Algorithmes quantiques

Voici ci-dessous des exemples d'algorithmes quantiques ainsi que les preuves correspondantes. Vous trouverez plus de détails sur les algorithmes quantiques existants se trouvent dans le livre de référence [4].

1 Algorithme de Grover

1.1 Notion d'oracle

L'algorithme de Grover [2] porte sur la recherche d'élément satisfaisant une propriété donnée dans un ensemble non-structuré. Il permet une accélération polynomiale par rapport à son équivalent classique. Même si l'accélération n'est pas aussi spectaculaire que pour d'autres algorithmes quantiques (Schor, QFT) qui montrent la supériorité quantique, il peut s'appliquer à un grand nombre de problèmes.
Il s'agit ici de chercher un élément spécifique dans un ensemble de 2^n éléments (indexé donc par des chaînes binaires sur n bits) et nous avons une fonction test $f : \{0,1\}^n \to \{0,1\}$ telle que $f(x) = 1$ si x est la solution et $f(x) = 0$ sinon.
Pour cela on utilise un opérateur unitaire O appelé *oracle* qui va encoder la fonction f pour reconnaitre la solution:

$$\begin{aligned} O : \mathbb{C}^{2^n} \otimes \mathbb{C}^2 &\to \mathbb{C}^{2^n} \otimes \mathbb{C}^2 \\ |x\rangle\,|y\rangle &\mapsto |x\rangle\,|f(x) \oplus y\rangle\,, \end{aligned}$$

où $|x\rangle$ est le registre *index* et $|y\rangle$ est un qubit auxiliaire appelé *qubit oracle*. On remarque que $|y\rangle$ est "inversé" si $f(x) = 1$ (i.e $0 \to 1$ et $1 \to 0$) et reste inchangé si $f(x) = 0$. On rappelle que $|x\rangle\,|y\rangle$ est une façon abrégée de noter $|x\rangle \otimes |y\rangle$.
Si on applique O à l'état $|x\rangle\,(|0\rangle - |1\rangle)/\sqrt{2}$, alors $|0\rangle$ and $|1\rangle$ sont interchangés si x est la solution et restent inchangés sinon. On a donc

$$O : |x\rangle\,(|0\rangle - |1\rangle)/\sqrt{2} \mapsto (-1)^{f(x)}\,|x\rangle\,(|0\rangle - |1\rangle)/\sqrt{2}.$$

1.2 Etapes de l'algorithme

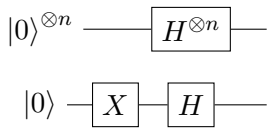
1) Initialisation: On prépare $|x\rangle$ dans l'état $|0\rangle^{\otimes n}$ et on applique une transformation d'Hadamard pour obtenir l'état $\frac{1}{\sqrt{2^n}} \sum_{s\in\{0,1\}^n} |s\rangle$ (mise en superposition).
Le qubit auxiliaire $|y\rangle$ (oracle) est initialisé à $H\,|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$.
Après initialisation, on a donc l'état

$$\begin{aligned} |\psi\rangle &= H^{\otimes n}\,|0\rangle^{\otimes n} \otimes H\,|1\rangle \\ &= \left(\frac{1}{\sqrt{2^n}} \sum_{s\in\{0,1\}^n} |s\rangle\right) \otimes \frac{(|0\rangle - |1\rangle)}{\sqrt{2}} \\ &= \left(\sum_{s\in\{0,1\}^n} \alpha_s\,|s\rangle\right) \otimes \frac{(|0\rangle - |1\rangle)}{\sqrt{2}}, \text{ avec } \alpha_s = \frac{1}{\sqrt{2^n}}. \end{aligned}$$

On remarque que les coefficients de $|\psi\rangle$ sont réels. Ils le resteront tout au long de l'algorithme. On a également, puisque $H\,|1\rangle = HX\,|0\rangle$,

$$|\psi\rangle = H^{\otimes n}\,|0\rangle^{\otimes n} \otimes HX\,|0\rangle = H^{\otimes n+1}(|0\rangle^{\otimes n} \otimes X\,|0\rangle) = H^{\otimes n+1}(I^{\otimes n} \otimes X)\,|0\rangle^{\otimes n+1}\,.$$

Le circuit permettant l'initialisation de l'algorithme de Grover est donc:



2) Changement de signe. On applique O à $|\psi\rangle$:

$$\begin{aligned} O|\psi\rangle &= O\left(\left(\sum_{s\in\{0,1\}^n}\alpha_s|s\rangle\right)\otimes\frac{(|0\rangle-|1\rangle)}{\sqrt{2}}\right) \\ &= \left(\sum_{s\in\{0,1\}^n}(-1)^{f(s)}\alpha_s|s\rangle\right)\otimes\frac{(|0\rangle-|1\rangle)}{\sqrt{2}} \\ &= \left(-\alpha_t|t\rangle+\sum_{s\neq t}\alpha_s|s\rangle\right)\otimes\frac{(|0\rangle-|1\rangle)}{\sqrt{2}}, \text{ si } t \text{ est la solution.} \end{aligned}$$

On remarque que le dernier qubit (oracle) reste inchangé et que le changement de signe n'affecte que les n premiers qubits. L'état que l'on obtient en appliquant O à $|\psi\rangle$ est ainsi le même que $|\psi\rangle$ sauf que le signe de $|t\rangle\otimes\frac{(|0\rangle-|1\rangle)}{\sqrt{2}}$ a changé.

3) Inversion par rapport à la moyenne. Il s'agit de l'opération:

$$\sum_{s\in\{0,1\}^n}\alpha_s|s\rangle\longrightarrow\sum_{s\in\{0,1\}^n}\left(2\left(\sum_{k\in\{0,1\}^n}\frac{\alpha_k}{2^n}\right)-\alpha_s\right)|s\rangle\,.$$

$\sum_{k\in\{0,1\}^n}\frac{\alpha_k}{2^n}$ étant la moyenne des coefficients, on soustrait chaque coefficient à 2 fois cette moyenne. On peut montrer qu'il s'agit d'une opération unitaire. Comme le coefficient α_t de l'état solution $|t\rangle$ est négatif, le soustraire à 2 fois la moyenne a pour effet de rendre le nouveau coefficient beaucoup plus grand que les autres et donc de mesurer $|t\rangle$ avec une plus grande probabilité.

Déroulement de l'algorithme: On répète les étapes 2) et 3) décrites ci-dessus pour augmenter α_t et à la fin mesurer $|t\rangle$ avec une probabilité proche de 1.
On montre que le nombre optimal d'itérations est $l\simeq\frac{\pi}{4}\sqrt{N}$ (avec $N=2^n$) i.e $\mathcal{O}(\sqrt{N})$, au lieu de $\mathcal{O}(N)$ pour l'algorithme classique, d'où une accélération (speed-up) quadratique.
A noter que si on effectue plus de l itérations, la probabilité décroît.

2 Transformée de Fourier quantique (QFT)

2.1 Expression de la QFT

On rappelle que la transformée de Fourier discrète classique est:

$$x_0, \ldots, x_{N-1} \mapsto y_0, \ldots, y_{N-1} \text{ avec } y_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j e^{2i\pi jk/N}.$$

Soit $\{|j\rangle\}_{j=0,N-1}$ une base orthonormée de $\mathbb{C}^N$ ($N = 2^n$), la transformée de Fourier quantique (Quantum Fourier Transform - QFT) est définie par:

$$QFT: \quad |j\rangle \quad \longmapsto \quad \tfrac{1}{\sqrt{N}} \textstyle\sum_{k=0}^{N-1} e^{2i\pi jk/N} |k\rangle \ . \tag{1}$$

Pour un état arbitraire $|x\rangle = \sum_{j=0}^{N-1} x_j |j\rangle$, on a donc:

$$\begin{aligned} QFT(|x\rangle) &= \sum_{j=0}^{N-1} x_j \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2i\pi jk/N} |k\rangle \\ &= \sum_{k=0}^{N-1} \underbrace{\left(\frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j e^{2i\pi jk/N} \right)}_{y_k} |k\rangle \,, \end{aligned}$$

où les y_k sont les transformées de Fourier discrètes des x_j.

Proposition 1. *La QFT est une transformation unitaire.*

Démonstration: On exprime tout d'abord la QFT sous forme matricielle. A partir de l'égalité précédente, on obtient

$$\begin{aligned} QFT(|x\rangle) &= \frac{1}{\sqrt{N}} \sum_j \sum_k e^{2i\pi jk/N} |k\rangle \, x_j \\ &= \frac{1}{\sqrt{N}} \sum_j \sum_k e^{2i\pi jk/N} |k\rangle \langle j|x\rangle \\ &= \underbrace{\left(\frac{1}{\sqrt{N}} \sum_j \sum_k e^{2i\pi jk/N} |k\rangle \langle j| \right)}_{U_{QFT}} |x\rangle \,. \end{aligned}$$

Montrons ensuite que U_{QFT} est une matrice unitaire, i.e que $U_{QFT} U_{QFT}^* = I_N$.

On a $U_{QFT}^* = \frac{1}{\sqrt{N}} \sum_j \sum_k e^{-2i\pi jk/N} |j\rangle \langle k|$ car $(|k\rangle \langle j|)^* = \langle j|^* |k\rangle^* = |j\rangle \langle k|$. D'où

$$\begin{aligned} U_{QFT} U_{QFT}^* &= \frac{1}{N} \sum_{j,k,l,m} e^{2i\pi(jk-lm)/N} |k\rangle \underbrace{(\langle j| \, |l\rangle)}_{1 \text{ si } j=l,\ 0 \text{ sinon}} \langle m| \\ &= \frac{1}{N} \sum_{j,k,m} e^{2i\pi j(k-m)/N} |k\rangle \langle m| \,, \text{ car } l = j \\ &= \frac{1}{N} \sum_k \sum_m \left(\sum_j e^{2i\pi j(k-m)/N} \right) |k\rangle \langle m| \,. \end{aligned}$$

Si $k=m$, alors $e^{2i\pi j(k-m)/N}=1$ et $\sum_{j=0}^{N-1} e^{2i\pi j(k-m)/N}=N$

Si $k\neq m$, alors $\sum_{j=0}^{N-1} e^{2i\pi j(k-m)/N}=\sum_{j=0}^{N-1}\left(e^{2i\pi(k-m)/N}\right)^j=\frac{1-\left(e^{2i\pi(k-m)/N}\right)^N}{1-e^{2i\pi(k-m)/N}}=0$. D'où

$$\begin{aligned} U_{QFT}U_{QFT}^* &= \frac{1}{N}\sum_{k=0}^{N-1} N\,|k\rangle\langle k| \ \text{(car } k=m\text{)} \\ &= \sum_{k=0}^{N-1}|k\rangle\langle k| \\ &= I_N \ \text{(completness relation)}. \end{aligned}$$

2.2 Représentation produit de la QFT

Dans la suite nous utilisons les notations suivantes:

- $j_1j_2\dots j_n$ est la représentation binaire de $j_1 2^{n-1}+j_2 2^{n-2}+\dots+j_n 2^0$.
- $0.j_l j_{l+1}\dots j_m$ est la représentation en fraction binaire de $\frac{j_l}{2^1}+\frac{j_{l+1}}{2^2}\cdots+\frac{j_m}{2^{m-l+1}}$.

Proposition 2. *La QFT peut s'exprimer en utilisant la représentation produit suivante*

$$|j_1\dots j_n\rangle \mapsto \frac{(|0\rangle+e^{2i\pi 0.j_n}|1\rangle)(|0\rangle+e^{2i\pi 0.j_{n-1}j_n}|1\rangle)\cdots(|0\rangle+e^{2i\pi 0.j_1j_2\dots j_n}|1\rangle)}{\sqrt{2^n}}.$$

Démonstration: Soit $|j\rangle$ un état de base de $\mathbb{C}^{2^n}$. En appliquant l'expression (1) et en écrivant $k=\underbrace{k_1\dots k_n}_{\text{chaine binaire}}=k_1 2^{n-1}+\dots+k_n 2^0$ et $\frac{k}{2^n}=k_1 2^{-1}+\dots+k_n 2^{-n}$, on obtient

$$\begin{aligned} QFT(|j\rangle) &= \frac{1}{\sqrt{2^n}}\sum_{k=0}^{2^n-1} e^{2i\pi jk/2^n}|k\rangle \\ &= \frac{1}{\sqrt{2^n}}\sum_{k_1=0}^{1}\cdots\sum_{k_n=0}^{1}\underbrace{e^{2i\pi j\sum_{l=1}^n k_l 2^{-l}}}_{\prod_{l=1}^n e^{2i\pi jk_l2^{-l}}}\underbrace{|k_1\dots k_n\rangle}_{|k_1\rangle\otimes\cdots\otimes|k_n\rangle} \\ &= \frac{1}{\sqrt{2^n}}\sum_{k_1=0}^{1}\cdots\sum_{k_n=0}^{1}\bigotimes_{l=1}^{n}\left(e^{2i\pi jk_l2^{-l}}|k_l\rangle\right) \\ &= \frac{1}{\sqrt{2^n}}\sum_{k_1\dots k_n\in\{0,1\}^n} e^{2i\pi jk_1 2^{-1}}|k_1\rangle\otimes\cdots\otimes e^{2i\pi jk_n2^{-n}}|k_n\rangle \\ &= \frac{1}{\sqrt{2^n}}\bigotimes_{l=1}^{n}\sum_{k_l=0}^{1} e^{2i\pi jk_l2^{-l}}|k_l\rangle \ \text{(distributivite)} \\ &= \frac{1}{\sqrt{2^n}}\bigotimes_{l=1}^{n}\left(|0\rangle+e^{2i\pi j2^{-l}}|1\rangle\right). \end{aligned}$$

Si $j=j_1j_2\dots j_n$ alors $j2^{-l}=j_1 2^{n-l-1}+\dots+j_n2^{-l}$ et donc

$$\begin{aligned} \text{pour } l=1: \quad j2^{-l} &= j_1 2^{n-2}+\dots+j_{n-1}2^0+j_n2^{-1}=0.j_n+m_1 \text{ avec } m_1\in\mathbb{N} \\ \text{pour } l=1: \quad j2^{-l} &= j_1 2^{n-3}+\dots+j_{n-1}2^{-1}+j_n2^{-1}=0.j_{n-1}j_n+m_2 \text{ avec } m_2\in\mathbb{N} \\ &\vdots \\ \text{pour } l=n: \quad j2^{-l} &= j_1 2^{-1}+\dots+j_n2^{-n}=0.j_1\dots j_n. \end{aligned}$$

D'où:

$$\begin{aligned} QFT(|j\rangle) &= \frac{1}{\sqrt{2^n}}(|0\rangle + e^{2i\pi(m_1+0.j_n)}|1\rangle)(|0\rangle + e^{2i\pi(m_2+0.j_{n-1}j_n)}|1\rangle)\cdots(|0\rangle + e^{2i\pi 0.j_1\ldots j_n}|1\rangle) \\ &= \frac{1}{\sqrt{2^n}}(|0\rangle + e^{2i\pi 0.j_n}|1\rangle)(|0\rangle + e^{2i\pi 0.j_{n-1}j_n}|1\rangle)\cdots(|0\rangle + e^{2i\pi 0.j_1\ldots j_n}|1\rangle). \end{aligned}$$

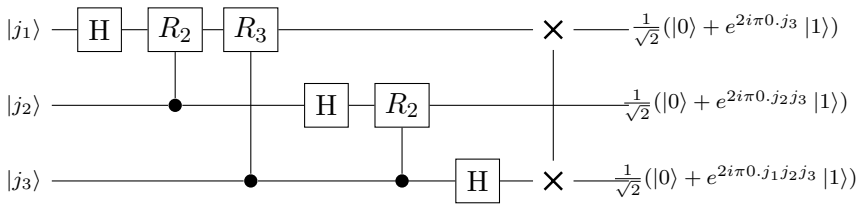
2.3 Circuit/algorithme pour la QFT

Le principe de l'algorithme consiste à calculer successivement les facteurs

$$\frac{1}{\sqrt{2}}(|0\rangle + e^{2i\pi 0.j_1 j_2\ldots j_n}|1\rangle), \frac{1}{\sqrt{2}}(|0\rangle + e^{2i\pi 0.j_2\ldots j_n}|1\rangle), \ldots$$

en utilisant des portes H (pour mettre initialement chaque qubit en superposition) et des rotations controlled-R_k avec $R_k = \begin{pmatrix} 1 & 0 \\ 0 & e^{2i\pi/2^k} \end{pmatrix}$. A la fin, des portes Swap inversent l'ordre des qubits.

Explication: Si on applique H au premier qubit j_1 on a $H|j_1\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{2i\pi 0.j_1}|1\rangle)$.
En effet, $0.j_1 = \frac{j_1}{2}$ donc si $j_1 = 0$, $H|j_1\rangle = |+\rangle$ et si $j_1 = 1$, $H|j_1\rangle = |-\rangle$.
On applique ensuite une porte R_2 à j_1, controlée par j_2:
$R_2 H|j_1\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{2i\pi 0.j_1} \underbrace{e^{2i\pi\frac{j_2}{2^2}}}_{=1 \text{ si } j_2=0}|1\rangle) = \frac{1}{\sqrt{2}}(|0\rangle + e^{2i\pi 0.j_1 j_2}|1\rangle)$.
Puis on applique R_3 controlée par j_3, etc. A la fin on obtient $\frac{1}{\sqrt{2}}(|0\rangle + e^{2i\pi 0.j_1\ldots j_n}|1\rangle)$.
On procède de même pour j_2 auquel on applique H puis R_3 controlée par j_3, etc, pour obtenir $\frac{1}{\sqrt{2}}(|0\rangle + e^{2i\pi 0.j_2\ldots j_n}|1\rangle)$.
Voici ci-dessous le circuit correspondant pour 3 qubits. On rappelle que les rotations R_2 and R_3 correspondent respectivement aux portes S et T.



Circuit pour la QFT (3 qubits).

Remarques:

- La QFT nécessite $\mathcal{O}(n^2)$ portes, au lieu de $\mathcal{O}(n2^n)$ pour le meilleur algorithme classique.
- La QFT ne peut pas être utilisée pour des applications classiques telles que le traitement du signal car on ne peut pas accéder directement aux amplitudes via la phase de mesure mais c'est une routine clé pour certains algorithmes quantiques importants (estimation de phase, Schor algorithm,...).

References

[1] E. Farhi, J. Goldstone, and S. Gutmann. A quantum approximate optimization algorithm. *arXiv preprint arXiv:1411.4028*, 2014.

[2] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the 28th Annual ACM Symposium on Theory of Computing*, pages 212–219. ACM, 1996.

[3] Michael Lubasch, Jaewoo Joo, Pierre Moinier, Martin Kiffner, and Dieter Jaksch. Variational quantum algorithms for nonlinear problems. *Phys. Rev. A*, 101:010301, 2020.

[4] Michael A Nielsen and Isaac L Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2011.

[5] Jules Tilly, Hongxiang Chen, Shuxiang Cao, Dario Picozzi, Kanav Setia, Ying Li, Edward Grant, Leonard Wossnig, Ivan Rungger, George H. Booth, and Jonathan Tennyson. The variational quantum eigensolver: A review of methods and best practices. *Physics Reports*, 986:1–128, 2022.

[6] Xiaosi Xu, Jinzhao Sun, Suguru Endo, Ying Li, Simon C. Benjamin, and Xiao Yuan. Variational algorithms for linear algebra. *Science Bulletin*, 66(21):2181–2188, 2021.