Une introduction à l'informatique quantique à la portée des étudiants en mathématiques ou informatique de premier cycle universitaire.

Rémi Lajugie

mail prenom.nom@ac-limoges.fr

Février 2021, version non finalisée

Résumé

Dans cet article, on cherche à familiariser avec l'informatique quantique le lecteur non-familier du formalisme de la mécanique quantique et plus habitué au formalisme classique des mathématiques ou de l'informatique. On commence par donner une autre vue sur l'informatique classique en interprétant de manière vectorielle les opérations usuelles sur les bits classiques. On s'intéresse ensuite à généraliser cette interprétation pour aboutir à la notion de bit quantique (qbit). On présentera ensuite les portes logiques quantiques qui régissent l'évolution des systèmes de qbits. Enfin, on présente l'algorithme de Deutsch qui a été le premier algorithme quantique dont on a démontré qu'il était intrinsèquement plus rapide que n'importe quel algorithme classique pour résoudre un problème.

Cette note vise à présenter quelques rudiments concernant l'informatique quantique. Elle est rédigée par un non-physicien et a pour objet d'être lue et comprise par des étudiants ayant un bagage solide en mathématiques (typiquement un étudiant de première ou deuxième année de classe préparatoire scientifique MP/PC/PSI/PT/MPI). On suppose uniquement connus quelques rudiments d'algèbre linéaire. On pourra se servir de cette note et des références y figurant comme base d'un travail d'initiative personnelle encadré (TIPE). Les principales références utilisées pour ce travail sont (Aaronson, 2013) et (Nielsen and Chuang, 2002).

1 Une formulation vectorielle de l'informatique classique

On sait qu'un ordinateur classique ne traite que des séquences de 0 et de 1. Au plus bas niveau de l'ordinateur, ces chiffres binaires, appelés *bits* (pour *binary digits*) sont traités au moyen de portes logiques traitant un ou plusieurs bits, par exemple les portes ET, OU, NON, OU EXCLUSIF, NAND etc.

1.1 Représentation vectorielle des bits

Le fait de représenter les bits par les symboles 0 et 1 est purement conventionnel. On pourrait choisir toute autre représentation.

Un seul bit. On va, dans la suite de cette note, représenter les bits comme des vecteurs du plan. On associe donc la valeur 0 au vecteur de coordonnées $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$ de la base canonique et la valeur 1 est associée au

vecteur $\begin{pmatrix} 0 \\ 1 \end{pmatrix}$ Il est usuel, dans le contexte de l'informatique quantique de noter $|0\rangle$ le vecteur $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$ et $|1\rangle$ le vecteur $\begin{pmatrix} 0 \\ 1 \end{pmatrix}$.

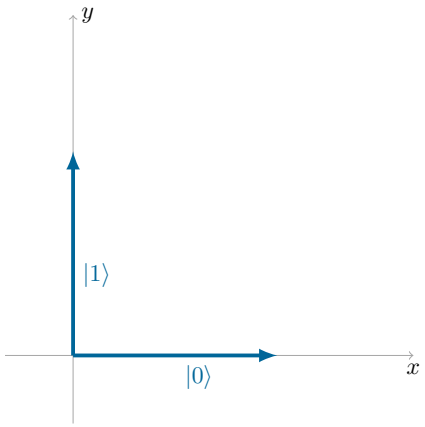

FIGURE 1 – Les deux vecteurs de base $|0\rangle$ et $|1\rangle$.

Pour le moment il s'agit seulement d'une représentation abstraite, qui sera commode pour les calculs comme on le verra plus loin.

Deux bits. Considérons deux bits a et b. On s'intéresse à la représentation de ab. Avec le choix que nous avons fait, il semble assez logique de représenter les 2 bits ab (par exemple 01) par un *couple* de vecteurs $(|a\rangle, |b\rangle)$ (sur l'exemple 01, cela correspond à $(\begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix})$. C'est certes tout à fait sensé mais cela fait perdre le caractère vectoriel car si l'on sait bien faire des calculs sur des vecteurs, il n'est a priori pas aisé d'en faire sur des couples de vecteurs.

C'est pour cela qu'on va représenter une paire de bits par un vecteur de dimension 4. On utilise pour cela la représentation suivante (il s'agit en fait d'associer à chacun des couples de bits possibles, un vecteur de la base canonique de $\mathbb{C}^4$) :

$$|00\rangle = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, |01\rangle = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, |10\rangle = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}, |11\rangle = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}.$$

L'opération que nous venons d'introduire s'appelle le *produit tensoriel* de $|a\rangle$ et de $|b\rangle$ et se note $|a\rangle \otimes |b\rangle$.

n bits. On considère désormais une séquence de n bits $a_1a_2a_3 \dots a_n$.

De manière générale, on va définir par récurrence $|a_1a_2a_3 \dots a_n\rangle = |a_1\rangle \oplus |a_2\rangle \oplus |a_3\rangle \dots \oplus |a_n\rangle$.

Bien évidemment, la représentation proposée ci-dessus a le défaut de ne pas être très compacte puisque la dimension de l'espace vectoriel considéré est de 2^n. Mais c'est cette représentation qui va nous être utiles lorsque nous allons passer des bits classiques aux bits quantiques.

1.2 Portes logiques classiques

Portes ET, OU, NON. Un ordinateur effectue des opérations sur les bits au moyen de portes logique. Il en existe une grande variété mais il suffit des trois portes ET, OU, NON pour réaliser ce que l'on appelle *l'algèbre de Boole* et cela sera suffisant pour l'exposé que nous faisons ici. Le lecteur curieux pourra consulter des références sur les portes logiques comme celles présentées dans (Aho et al., 1996), on y trouvera par exemple comment on peut réaliser concrètement un additionneur avec retenue.

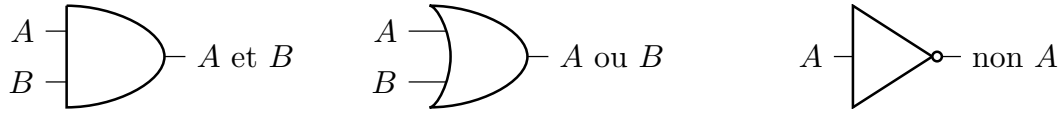

FIGURE 2 – Symboles conventionnels pour les portes logiques classiques.

Tables de vérité. En général, on donne les tables de vérité des portes logiques, c'est à dire le tableau des diverses valeurs possibles pour les sorties en fonction des entrées.

a	b	a ET b
0	0	0
0	1	0
1	0	0
1	1	1

a	b	a OU b
0	0	0
0	1	1
1	0	1
1	1	1

a	NON A
0	1
0	0

FIGURE 3 – Table de vérité des opérateurs de l'algèbre de Boole.

Interprétation matricielle. Des trois opérateurs que l'on a vu, un seul n'opère que sur un bit. C'est l'opérateur NON. Avec la représentation vectorielle que nous avons vu au paragraphe 1.1, à savoir associer $|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ au bit 0 et $|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ au bit valant 1, l'effet de la porte non est celui d'une application qui envoie $|0\rangle$ sur $|1\rangle$ et vice-versa.

On peut alors représenter la porte non par la matrice, appelée *porte de Pauli*, suivante :

$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$.

Un calcul matriciel montre que l'on a bien $X|0\rangle = |1\rangle$ et $X|1\rangle = |0\rangle$.

En ce qui concerne les portes ET et OU, elles opèrent à priori sur des espaces de dimension 4 dont une base est $|00\rangle, |01\rangle, |10\rangle, |11\rangle$ et renvoient un élément d'un espace de dimension 2, $|0\rangle$ et $|1\rangle$. On va donc pouvoir les représenter par les matrices 2 lignes par 4 colonnes suivantes :

— Pour le ET logique : $\begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

— Pour le OU logique : $\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{pmatrix}$

.

2 Bases de l'informatique quantique

Dans cette partie, on va s'intéresser à développer les bases de l'informatique quantique sans rentrer dans plus de développements physique ou mathématiques que nécessaire. Le lecteur curieux pourra se lancer dans l'étude du livre (Nielsen and Chuang, 2002).

2.1 Les qbits

L'informatique quantique repose sur des qbits qui sont des généralisations des bits classiques. Nous allons les décrire comme des objets mathématiques sans nous intéresser à leur réalisation concrète (qui est possible au travers de nombreux phénomènes physiques régis par la mécanique quantique).

En informatique quantique, il est possible d'avoir des états différents des seuls $|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ ou $|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$. Plus précisément, les qbits peuvent être dans un état correspondant à n'importe quelle combinaison linéaire de $|0\rangle$ et $|1\rangle$ de norme 1.

Mathématiquement, un qbit peut être dans n'importe quel état $|psi\rangle = \alpha|0\rangle + \beta|1\rangle$ où $\alpha, \beta \in \mathbb{C}$ et $|\alpha^2| + |\beta^2| = 1$.

On dit alors que les qbits sont alors dans une *superposition* des états de base $|0\rangle$ et $|1\rangle$.

On va maintenant lister les principales règles qui gouvernent le comportement des bits quantiques. La troisième nous permettra notamment d'interpréter le rôle des coefficients α et β mentionnés plus haut.

2.2 Quelques règles d'évolution des bits quantiques.

On va ici lister les principes d'évolution des qbits sous la forme de trois postulats. On trouvera de plus amples détails dans (Nielsen and Chuang, 2002).

Postulat 1 : l'espace des états. A tout système quantique isolé, il est possible d'associer un espace vectoriel (sur le corps des nombres complexes), Hilbertien (c'est à dire muni d'un produit scalaire hermitien). Pour des compléments sur les produits scalaires hermitiens, on pourra se référer à l'annexe du présent document (en section 5).

En pratique, dans cet article, les espaces vectoriels seront tous *de dimension finie*. Il n'est pas utile de recourir à toute la théorie des espaces Hilbertiens généraux. Celle des espaces hermitiens (cf section 5), à la lisière du programme des premières années de licence ou de classes préparatoires, est suffisante.

Par exemple, pour un système de 1 qbit, on pourra considérer l'espace hermitien $\mathbb{C}^2$ muni du produit scalaire hermitien usuel : $\langle x, y\rangle = {}^t\overline{x}y$.

Pour un système à 2 qbits, on aura recours à l'espace $\mathbb{C}^4$ et plus généralement pour un système de n qbits, on aura recours à l'espace $\mathbb{C}^{2^n}$.

Postulat 2 : l'évolution du système. En informatique quantique, on a rarement besoin d'utiliser l'équation de Schrodinger complète pour décrire l'évolution des qbits. C'est pourquoi nous nous contentons d'en donner une version discrète. Le deuxième postulat peut donc s'exprimer ainsi :

Soit $|\psi\rangle$ l'état d'un ensemble de qbits à un instant t et $|\psi'\rangle$ l'état du même ensemble de qbits à un instant t'. Alors il existe une transformation *unitaire* telle que $|psi'\rangle = U|\psi\rangle$. Réciproquement, toute matrice unitaire correspond à une transformation valide d'un système quantique.

En particulier, ce postulat impose donc que les matrices intervenant dans les transformations des qbits soient *inversibles*.

Par exemple, la porte NON donnée par $\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ est unitaire, et correspond donc à une transformation valide d'un qbit. En revanche, les matrices $\begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$ et $\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{pmatrix}$ correspondant au ET et au OU ne pourront pas correspondre à des transformations valides de qbit [1].

Postulat 3 : mesure d'un système. Dernier postulat utile pour cet article : la mesure d'un système quantique dans un état superposé $\alpha_1 |\psi_1\rangle + \alpha_2 |\psi_2\rangle + \ldots + \alpha_2 |\psi_n\rangle$ avec $|\psi_k\rangle$ les états de base du système et $|\alpha_1|^2 + |\alpha_2|^2 + \ldots + |\alpha_n|^2 = 1$ donnera une observation étant un seul des $|\psi_k\rangle$ avec probabilité $|\alpha_k|$.

2.3 Les portes logiques quantiques

Du point de vue informatique, les postulats de la mécanique quantique nous permettent de définir des *portes logiques quantiques* qui vont travailler sur des systèmes de n qbits et leur appliquer une transformation qui sera unitaire. En d'autres termes, si on a un système de n qbits dans un état $|psi\rangle$ (vecteur de taille 2^n), on va lui appliquer une transformation unitaire (représentée par une matrice carrée U de taille 2^n également et telle que ${}^t\overline{U}U = I$).

D'après le postulat 2, on sait qu'il est possible de réaliser physiquement toutes les transformations unitaires. La seule contrainte sur un ordinateur quantique est d'employer des portes logiques correspondant à des transformations unitaires.

On va donner ici donner quelques portes utiles pour la suite de l'article. Le lecteur intéressé pourra en trouver bien d'autres dand (Aho et al., 1996).

Porte de Pauli. Il s'agit d'une porte opérant sur un seul qbit :

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = |0\rangle\langle 1| + |1\rangle\langle 0|$$

Elle agit comme un inverseur :

$$X|0\rangle = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = |1\rangle.$$

Porte de Hadamard. La porte de Hadamard agit sur un seul qbit et lui applique une transformation permettant de *symétriser* les états de base :

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

On a alors $H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ et $H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix}$.

1. Avec un peu d'astuce, il est néanmoins possible d'implémenter des transformations logiques correspondant au ET et au OU sur un ordinateur quantique. Mais il faut rajouter des qbit «superflu». C'est ce que l'on appelle les *portes de Toffoli*.

Porte CNOT. La porte logique CNOT (non-conditionnel) agit sur deux qbit. L'idée est d'inverser le second qbit si et seulement si le premier (qbit de contrôle) est à la valeur $|1\rangle$.

Voici la table de vérité associée aux bits classiques :

Entrée $(qbitcontrle, qbitcible$	Sortie $(qbitcontrle, qbitcible$
00	00
01	01
10	11
11	10

.

Cette table de vérité suggère de considérer la transformation unitaire suivante :

$$\text{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

3 Un problème sur lequel travailler avec un ordinateur quantique est meilleur qu'avec un ordinateur classique

Dans cette partie, nous allons enfin voir l'intérêt de travailler avec des qbits plutôt qu'avec des bits.

Pour cela nous allons nous intéresser à un problème sur lequel un ordinateur quantique fait moins de travail que n'importe quel ordinateur classique.

3.1 Présentation du problème

Fonctions booléennes. Considérons une fonction booléenne, c'est à dire une fonction $f : \{0,1\} \to \{0,1\}$.

A priori, il n'existe que quatre fonctions de ce genre :

— La fonction constante égale à 0,
— la fonction constante égale à 1,
— la fonction identité ($f(0) = 0, f(1) = 1$),
— la fonction NON ($f(0) = 1, f(1) = 0$).

Le problème à résoudre. Supposons qu'un oracle nous fournisse une boîte noire contenant l'une des quatre fonctions booléennes possibles. La seule chose que nous pouvons faire c'est d'utiliser cette boite noire pour demander de calculer des valeurs.

On se pose alors la question suivante :

Est-ce que l'oracle nous a donné une fonction constante ou non ?

3.2 Résolution classique du problème

Dans un contexte informatique classique, pour répondre à cette question avec certitude, nous avons besoin d'interroger la boîte noire deux fois. En effet, après une seule utilisation de la boîte noire nous ne pouvons toujours pas discerner si la fonction est constante ou non.

Nous allons voir que dans un contexte d'informatique quantique, il suffit de ne n'interroger qu'une seule fois notre boîte noire. Néanmoins il faut que la boîte noire fournie soit quantique.

3.3 Résolution quantique du problème

Irréversibilité de la boîte noire et câblage d'un oracle quantique. Dans ce paragraphe, nous devons nous assurer que la boîte noire que nous fournit notre oracle peut bien se coder de manière quantique, sinon le problème auquel nous nous attaquons est sans objet. Dans le cadre classique, on pouvait imaginer que la boîte noire soit n'importe quelle fonction, mais dans le cadre quantique, les transformations appliquées aux qbit doivent être unitaires. Or, des quatre fonctions potentielles, seules deux sont unitaires (les deux autres ne sont pas inversible, ce qui est le gros problème).

Néanmoins, notre oracle peut quand même câbler ces fonctions non réversibles de manière quantique. Il s'agit pour lui, d'ajouter un deuxième fil, sur lequel on met un qbit dans l'état $|0\rangle$. L'idée est donc d'avoir deux fils en entrée de notre porte quantique : l'une porte l'entrée, le qbit dont on veut calculer l'image par f et l'autre avec $|0\rangle$. Le premier fil restera inchangé, il portera toujours l'entrée après passage dans la boîte noire, et le deuxième portera la sortie $f\left(|x\rangle\right)$.

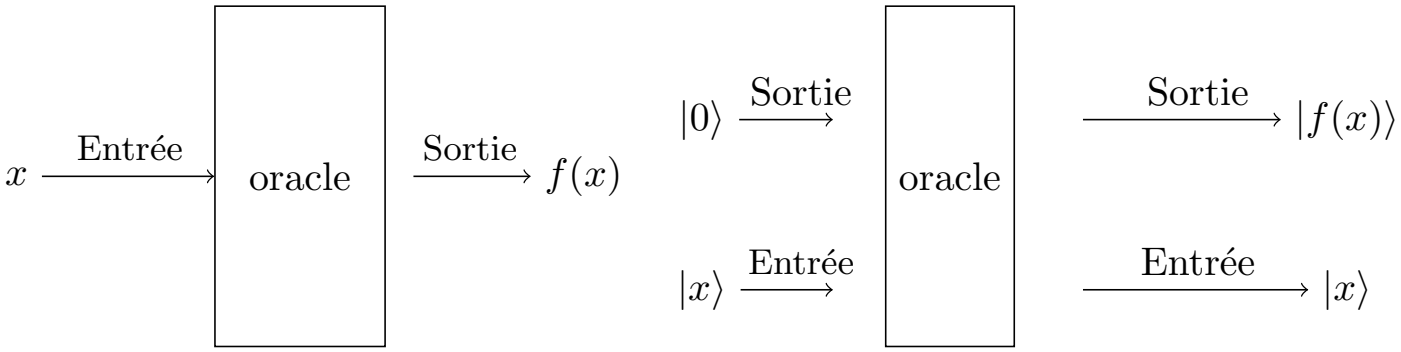

FIGURE 4 – Allure d'un oracle classique à gauche et à droite d'un oracle quantique

Les quatre fonctions booléennes peuvent alors s'écrire comme des transformations sur des qbits de la manière suivante :

- **Fonction constante valant** 0 : il suffit de ne rien faire subir aux deux qbits ;
- **Fonction constante valant** 1 : il suffit d'appliquer une porte de Pauli X (porte NOT de matrice $\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ au fil de sortie ;
- **Fonction identité :** il faut appliquer une porte CNOT dont le bit de contrôle est le fil d'entrée. Ainsi quand le qbit d'entrée est à $|1\rangle$, celui de sortie est mis à $|1\rangle$ et quand le qbit d'entrée est à $|0\rangle$, celui de sortie est mis à $|1\rangle$.
- **Fonction NON :** il faut appliquer la porte CNOT précédente et ensuite ajouter une porte de Pauli pour le qbit de sortie.

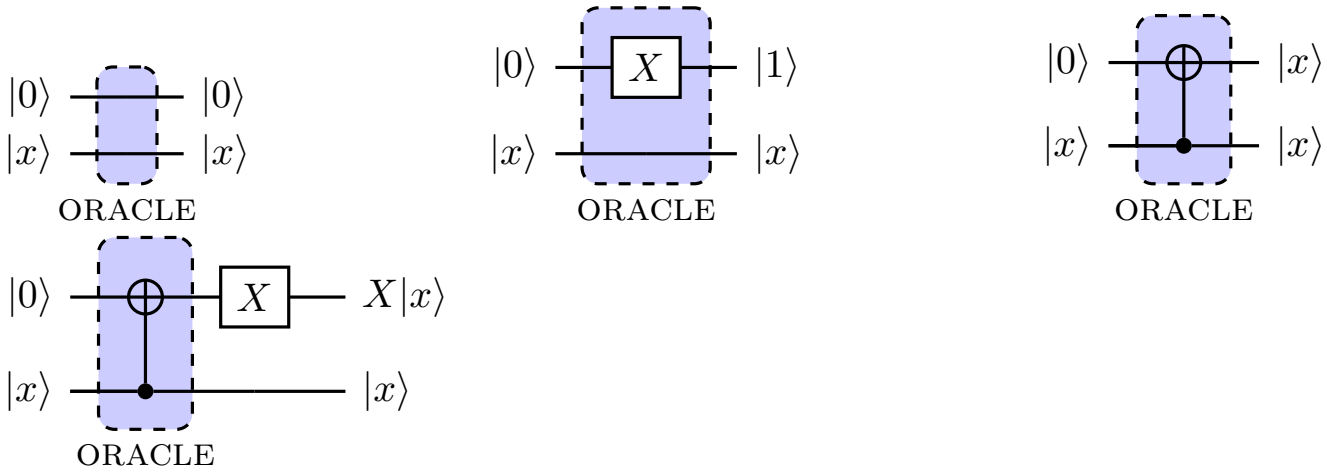

FIGURE 5 – Les quatre circuits quantiques permettant de réaliser les fonctions booléennes.

La méthode de Deutsch. En 1985, pour résoudre cet étonnant problème, David Deutsch a proposé, dans l'article (Deutsch, 1985), un circuit quantique ne nécessitant qu'un seul appel à la version quantique de l'oracle. Il s'agit d'un astucieux enchaînement de portes de Hadamard H et de Pauli X, comme décrit dans la figure **??**.

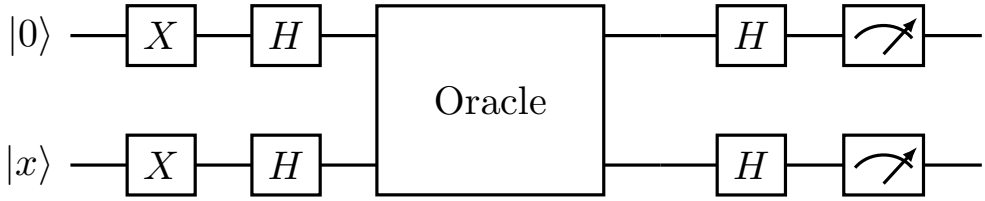

FIGURE 6 – Circuit quantique de Deutsch.

On peut alors démontrer que, si l'on envoie un doublet de qbits d'entrée $|00\rangle$, c'est à dire $|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ sur le premier fil et $|0\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ sur le second, on obtient en sortie :

— $|11\rangle$ si la fonction est constante,
— $|01\rangle$ si la fonction n'est pas constante.

La démonstration peut se faire avec du produit matriciel.

Considérons la paire de qbits $|00\rangle$. Après passage dans les portes de Pauli X et de Hadamard. Chacun des qbit a été passé à l'état $HX|0\rangle$ soit encore en explicitant le calcul matriciel :

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

Maintenant, passons dans l'oracle puis dans la porte de Hadamard qui suit.

— Si l'oracle est constant à $|0\rangle$, d'après son implémentation en porte quantique, rien n'est changé, on passe donc directement dans la porte de Hadamard. On fait le calcul matriciel et on se rend compte que chacun des qbit est envoyé sur $|1\rangle$. Ainsi, la mesure donnera, avec probabilité 1, l'état $|11\rangle$.
— Si l'oracle est constant égal à $|1\rangle$, d'après son implémentation en porte quantique, il se passe pour le qbit d'entrée la même chose qu'au point précédent. En revanche, le qbit du fil de sortie est changé

multiplié par la porte de Pauli X donc le qbit du fil de sortie passe de l'état $\frac{|0\rangle-|1\rangle}{\sqrt{2}}$ à l'état $\frac{|1\rangle-|0\rangle}{\sqrt{2}}$. Le passage dans la porte de Hadamard l'envoie sur l'état $-|1\rangle$. Ainsi avec probabilité 1, le qbit du fil de sortie est mesuré à $|1\rangle$. On rappelle en effet que les probabilités d'occurrence des mesures sont proportionnelles au carré du coefficient devant $|0\rangle$ et $|1\rangle$. Ici, $(-1)^2$ donne bien une probabilité de 1.

— Si l'oracle est l'identité, cette fois, le passage à travers l'oracle correspond à une porte CNOT sur la paire de qbits. La paire de qbits se trouve dans l'état $\left(\begin{pmatrix}\frac{1}{\sqrt{2}}\\ \frac{-1}{\sqrt{2}}\end{pmatrix}\otimes\begin{pmatrix}\frac{1}{\sqrt{2}}\\ \frac{-1}{\sqrt{2}}\end{pmatrix}\right)=\begin{pmatrix}\frac{1}{2}\\ \frac{-1}{2}\\ \frac{-1}{2}\\ \frac{1}{2}\end{pmatrix}$.

Après passage dans la porte CNOT, on obtient

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

Le passage de chacun des qbits dans une porte de Hadamard met le qbit d'entrée dans l'état $|0\rangle$ et le deuxième dans l'état $|1\rangle$.

— De manière analogue, on peut traiter le cas d'un oracle qui inverse les valeurs des qbits.

Ainsi, en faisant un seul appel à l'oracle quantique, on est capable de donner une information sur cet oracle qu'en informatique quantique nous n'aurions pas pu avoir sans faire au moins deux appels à l'oracle.

Ce problème peut se généraliser à la question de savoir si une fonction à valeurs booléennes est équilibrée, comme décrit dans l'algorithme de Deutsch-Josza (Deutsch and Jozsa, 1992).

4 Conclusion

Dans ce court article, on n'a pu qu'effleurer les principes de l'informatique quantique. Néanmoins, sur un problème simple, on a pu voir l'étrangeté du domaine puisqu'on trouve un moyen de résoudre un problème bien plus rapidement qu'en informatique classique.

Historiquement, le problème présenté ici a constitué un moment majeur dans le développement de l'informatique quantique, car, bien que le problème n'ait aucun intérêt en pratique, il a ouvert la voie à des algorithmes quantiques performants comme les algorithmes de recherche de type Grover ou le fameux algorithme de Shor, susceptible de permettre de casser le système de chiffrement RSA à plus ou moins long terme.

5 Compléments sur les espaces hermitiens

Nous donnons ici quelques compléments niveau première et deuxième année de licence sur les espaces hermitiens.

On rappelle que si $z = a + ib$ est un nombre complexe de partie réelle a et de partie imaginaire b, on appelle conjugué de z et on note $\overline{z}$ le nombre complexe $a - ib$.

5.1 Produits scalaires hermitiens

Soit $n \in \mathbb{N}$, un espace vectoriel E sur le corps des nombres complexes est appelé un espace hermitien s'il est muni d'une application $\langle\cdot|\cdot\rangle$ qui vérifie les propriétés suivantes :

1. elle est semi linéaire à gauche, c'est à dire que pour tous $x, y, z \in E, \lambda \in \mathbb{C}$, $\langle \lambda x + y | z \rangle = \overline{\lambda} \langle x | z \rangle + \langle y | z \rangle$,
2. elle est linéaire à droite, c'est à dire que pour tous $x, y, z \in E, \lambda \in \mathbb{C}$, $\langle z | \lambda x + y \rangle = \lambda \langle z | x \rangle + \langle z | y \rangle$,
3. elle est à symétrie hermitienne, c'est à dire que $\langle x | y \rangle = \overline{\langle y | x \rangle}$,
4. elle est définie, c'est à dire que $\langle x | x \rangle = 0$ si et seulement si $x = 0$,
5. elle est positive, c'est à dire que $\langle x | x \rangle \geq 0$ quel que soit $x \in E$.

5.2 Structure hermitienne de $\mathbb{C}^n$

L'exemple le plus utile pour comprendre l'informatique quantique est l'espace hermitien $\mathbb{C}^n$ muni de sa structure vectorielle canonique et du produit scalaire hermitien :

$$\forall x, y \in \mathbb{C}^n, \langle x | y \rangle = \sum_{i=1}^{n} \overline{x}_i y_i.$$

En notation matricielle, cela peut s'écrire, en posant $x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$ et $y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}$,

$$\langle x, y \rangle = {}^t\overline{x} y,$$

où $\overline{x}$ est le vecteur obtenu en prenant le conjugué de chaque composante de x.

Cette interprétation vectorielle du produit scalaire permet de justifier la notation de Dirac "bra" et "ket" (les symboles $\langle \cdot | \cdot \rangle$ se lisent "brackets" en anglais).

En effet, en physique quantique, il est usuel de noter un vecteur colonne x avec la notation ket x, que l'on note $|x\rangle$. La notation $\langle x|$ désigne le transposé du vecteur conjugué de x, autrement dit ${}^t\overline{x}$.

Ainsi, en physique, on peut utiliser la notation $\langle x | y \rangle = \langle x | y \rangle$.

5.3 Groupe des matrices unitaires

Comme dans le cadre des espaces euclidiens, un ensemble de matrices a un statut particulier dans les espaces hermitiens, c'est l'ensemble des matrices unitaires.

Mathématiquement, on dit qu'une matrice est unitaire si ${}^t\overline{U}U = I$, autrement dit si son inverse est la transposée de sa matrice conjuguée.

Donnons ici un certain nombre de propriétés des matrices unitaires :

— une matrice unitaire est inversible,

— les colonnes des matrices unitaires forment une base orthonormée de E pour le produit hermitien,

— leur déterminant vaut 1,

— elles préservent la norment des vecteurs auxquels on les applique.

Références

Aaronson, S. (2013). *Quantum computing since Democritus*. Cambridge University Press.

Aho, A., Ullman, J., and Cazin, X. (1996). *Concepts fondamentaux de l'informatique*. Sciences sup. Dunod.

Deutsch, D. (1985). Quantum theory, the church–turing principle and the universal quantum computer. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, 400(1818) :97–117.

Deutsch, D. and Jozsa, R. (1992). Rapid Solution of Problems by Quantum Computation. *Proceedings of the Royal Society of London Series A*, 439(1907) :553–558.

Nielsen, M. A. and Chuang, I. (2002). *Quantum computation and quantum information*. Cambridge University Press.