version du 8 janvier 2021

Remerciements

Table des matières

Table des figures

Chapitre 1

Introduction

Chapitre 2

Informatique quantique : éléments de base

2.1 Le qubit

2.1.1 Représentation d'état

Prenons un système composé

2.1.2 Formalisme mathématique

Qubit unique

Le qubit possède deux états de base, correspondant aux états des bits classiques. On les représente par $|0\rangle$, correspondant à l'état 0 classique, et par $|1\rangle$ pour l'état 1 classique. A la différence d'un bit classique, un qubit peut également prendre une infinité d'autres états que ses états de base. La question se pose alors de la mesure : que va-t-on mesurer quand un qubit est dans un état autre que $|0\rangle$ ou $|1\rangle$? C'est là qu'apparaissent les bizarreries de la mécanique quantique. La mesure va donner au hasard 0 ou 1, suivant des probabilités définies.

Pour représenter ce comportement, on note un qubit de la façon suivante :

$$|\psi\rangle = \alpha \cdot |0\rangle + \beta \cdot |1\rangle \tag{2.1}$$

Le qubit est alors représenté par une combinaison linéaire des deux états de base $|0\rangle$ et $|1\rangle$, suivant les coefficients complexes α et β. Ces coefficients représentent les amplitudes de probabilité suivant lesquelles on va mesurer $|0\rangle$ ou $|1\rangle$.

Ces deux coefficients complexes doivent absolument respecter la propriété suivante :

$$\|\alpha\|^2 + \|\alpha\|^2 = 1 \tag{2.2}$$

Multiples qubits

Une fois ces éléments posés, on peut commencer à travailler avec plusieurs qubits, notés n-qubits. Mathématiquement, une combinaison de qubits correspond à un produit tensoriel de deux vecteurs.

Prenons les qubits $|0\rangle$ et $|1\rangle$. La combinaisons en un 2-qbits est donc :

$$|\psi\rangle = |0\rangle \otimes |1\rangle \tag{2.3}$$

qu'on peut écrire plus simplement :

$$|\psi\rangle = |01\rangle \tag{2.4}$$

Un 2-qubit a donc 4 états de bases, représentés par : $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$, et peut donc être la combinaison linéaire de n'importe quel de ces états de base

Intrication

Prenons un 2-qubit formé par la combinaison de 2 qubits :

$$\begin{aligned}|\psi\rangle &= (\alpha_1 \cdot |0\rangle + \beta_1 \cdot |1\rangle) \otimes (\alpha_2 \cdot |0\rangle + \beta_2 \cdot |1\rangle) \\ &= \alpha_1\alpha_2 |0\rangle \otimes |0\rangle + \alpha_1\beta_2 |0\rangle \otimes |1\rangle + \beta_1\alpha_2 |1\rangle \otimes |0\rangle + \beta_1\beta_2 |1\rangle \otimes |1\rangle \\ &= \gamma_1 |00\rangle + \gamma_2 |01\rangle + \gamma_3 |10\rangle + \gamma_4 |11\rangle\end{aligned}$$

On peut donc, si on a un 2-qubit combinaison linéaire de tout les états de bases, le séparer en deux qubits individuels, sur lesquels on va pouvoir agir.

Considérons maintenant le 2-qubit suivant :

$$|\psi\rangle = \gamma_1 |00\rangle + \gamma_2 |11\rangle$$

Il parait évident alors qu'on ne peut pas séparer ce 2-qubit en produit tensoriel de 2 qubits individuels. Dans ce cas, on dit que les deux qubits sont **intriqués** et donc non séparables.

2.2 Représentation géométrique

2.3 Programmer un processeur quantique

Chapitre 3

Algorithme de Deutsch-Jozsa

3.1 Problème à résoudre

Soit une fonction f booléenne définie par

$$\begin{array}{rccl} f: & \{0,1\}^n & \to & \{0,1\} \\ & (x_0, x_1, \ldots, x_n) & \mapsto & y = f(x_0, x_1, \ldots, x_n), \end{array}$$

Définition 1. *Une fonction booléenne f est dite équilibrée si f retourne 0 pour la moitié de ses entrées.*

Définition 2. *Une fonction est dite constante si elle retourne une constante pour toutes ses entrées.*

Remarque 1. *Avec n un entier et comme les fonctions booléennes sont à valeur dans $\{0,1\}$, il n'existe que deux fonctions constantes f_0 et f_1.*

Exemple 1. *Soit f la fonction booléenne $f : \{0,1\}^2 \to \{0,1\}$ définie par la table de vérité suivante :*

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

Cette fonction est équilibrée. On notera qu'elle correspond au classique "ou exclusif". Cette fonction pourrait être représentée par le vecteur de ces valeurs : $(0,1,1,0)$. Elle peut aussi être codée en listant les emplacements où elle est vraie, ici $\{1,2\}$.

Problème 1 (Deutsch-Jozsa)**.** *Etant donnée une fonction f qui est soit équilibrée, soit constante. Le problème de Deutsch-Jozsa est de déterminer si f est constante ou équilibrée.*

3.1.1 Solution classique

Dans le cas classique, il faut effectuer au pire $2^{n-1}+1$ évaluations pour déterminer si f est constante ou équilibrée. Tout d'abord, dès que deux évaluations sont différentes, f est nécessairement équilibrée. De plus, si après avoir évalué 2^{n-1} entrées et obtenu la même valeur, une évaluation supplémentaire nous permet de connaitre dans quelle catégorie f se trouve.

3.1.2 Solution quantique

Dans le cas quantique, ce problème se résout en une seule évaluation quantique de f.

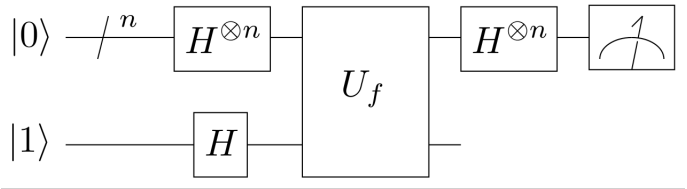


FIGURE 3.1 – Schéma de l'algorithme

Initialisation

On commence avec : $|u_0\rangle = (|0\rangle^{\bigotimes n}) \otimes |1\rangle$: n-qubits à $|0\rangle$ et 1-qubit à $|1\rangle$

Etape 1

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

Etape 2

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

$$o : |x\rangle |y\rangle \mapsto |x\rangle |y \oplus f(x)\rangle .$$

Posons x, on est alors dans l'une des deux situations disjointes suivantes :
— $f(x) = 0$,
— $f(x) = 1$.

Analysons chacune de ces situations, tout d'abord si $f(x) = 0$ alors

$$o : |x\rangle (|0\rangle - |1\rangle) \mapsto |x\rangle (|0\rangle - |1\rangle)$$

Autrement dit $|x\rangle (|0\rangle - |1\rangle$ est un point fixe de o.

Dans l'autre situation, on a $f(x) = 1$ et on en déduit

$$o : |x\rangle (|0\rangle - |1\rangle) \mapsto |x\rangle (|1\rangle - |0\rangle)$$

Autrement dit, dans ce cas, le vecteur $|x\rangle (|0\rangle - |1\rangle)$ est envoyé sur son opposé via o.

Finalement, les deux cas précédents peuvent être résumé sous la forme suivante

$$o : |x\rangle (|0\rangle - |1\rangle) \mapsto (-1)^{f(x)} |x\rangle (|0\rangle - |1\rangle)$$

Par linéarité, on en déduit :

$$|u_2\rangle = \frac{1}{\sqrt{2^{n+1}}} \sum_{x=0}^{2^n-1} (-1)^{f(x)} |x\rangle (|0\rangle - |1\rangle) \tag{3.1}$$

On peut ignorer le dernier qubit $(|0\rangle - |1\rangle)$ comme il est constant. Finalement, on en déduit :

$$|u_2\rangle = \frac{1}{\sqrt{2^{n+1}}} \sum_{x=0}^{2^n-1} (-1)^{f(x)} |x\rangle \tag{3.2}$$

Etape 3

Maintenant qu'on a appliqué notre oracle, on est toujours dans un état "probabiliste", et en mesurant nous n'obtiendrons pas une réponse exacte à notre problème. L'objectif est donc maintenant de ramener les solutions sur un état déterminé pour obtenir la réponse systématique. En appliquant la porte de Hadamard, on va pouvoir forcer un état à apparaître pour un type de fonction f, et le forcer à disparaître dans l'autre cas, ce qui nous permet d'avoir une réponse systématique sur le type de la fonction : est-elle équilibrée ou bien constante ?

On réapplique une porte Hadamard à chaque qubit sortant, ce qui donne :

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

Par linéarité, on a :

$$|u_3\rangle = \frac{1}{2^n} \sum_{x=0}^{2^n-1} \sum_{y=0}^{2^n-1} (-1)^{f(x)} (-1)^{x.y} |y\rangle \tag{3.3}$$

La probabilité $|p|$ de mesurer $|0\rangle^{\bigotimes n}$ est donc :

$$|\frac{1}{2^n}\sum_{x=0}^{2^n-1}(-1)^{f(x)}| \tag{3.4}$$

avec $p = \frac{1}{2^n}\sum_{x=0}^{2^n-1}(-1)^{f(x)}$.

Si on a une fonction $f(x)$ constante, alors chaque élément de la somme retourne la même valeur (1 ou -1 suivant que $f(x)$ retourne 0 ou 1), la somme va donc valoir $\pm 2^n$. Dans le cas où la fonction est équilibrée, on va avoir autant de 1 que de -1, la somme est donc nulle.

On a donc les valeurs suivantes dépendant du type de $f(x)$:

1. Si $f(x)$ est constante : $p = \pm\frac{1}{2^n} \times 2^n = \pm 1$,
2. Si $f(x)$ est équilibrée : $p = \pm\frac{1}{2^n} \times 0 = 0$.

Dans le cas constant, on ne peut donc que mesurer $|0\rangle^{\bigotimes n}$ puisqu'il a une probabilité de 1 d'apparaître. Dans le cas équilibré, on ne mesure jamais $|0\rangle^{\bigotimes n}$ puisque sa probabilité est nulle.

On en conclut que, lorsqu'on effectue une mesure, si on tombe sur $|0\rangle^{\bigotimes n}$ alors la fonction est constante, sinon elle est équilibrée.

3.1.3 Exemple

Prenons une fonction f comme définie précédemment avec $n = 2$, sans savoir si elle est constante ou équilibrée.

Etape 1

On commence avec $|u_0\rangle = |001\rangle$. La première étape est l'application de la porte d'hadamard à $|u_0\rangle$:

$$\begin{aligned}
|u_1\rangle &= H|u_0\rangle = H|0\rangle \otimes H|0\rangle \otimes H|1\rangle \\
&= \frac{1}{2\sqrt{2}}((|0\rangle + |1\rangle) \otimes (|0\rangle + |1\rangle) \otimes (|0\rangle - |1\rangle)) \\
&= \frac{1}{2\sqrt{2}}\{|000\rangle - |001\rangle + |010\rangle - |011\rangle + |100\rangle - |101\rangle + |110\rangle - |111\rangle\} \\
&= \frac{1}{2\sqrt{2}}\{|00\rangle(|0\rangle - |1\rangle) + |01\rangle(|0\rangle - |1\rangle) + |10\rangle(|0\rangle - |1\rangle) + |11\rangle(|0\rangle - |1\rangle)\}
\end{aligned} \tag{3.5}$$

Etape 2 : oracle quantique

On applique à $|u_1\rangle$ l'oracle quantique $|x\rangle|y\rangle \rightarrow |x\rangle|y \oplus f(x)\rangle$:

$$\begin{aligned}|u_2\rangle = \frac{1}{2\sqrt{2}} &|00\rangle\left(|0 \oplus f(00)\rangle - |1 \oplus f(00)\rangle\right)+\\ &|01\rangle\left(|0 \oplus f(01)\rangle - |1 \oplus f(01)\rangle\right)+\\ &|10\rangle\left(|0 \oplus f(10)\rangle - |1 \oplus f(10)\rangle\right)+\\ &|11\rangle\left(|0 \oplus f(11)\rangle - |1 \oplus f(11)\rangle\right)\end{aligned}$$

On peut alors réécrire l'équation de la façon suivante :

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

Par la suite, on va appliquer une porte de Hadamard à $|u_2\rangle$. Le qubit $|0\rangle - |1\rangle$ donne $|1\rangle$ par la cette porte, il est donc constant par rapport à $|u_0\rangle$. On peut donc le retirer de l'équation, ce qui nous donne pour $|u_2\rangle$:

$$|u_2\rangle = \frac{1}{2\sqrt{2}}\left((-1)^{f(00)}|00\rangle + (-1)^{f(01)}|01\rangle + (-1)^{f(10)}|10\rangle + (-1)^{f(11)}|11\rangle\right) \tag{3.6}$$

Matriciellement, on peut donc écrire

$$|u_2\rangle = \begin{pmatrix} (-1)^{f(00)} & 0 & 0 & 0 \\ 0 & (-1)^{f(01)} & 0 & 0 \\ 0 & 0 & (-1)^{f(10)} & 0 \\ 0 & 0 & 0 & (-1)^{f(11)} \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \tag{3.7}$$

Etape 3 : porte de Hadamard

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

$$|u_3\rangle = \frac{1}{2\sqrt{2}}H\left((-1)^{f(00)}|00\rangle + (-1)^{f(01)}|01\rangle + (-1)^{f(10)}|10\rangle + (-1)^{f(11)}|11\rangle\right) \tag{3.8}$$

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

$$
\begin{aligned}
|u_3\rangle &= \frac{1}{4\sqrt{2}} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \begin{bmatrix} (-1)^{f(00)} \\ (-1)^{f(01)} \\ (-1)^{f(10)} \\ (-1)^{f(11)} \end{bmatrix}, \\
&= \frac{1}{4\sqrt{2}} \begin{bmatrix} (-1)^{f(00)} + (-1)^{f(01)} + (-1)^{f(10)} + (-1)^{f(11)} \\ (-1)^{f(00)} - (-1)^{f(01)} + (-1)^{f(10)} - (-1)^{f(11)} \\ (-1)^{f(00)} + (-1)^{f(01)} - (-1)^{f(10)} - (-1)^{f(11)} \\ (-1)^{f(00)} - (-1)^{f(01)} - (-1)^{f(10)} + (-1)^{f(11)} \end{bmatrix}. \qquad (3.9)
\end{aligned}
$$

Si f est constante , alors $(-1)^{f(00)} = (-1)^{f(01)} = (-1)^{f(10)} = (-1)^{f(11)}$. En fonction du fait que $f = 0$ ou bien $f = 1$:

$$
|u_3\rangle = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \text{ ou bien } |u_3\rangle = \frac{1}{\sqrt{2}} \begin{bmatrix} -1 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \qquad (3.10)
$$

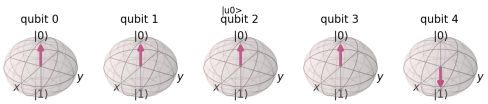
On a donc une probabilité de 1 de mesurer l'état $|00\rangle$.

En revanche, si f est équilibrée, la moitié des valeurs vont valoir $(-1)^0 = 1$ et l'autre moitié $(-1)^1 = -1$. La première ligne du vecteur $|u_3\rangle$ donne donc systématiquement 0, on ne mesure donc jamais l'état $|00\rangle$.

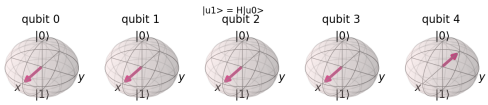
3.2 Visualisation géométrique

Reprenons cet algorithme avec $n = 4$ qubits et affichons l'évolution des états des qubits avec des sphères de bloch.

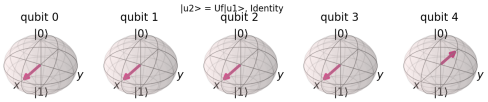
3.2.1 Fonction constante $f_0(x) = 0$



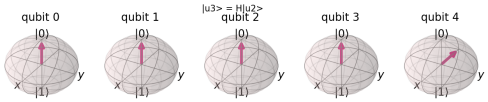
(a) $|u0\rangle = |0001\rangle$



(b) $|u1\rangle = H\,|u0\rangle = |+++-\rangle$



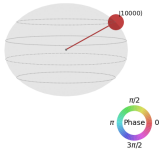
(c) $|u2\rangle = U_f\,|u1\rangle$



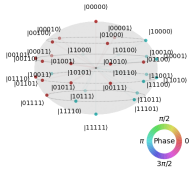
(d) $|u3\rangle = H\,|u2\rangle$

FIGURE 3.2 – Evolution des états pour une fonction f constante, vecteurs d'états séparés

On peut aussi visualiser ces figures avec une shère de bloch représentant les 5 qubits ensembles avec les phases :

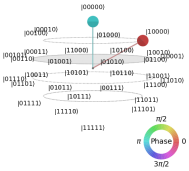


(a) $|u0\rangle = |0001\rangle$



(b) $|u1\rangle = H\,|u0\rangle$

(c) $|u2\rangle = U_f\,|u1\rangle$



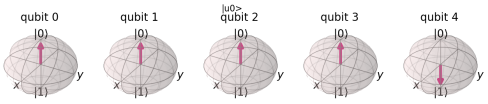
(d) $|u3\rangle = H\,|u2\rangle$

FIGURE 3.3 – Evolution des états pour une fonction f constante

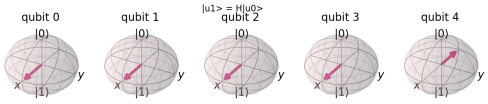
On voit ici l'ensemble des états que prends le registre de sortie. Dans le cas constant, on se retrouve bien à mesure exclusivement la valeur 0 (pour rappel, on ne mesure pas le dernier qubit qui est constant à 1). Pour les deux étapes intermédiaires, on visualise bien qu'on se retrouve dans une certaine superposition des états possibles.

3.2.2 Fonction équilibrée quelconque $f_1(x)$

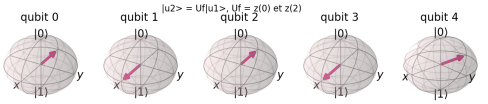
Les deux figues suivantes présentent la même visualisation, pour une fonction équilibrée :



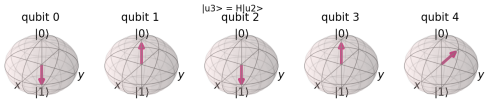
(a) $|u0\rangle = |0001\rangle$



(b) $|u1\rangle = H\,|u0\rangle$

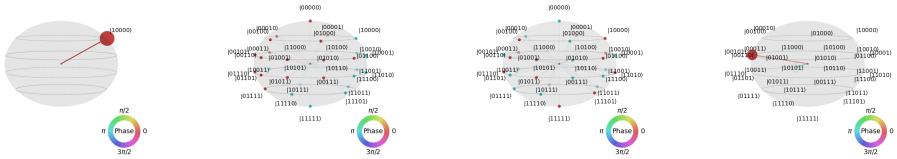


(c) $|u2\rangle = U_f\,|u1\rangle$



(d) $|u3\rangle = H\,|u2\rangle$

FIGURE 3.4 – Evolution des états pour une fonction f équilibrée, vecteurs d'états séparés



(a) $|u0\rangle = |0001\rangle$ (b) $|u1\rangle = H\,|u0\rangle$ (c) $|u2\rangle = U_f\,|u1\rangle$ (d) $|u3\rangle = H\,|u2\rangle$

FIGURE 3.5 – Evolution des états pour une fonction f équilibrée

On voit bien sur cette figure que dans le cas d'une fonction équilibrée, l'état va se situer à un des points indiqués sur la sphère de Bloch mais jamais sur le point 0.

Chapitre 4

Algorithme dérivé de Deutsch-Jozsa : Bernstein-Vazirani

4.1 Problème à résoudre

Soient x et s tels que $x, s \in \{0,1\}^n$.
On pose une fonction f définie par :

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

Exemple 2. *Soit s le mot booléen suivant : $s = 10$. La fonction f a donc la table de véritée 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 2 (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 ?*

4.1.1 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 $\mathcal{O}(n)$

4.1.2 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.

Initialisation

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

Etape 1

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

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 :

$$|u_2\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} (-1)^{s\cdot x \pmod 2}\,|x\rangle \tag{4.1}$$

Etape 3

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

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

$$|u_3\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} (-1)^{s\cdot x \pmod 2} \left(\frac{1}{\sqrt{2^n}} \sum_{y=0}^{2^n-1} (-1)^{x.y} |y\rangle \right) \tag{4.2}$$

$$= \frac{1}{2^n} \sum_{x=0}^{2^n-1} \sum_{y=0}^{2^n-1} (-1)^{(s\cdot x \pmod 2)+x.y} |y\rangle \tag{4.3}$$

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

4.1.3 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 $|u_0\rangle$:

$$|u_1\rangle = H|u_0\rangle = H|0\rangle \otimes H|0\rangle \tag{4.4}$$

$$= \frac{1}{2}((|0\rangle + |1\rangle) \otimes (|0\rangle + |1\rangle)) \tag{4.5}$$

$$= \frac{1}{2}\{|00\rangle + |01\rangle + |10\rangle + |11\rangle\} \tag{4.6}$$

Etape 2 : oracle quantique

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

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

Etape 3 : porte de Hadamard

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

$$|u_3\rangle = \frac{1}{2}H((|00\rangle + |01\rangle - |10\rangle + |11\rangle)) \tag{4.7}$$

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

$$|u_3\rangle = \frac{1}{4}\begin{bmatrix}1 & 1 & 1 & 1\\ 1 & -1 & 1 & -1\\ 1 & 1 & -1 & -1\\ 1 & -1 & -1 & 1\end{bmatrix}\begin{bmatrix}1\\ 1\\ -1\\ -1\end{bmatrix}, \tag{4.8}$$

$$= \frac{1}{4}\begin{bmatrix}0\\ 0\\ 4\\ 0\end{bmatrix}. \tag{4.9}$$

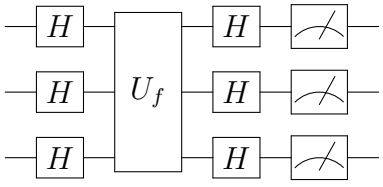
Lors de la mesure, on va obtenir l'état $|10\rangle$ avec une probabilité de 1, qui était bien notre mot binaire s de départ.

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, peut 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.

4.1.4 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 :

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

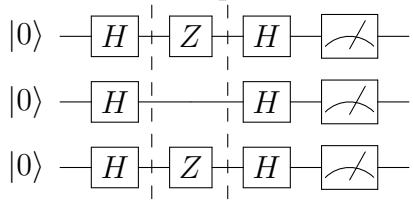
Pour $n = 2$, on a $s \in \{00, 01, 10, 11\}$. En reprenant les matrices correspondantes, on obtient les produits tensoriels suivant :

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

On peut généraliser sur l'implémentation en disant :

$$U_f = \bigotimes_{i=0}^{n} U_i,\ U_i = \begin{cases} I & \text{si } s_i = 0 \\ Z & \text{si } s_i = 1 \end{cases} \tag{4.10}$$

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



Chapitre 5

Algorithme de Grover

5.1 Rappels d'algèbre : projection et reflection

Soient deux vecteurs $\overrightarrow{u}$ et $\overrightarrow{v}$, avec $\overrightarrow{v}$ normalisé.

Définition 3. *La matrice de projection P de $\overrightarrow{u}$ sur $\overrightarrow{v}$ est définie par $P = \overrightarrow{v} \cdot \overrightarrow{v^T}$.*

Définition 4. *La matrice de reflection R de $\overrightarrow{u}$ par rapport à $\overrightarrow{v}$ est définie par $R = 2\overrightarrow{v} \cdot \overrightarrow{v^T} - I$.*

Exemple 3. *Prenons $\overrightarrow{u} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$ et $\overrightarrow{v} = \begin{bmatrix} 1 \\ -2 \end{bmatrix}$.*

On projete $\overrightarrow{u}$ sur $\overrightarrow{v}$:

$P = \frac{\overrightarrow{v} \cdot \overrightarrow{v^T}}{\|v\|^2} = \begin{bmatrix} \frac{1}{\sqrt{5}} & \frac{-2}{\sqrt{5}} \\ \frac{-2}{\sqrt{5}} & \frac{4}{\sqrt{5}} \end{bmatrix}$

Soit : $\overrightarrow{u_v} = P\overrightarrow{u} = \begin{bmatrix} -0.8 \\ 1.6 \end{bmatrix}$

Exemple 4. *Prenons à nouveau $\overrightarrow{u} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$ et $\overrightarrow{v} = \begin{bmatrix} 1 \\ -2 \end{bmatrix}$. On effectue une reflection de $\overrightarrow{u}$*

$R = 2 \times \frac{\overrightarrow{v} \cdot \overrightarrow{v^T}}{\|v\|^2} - I = 2 \times \begin{bmatrix} \frac{1}{\sqrt{5}} & \frac{-2}{\sqrt{5}} \\ \frac{-2}{\sqrt{5}} & \frac{4}{\sqrt{5}} \end{bmatrix} - \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

La première étape est la double projection $2 \times P$, ce qui donne le vecteur $\begin{bmatrix} -1.6 \\ 3.2 \end{bmatrix}$.

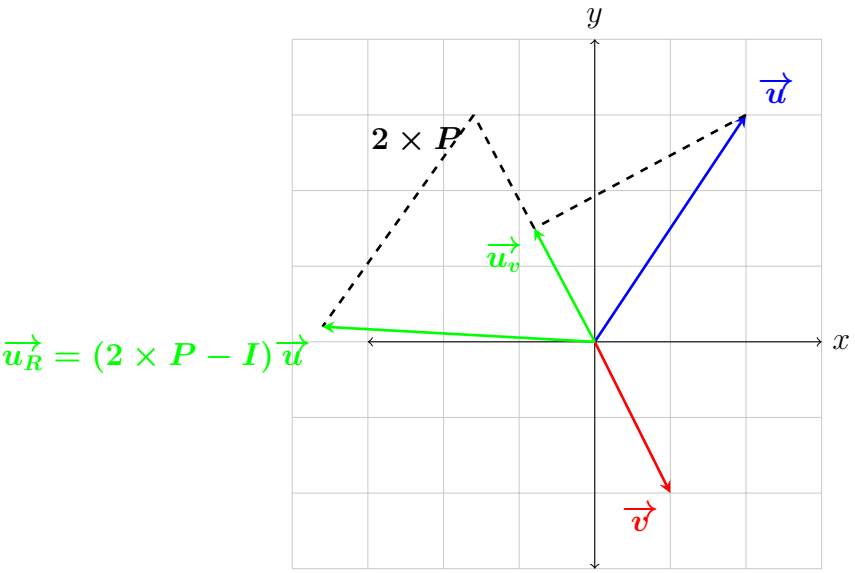
La deuxième étape est d'enlever le vecteur initial, ce qui donne le vecteur $\overrightarrow{u_R} = \begin{bmatrix} -3.6 \\ 0.2 \end{bmatrix}$.

On peut vérifier les angles θ_{UV} et θ_{VU_R} :

$\theta_{UV} = \arccos(\frac{u \cdot v}{\|u\|\|v\|}) = \arccos(\frac{-4}{\sqrt{13} \times \sqrt{5}}) = 119.7°$

$\theta_{VU_R} = \arccos(\frac{v \cdot u_R}{\|v\|\|u_R\|}) = \arccos(\frac{-4}{\sqrt{5} \times \sqrt{13}}) = 119.7°$

Les deux angles sont bien égaux, on a effectué une reflection.



5.2 Problème à résoudre

Soit une base de données non triée à N entrées. Nous voulons trouver un algorithme permettant de chercher efficacement un enregistrement dans cette base.

5.2.1 Principe de l'algorithme

L'algorithme de Grover permet de résoudre ce problème en quantique, en disposant de N qubits intriqués pour calculer 2^N état (donc si on a N entrées dans la base, il nous faut $log_2(N)$ qubits intriqués). Dans le cas de cet algorithme, on considère le problème suivant :

On marque $\{0, 1, 2, ..., N-1\}$ les enregistrements de la base de données, et on dénote ω l'état inconnu recherché. On dispose de la fonction suivante :

$$f(x) = \begin{cases} 1, & \text{si x vérifie le critère } \omega \\ 0, & \text{sinon} \end{cases}$$

A la fin, on obtient un set de résultat. Or, lors de la mesure on va avoir au hasard une des solutions suivant les probabilités de chaque état, alors qu'on cherche juste à savoir la (ou les) bonnes solutions. On rajoute donc une amplification d'amplitude permettant d'augmenter les probabilités des bons résultats et de diminuer celles des mauvais.

Initialisation

On commence avec : $|u_0\rangle = (|0\rangle^{\bigotimes n}) \otimes |1\rangle$: n-qubits à $|0\rangle$ et 1-qubit à $|1\rangle$

Etape 1

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

On pose alors $|s\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle$

Etape 2 : opérateurs de Grover

On définit les deux opérateurs suivants :

$U_w = I - 2|w\rangle\langle w|$, avec w état cible correspondant à la solution du problème (amplitude de 1 sur l'état visé, amplitude nulle sur le reste)

$U_s = 2|s\rangle\langle s| - I$

Remarque 2. *On reconnait ici que ces deux opérateurs sont semblables à la reflection vue dans la partie 1.*

On effectue ici un changement de base : au lieu de continuer les calculs dans la base canonique $\{|0\rangle, |1\rangle\}$, on se place dans la base $\{|w\rangle, |s\rangle\}$

Inversion d'amplitude L'opérateur U_w effectue l'inversion de l'amplitude de l'état cible, tandis que l'opérateur U_s effectue le miroir des amplitudes par rapport à la moyenne.

On applique U_w puis U_s :

$U_w|s\rangle = (I - 2|w\rangle\langle w|)|s\rangle = |s\rangle - 2|w\rangle\langle w|s\rangle$

Or, $\langle w|s\rangle$ est un produit scalaire. $|w\rangle$ est définit plus haut, et $|s\rangle$ est l'état équiprobable obtenu après la porte de hadamard. Le résultat est donc $\langle w|s\rangle = \frac{1}{\sqrt{2^n}}$. On peut donc réécrire :

$|u_3\rangle = U_w|s\rangle = |s\rangle - \frac{2}{\sqrt{2^n}}|w\rangle$

Miroir à la moyenne On applique ensuite l'opérateur U_s au résultat de U_w. On peut voir qu'en pratique U_s effectue un miroir de $|u_3\rangle$ par rapport à $|s\rangle$.

$$\begin{aligned} U_s |u_3\rangle &= (2 |s\rangle \langle s| - I)(|s\rangle - \frac{2}{\sqrt{2^n}} |w\rangle) \\ &= 2 |s\rangle \langle s|s\rangle - |s\rangle - \frac{4}{\sqrt{2^n}} |s\rangle \langle s|w\rangle + \frac{2}{\sqrt{2^n}} |w\rangle \\ &= 2 |s\rangle - |s\rangle + \frac{4}{\sqrt{2^n}} \times \frac{1}{\sqrt{2^n}} |s\rangle + \frac{2}{\sqrt{2^n}} |w\rangle \\ &= |s\rangle - \frac{4}{2^n} |s\rangle + \frac{2}{\sqrt{2^n}} |w\rangle \\ |u_4\rangle &= \frac{2^n - 4}{2^n} |s\rangle + \frac{2}{\sqrt{2^n}} |w\rangle \end{aligned} \tag{5.1}$$

Plus généralement, cette application de U_w puis U_s reviens à appliquer la matrice suivante à l'état d'entrée, dans la base $\{|w\rangle, |s\rangle\}$: $\begin{bmatrix} 1 & \frac{2}{\sqrt{2^n}} \\ \frac{-2}{\sqrt{N}} & \frac{2^n-4}{2^n} \end{bmatrix}$

5.2.2 Exemple

Prenons par exemple une base de données de 4 bits ($n = 4$), avec l'état $|w\rangle$ cible valant l'état $|0100\rangle$ (amplitude de 1 sur cet état, et de 0 sur l'ensemble de 15 autres).

On initialise un (n+1)-qubit à l'état suivant :

$$|u_0\rangle = |00001\rangle \tag{5.2}$$

Etape 1

On applique la porte de Hadamard à l'état initial $|u_0\rangle$:

$$|u_1\rangle = \frac{1}{16} \sum_{x=0}^{15} |x\rangle (|0\rangle - |1\rangle) \tag{5.3}$$

On obtient donc les deux états formant notre base pour les calculs suivant : $|s\rangle = |u_1\rangle$ et $|w\rangle$.

Etape 2 : Opérateur de Grover

On applique la transformation $U_s U_w = \begin{bmatrix} 1 & \frac{2}{\sqrt{2^n}} \\ \frac{-2}{\sqrt{2^n}} & \frac{2^n-4}{2^n} \end{bmatrix}$ pour $n = 4$ soit $U_s U_w = \begin{bmatrix} 1 & \frac{1}{2} \\ -\frac{1}{2} & \frac{3}{4} \end{bmatrix}$:

$$|u_2\rangle = U_s U_w \cdot |s\rangle = \frac{3}{4}|s\rangle + \frac{1}{2}|w\rangle \tag{5.4}$$

On voit que l'état cible $|w\rangle$ est passé d'une amplitude de 0 à une amplitude de 0.5. On peut effectuer l'opération plusieurs fois pour obtenir un résultat voulu. La figure suivante montre l'évolution des amplitudes de $|s\rangle$ et de $|w\rangle$ pour $n = 16$, pour 1000 itérations de l'opérateur. On observe qu'on arrive à l'état voulu $|w\rangle$ mais qu'on ne reste pas à cet état une fois atteint. Cela montre bien qu'il y a un nombre optimal d'itérations à effectuer, à ne pas dépasser. (la courbe verte sert d'indicateur, pour vérifier qu'on reste dans un état valide où la somme des amplitudes vaut bien 1)

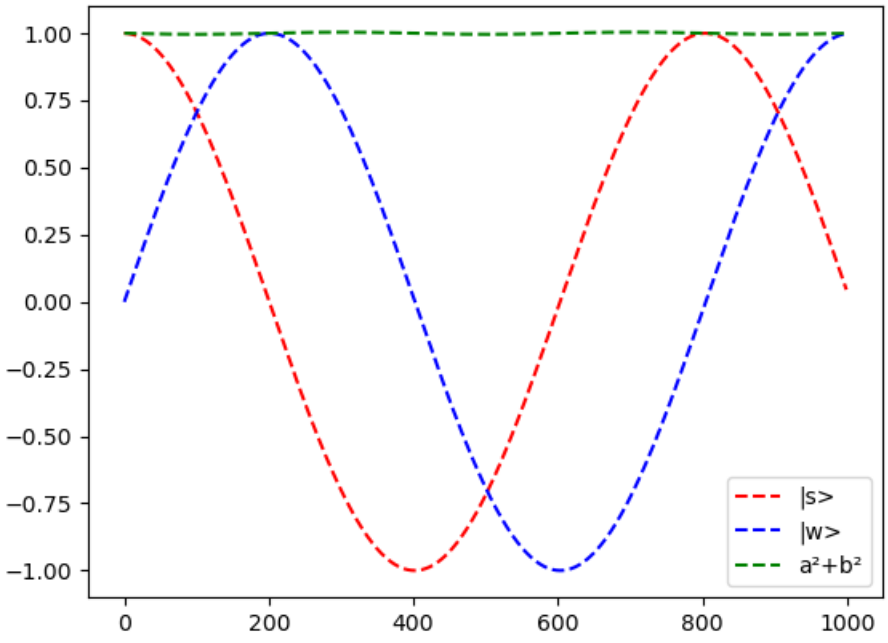


FIGURE 5.1 – Evolution des amplitudes pour n=16, sur 1000 itérations

5.2.3 Implémentation

Simulation sur un ordinateur classique

Data: w vector of size 2^n of 0 with target index to 1 ;
Output: x vector of amplitudes (largest amplitude corresponding to wanted index)
begin
  s vector of size 2^n of $\frac{1}{\sqrt{2^n}}$ $N \longleftarrow 2^n$;
  $N_{iter} \longleftarrow floor(\frac{\pi}{4}\sqrt{N})$;
  `/* Compute grover operator */`
  $U_w \longleftarrow I_N - 2w \cdot w^T$;
  $U_s \longleftarrow 2s \cdot s^T - I_N$;
  $g \longleftarrow U_s \cdot U_w$;
  $x \longleftarrow s$;
  `/* Apply grover operator` N_{iter} `times */`
  for $i = 0$ **to** N_{iter} **do**
    $x \longleftarrow g \cdot x$;
  end
end

Algorithme quantique

```
import numpy as np
import math
import random
from qiskit import QuantumRegister, ClassicalRegister, QuantumCircuit

var_qubits = QuantumRegister(4, name='v')
clause_qubits = QuantumRegister(4, name='c')
output_qubits = QuantumRegister(1, name='out')
cbits = ClassicalRegister(4, name='cbits')
qc = QuantumCircuit(var_qubits, clause_qubits, output_qubits, cbits)
qc.initialize([1, -1]/np.sqrt(2), output_qubits)
qc.h(var_qubits)
qc.barrier()
```