Algorithmes quantiques.

David Deutsch

Peter Shor

Simulation du calcul d'une fonction par réseau de portes quantiques.

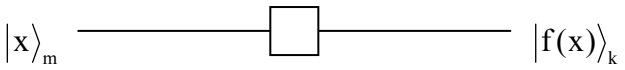
En agençant un nombre arbitraire de portes quantiques de toutes les manières possibles, on construit des réseaux qui transforment globalement n qubits d'entrée en n qubits de sortie. On peut naturellement concevoir la transformation résultante comme l'exercice du calcul d'une fonction, f(x) : $\{0,1\}^n \rightarrow \{0,1\}^n$. Toutefois c'est la question inverse et généralisée qui est la plus intéressante : peut-on toujours associer un réseau quantique à une fonction donnée, f(x) : $\{0,1\}^m \rightarrow \{0,1\}^k$? L'universalité au sens de Turing l'exige mais cela ne se fait pas sans précautions. L'ordinateur quantique ne considère que les portes unitaires donc invertibles ce qui exige que le nombres des bits d'entrée et de sortie coïncident (m=k) et encore cela ne suffit pas car même lorsque m=k, la majorité des portes restent non invertibles ainsi que le montre un simple argument de comptage.

Il existe 2^{k2^m} fonctions binaires, $\{0,1\}^m \rightarrow \{0,1\}^k$. Chacune de ces fonctions peut être vue comme une porte dont la table logique exprime les instances de la fonction. En particulier, il existe 2^{n2^n} fonctions binaires, $\{0,1\}^n \rightarrow \{0,1\}^n$, parmi lesquelles $2^n!$ seulement sont invertibles. Voici par exemple, dans le cas, n=2 :

une des 24 portes invertibles : $\begin{matrix} 0 & 0 \\ \rightarrow & \rightarrow \\ 0 & 0 \end{matrix} \quad \begin{matrix} 0 & 1 \\ \rightarrow & \rightarrow \\ 1 & 0 \end{matrix} \quad \begin{matrix} 1 & 1 \\ \rightarrow & \rightarrow \\ 0 & 1 \end{matrix} \quad \begin{matrix} 1 & 0 \\ \rightarrow & \rightarrow \\ 1 & 1 \end{matrix}$,

une des 232 (= 256-24) portes non invertibles : $\begin{matrix} 0 & 0 \\ \rightarrow & \rightarrow \\ 0 & 0 \end{matrix} \quad \begin{matrix} 0 & 1 \\ \rightarrow & \rightarrow \\ 1 & 0 \end{matrix} \quad \begin{matrix} 1 & 1 \\ \rightarrow & \rightarrow \\ 0 & 0 \end{matrix} \quad \begin{matrix} 1 & 0 \\ \rightarrow & \rightarrow \\ 1 & 1 \end{matrix}$.

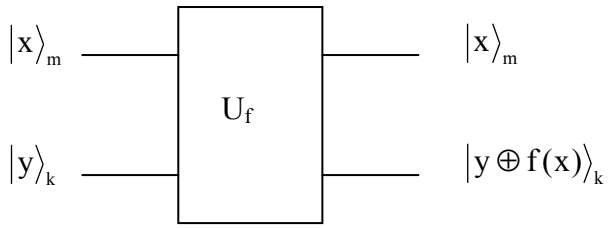
Il est, en général, inutile d'espérer construire un réseau quantique équivalent au calcul strict et rien de plus d'une fonction donnée, du type :



Si m≠k, c'est évident car un réseau quantique est toujours invertible et ce schéma ne l'est pas. Même lorsque m=k, nous savons qu'un calcul invertible exige qu'on ajoute aux données spécifiques du problème posé (les arguments de la fonction) un certain nombre de qubits de contrôle qui peuvent sans inconvénients être posés à zéro au début du calcul. A la fin du calcul, on récupère les résultats escomptés plus des qubits de déchet, inutiles en regard du problème posé mais indispensables pour garantir l'invertibilité du calcul. Le schéma précédent peut, par contre, toujours être remplacé par le suivant.

Quelle que soit la fonction classique, f(x) : $\{0,1\}^m \rightarrow \{0,1\}^k$, qui calcule k bits de sortie à partir de m bits d'entrée, il est possible de trouver une transformation unitaire (qui est d'ailleurs sa propre inverse), U_f, agissant sur les m qubits d'entrée plus k qubits supplémentaires, y, dits de contrôle tels qu'on retrouve intacts à la sortie les m qubits de

données flanqués de k qubits, y Xor f(x). En particulier, le calcul de f(x) s'obtient en posant les k qubits de y égaux à zéro.



On voit que le calcul quantique d'une fonction, f(x) : $\{0,1\}^m \rightarrow \{0,1\}^k$, ne se fait en toute certitude qu'à l'aide d'une porte de dimension, n=m+k, que l'on note :

$$U_f |x\rangle_m |y\rangle_k = |x\rangle_m |y \oplus f(x)\rangle_k .$$

Dans la base calculatoire, la représentation matricielle de U_f prend la forme d'une des $2^n!$ matrices de permutations. Illustrons ce qui vient d'être dit sur l'exemple, m=k=1.

Il existe 4 fonctions binaires, f(x) : $\{0,1\} \rightarrow \{0,1\}$, notées, f_0, f_1, f_2 et f_3. La table de leurs valeurs s'écrit :

x	$f_0(x)$	$f_1(x)$	$f_2(x)$	$f_3(x)$
0	0	0	1	1
1	0	1	0	1

La fonction, $f_2(x)$, par exemple, est telle que :

$$\begin{cases} B_{f_2}|0\rangle|y\rangle = |0\rangle|y \oplus 1\rangle \\ B_{f_2}|1\rangle|y\rangle = |1\rangle|y \oplus 0\rangle \end{cases}$$

et les autres suivent sur le même modèle. Les représentations matricielles des opérateurs, B_f, se notent, dans la base calculatoire habituelle :

$$B_{f_0} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad B_{f_1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad B_{f_2} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad B_{f_3} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

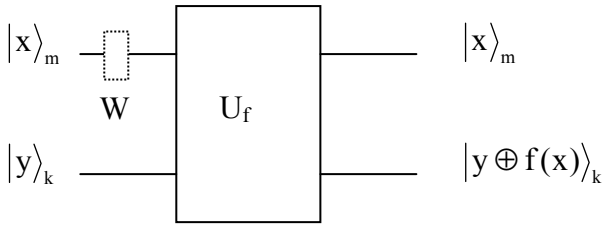
Calcul parallèle des valeurs d'une fonction.

La grande ambition de l'ordinateur quantique est d'être capable de traiter en parallèle un grand nombre, N, d'instances d'un problème donné. Dans l'exemple du calcul d'une fonction, f(x), il doit être capable d'évaluer en un seul passage l'application à tous les entiers binaires allant de 0 à 2^{m-1}. On y parvient comme suit.

On commence par préparer les données, $|x\rangle_m$, dans l'état de base particulier, $|000\cdots0\rangle$. Si on s'en tenait là, le réseau ne calculerait que f(0). Si on leur applique en sus l'opérateur de Walsh-Hadamard, $W = \bigotimes_{i=1}^{m} H$, elles entrent dans un état de superposition maximum :

$$W|000\cdots00\rangle = \frac{1}{\sqrt{2^m}} \sum_{i=0}^{2^m-1} |i\rangle,$$

qui peut être vu comme la superposition de tous les entiers binaires allant de 0 à 2^{m-1}. Le réseau de portes quantiques appliqué à ce nouvel état calculera, par linéarité, les 2^m instances de f(x), $\frac{1}{\sqrt{2^m}} \sum_{i=0}^{2^m-1} |i\rangle_m |f(i)\rangle$. Cette relation exprime le parallélisme quantique.



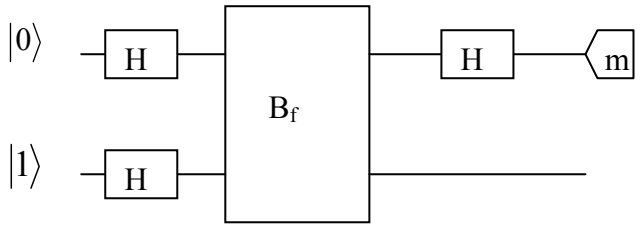
On pourrait se demander ce qu'on gagne concrètement du fait que pour prendre connaissance du résultat du calcul, une mesure effectuée sur les qubits de sortie est nécessaire qui ne révélera jamais qu'un seul des résultats calculés en parallèle et encore sans certitude a priori duquel il s'agira ! Cependant, on peut espérer montrer que soit ces probabilités d'occurrence ne sont pas égales et qu'elles favorisent à la longue certaines occurrences de solutions facilement vérifiables soit que quels que soient les tirages, certains invariants subsistent qui mènent à la solution cherchée.

On voit que la programmation quantique est un art très différent de son homologue classique. On ne connaît actuellement que fort peu d'algorithmes viables mais les recherches se poursuivent sur ce terrain neuf. Voici quelques exemples connus basés sur des principes d'action fort différents. Ils s'inspirent largement d'un exposé dû à John Watrous.

L'oracle de Deutsch.

Le problème apparenté suivant, encore dû à Deutsch, illustre la notion d'invariant. Un oracle est un système uniquement capable de répondre par oui ou part non à une question posée. Reconsidérons les 4 fonctions binaires, f(x) : $\{0,1\} \rightarrow \{0,1\}$, notées, f_0, f_1, f_2 et f_3. Deux, f_0 et f_3, sont dites constantes (f(0)=f(1)) et deux, f_1 et f_2, sont dites balancées (f(0)≠f(1)). Imaginons que le réseau quantique qui calcule une de ces quatre fonctions est effectivement prisonnier d'une boîte noire dont le contenu est inaccessible sauf qu'on peut lui soumettre deux qubits d'entrée et mesurer les deux qubits de sortie, une opération qu'on appellera un « passage ». Combien de passages sont-ils nécessaires pour découvrir si la fonction cachée est constante ou balancée ?

Le même problème posé en informatique classique exige deux passages qui soumettent successivement l'argument x='0' puis x='1'. L'ordinateur quantique fait mieux : un seul passage suffit avec un coût minime d'un qubit additionnel de contrôle. Voici le design du réseau.



On peut suivre l'évolution du registre initial, $|0\rangle \otimes |1\rangle$, à mesure que les différentes portes sont franchies :

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

$$\xrightarrow{B_f} \frac{1}{\sqrt{2}}\left((-1)^{f(0)}|0\rangle + (-1)^{f(1)}|1\rangle\right) \otimes \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \xrightarrow{H} (-1)^{f(0)}|f(0) \oplus f(1)\rangle \otimes \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$$

On constate que quelle que soit la fonction cachée, $f_j(x)$ (j = 0, 1, 2, 3), la mesure du premier qubit donne '0' si f est constante et '1' si elle est balancée. Le deuxième qubit est inutile et il n'a pas besoin d'être mesuré. Cet algorithme fonctionne donc sur base de l'existence d'un invariant, $f(0) \oplus f(1)$, commun aux solutions cherchées.

On peut rechercher la représentation matricielle de l'opérateur, R, qui condense à lui seul la totalité des portes du réseau et vérifier qu'elle a bien le comportement annoncé. Voici l'exemple, R_2, associé à la fonction f_2 (attention à l'ordre !):

$$R_2 = (H_A \otimes Id_B) \cdot B_{f_2} \cdot (H_A \otimes H_B) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & -1 \\ 1 & 0 & 0 & 1 \\ 0 & -1 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

$$R_2 \cdot (|0\rangle_A \otimes |1\rangle_B) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & -1 \\ 1 & 0 & 0 & 1 \\ 0 & -1 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 0 \\ 0 \\ -1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}_A \otimes \frac{1}{\sqrt{2}} \begin{pmatrix} -1 \\ 1 \end{pmatrix}_B$$

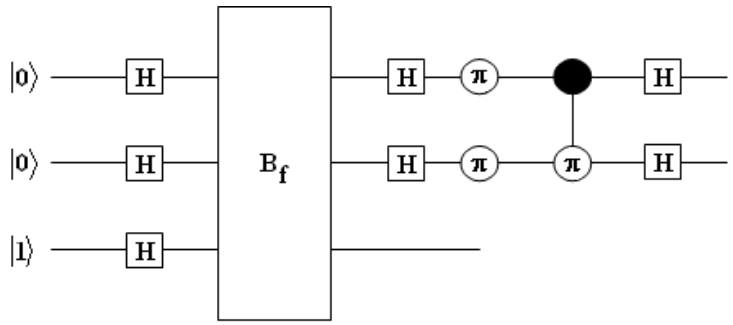
On voit que l'état final est factorisable et que la mesure du premier qubit le révèle obligatoirement dans l'état, $|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$, signe que la fonction, f_2, est balancée.

Algorithme d'identification.

Il existe 16 fonctions binaires, f(x) : $\{0,1\}^2 \rightarrow \{0,1\}$, ce sont les fonctions booléennes de base. Nous allons enfermer une de ces fonctions dans une boîte noire mais pour ne pas compliquer l'exposé, nous convenons de nous restreindre à quatre d'entre elles, précisément :

pq	$f_8 = Nor(p,q)$	pq	$f_4 = p < q$	pq	$f_2 = p > q$	pq	$f_1 = And(p,q)$
00	1	00	0	00	0	00	0
01	0	01	1	01	0	01	0
10	0	10	0	10	1	10	0
11	0	11	0	11	0	11	1

Le problème posé consiste à découvrir en un seul passage laquelle de ces quatre fonctions la boîte noire calcule. Cette performance est manifestement hors de portée d'un ordinateur classique. Le circuit suivant répond à la question posée.



On peut le vérifier en suivant pas à pas l'évolution du registre ou en calculant la représentation matricielle équivalente, nécessairement 8x8, ou encore en recourant à la notation tensorielle. Il s'avère que le troisième qubit est inutile et que la mesure des deux premiers livre la réponse cherchée selon le code :

$|00\rangle_{AB} \rightarrow f_8 \qquad |01\rangle_{AB} \rightarrow f_4 \qquad |10\rangle_{AB} \rightarrow f_2 \qquad |11\rangle_{AB} \rightarrow f_1 .$

Algorithme de recherche dans une base de données.

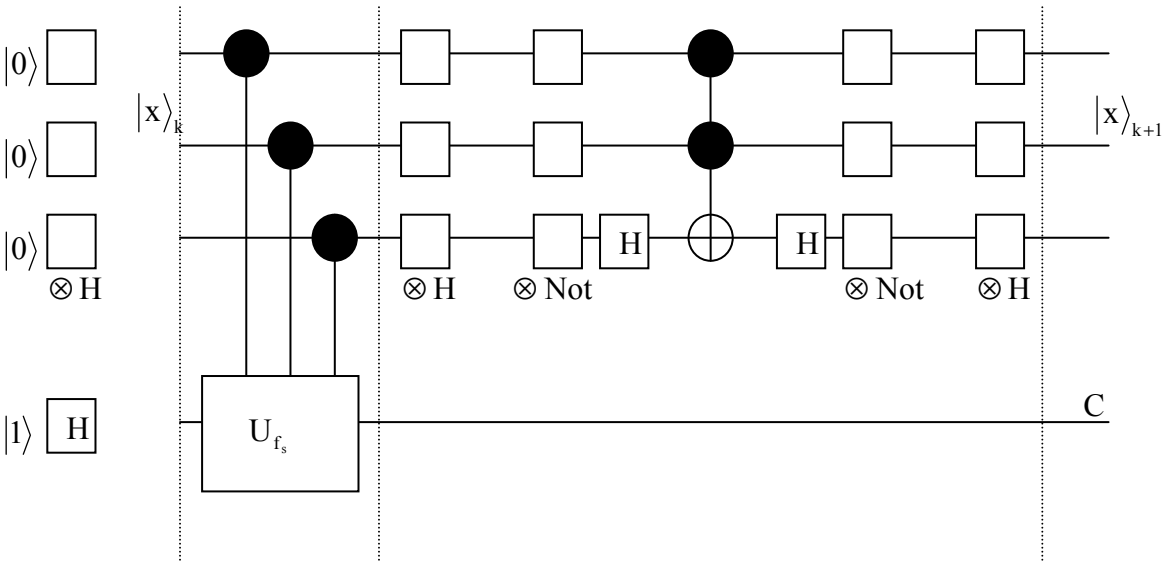
Les problèmes qui précèdent sont artificiels à plus d'un titre. D'une part, la question posée n'est pas particulièrement intéressante et d'autre part, personne ne passera jamais son temps à enfermer un système dans une boîte noire pour le plaisir de compliquer la situation. Le problème suivant est nettement plus réaliste. Rappelons qu'il existe $\boldsymbol{2^{2^n}}$ fonctions, $f(x):\boldsymbol{\{0,1\}^n \rightarrow \{0,1\}}$ parmi lesquelles 2^n sont est nulles pour toutes les valeurs de ses variables sauf une, disons x_a. Le problème est précisément de trouver x_a tel que $f(x_a)=1$. Vu que x_a est assimilable à une suite, s, de '0' et de '1', on voit que ce problème est apparenté à la recherche d'un abonné dans un annuaire classé dans l'ordre alphabétique quand on ne connaît que son numéro d'appel.

La fonction que nous avons en vue et son implémentation quantique se notent respectivement :

$$f_s(x)=\begin{cases} 0 & \text{si} \quad x \neq s \\ 1 & \text{si} \quad x=s \end{cases}$$

$$U_{f_s}|x\rangle|y\rangle = |x\rangle|y \oplus f_s(x)\rangle .$$

Le circuit suivant, imaginé par Grover résout par itération le problème posé :



Il se compose de n lignes (on n'en a dessiné que trois) qui encodent le vecteur d'état, $|x\rangle$, du système, à tout instant plus une ligne de contrôle, C. On prépare initialement le registre dans l'état de base, $|\boldsymbol{00\cdots 0}\rangle$, que l'on fait entrer dans l'état de superposition maximum grâce à n portes de Hadamard disposées en parallèle :

$$|\boldsymbol{x_0}\rangle = \bigotimes_{1}^{n} \boldsymbol{H}|\boldsymbol{00\cdots 0}\rangle = \frac{\boldsymbol{1}}{\boldsymbol{2^{n/2}}} \sum_{\boldsymbol{k=0}}^{\boldsymbol{2^n-1}} |\boldsymbol{k}\rangle .$$

Quant au qubit de contrôle, C, on l'initialise dans l'état, $|1\rangle$, qu'une porte de Hadamard transforme immédiatement en : $|1\rangle \to \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. En résumé, le système démarre dans l'état, $|x_0\rangle \otimes \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$.

Dans l'écriture du vecteur d'état, $|x_0\rangle$, les N états de base sont mis sur un pied d'égalité. La probabilité qu'une mesure du registre révèle, à ce stade, la valeur k cherchée ne vaut évidemment que $|\langle x_0 | s\rangle|^2 = 1/N$, soit la valeur prédite par un tirage au sort honnête. C'est cette situation que le réseau conçu par Grover se propose d'améliorer. Ce réseau fonctionne itérativement, transformant à chaque passage le vecteur d'état, $|x_k\rangle$ $(k = 0,1,\cdots)$, en un nouveau vecteur d'état, $|x_{k+1}\rangle$, qui se rapproche de $|s\rangle$.

Il est commode de décomposer à tout instant le vecteur d'état selon la direction définie par $|s\rangle$ et la résultante des composantes orthogonales qui s'aligne sur $|u\rangle$. Au départ, on a :

$$|x_0\rangle = \sqrt{\frac{N-1}{N}}|u\rangle + \frac{1}{\sqrt{N}}|s\rangle = \cos(\theta/2)|u\rangle + \sin(\theta/2)|s\rangle$$

et on cherche λ_k et μ_k tels que l'on a encore à tout instant ultérieur,

$$|x_k\rangle = \lambda_k|u\rangle + \mu_k|s\rangle.$$

Le bloc itératif se compose de deux unités distinctes que la figure a séparé par un trait pointillé. Lors de la k^{ième} itération, la première unité, qui est une porte U_f, a pour seul effet d'inverser le signe de la composante de $|x_k\rangle$ selon $|s\rangle$ (le qubit de contrôle n'est pas altéré) :

$$U_{f_s}|x_k\rangle \otimes \left|\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)\right\rangle = \frac{1}{\sqrt{2}}|x_k\rangle \otimes \left(|0 \oplus f_s(x_k)\rangle - |1 \oplus f_s(x_k)\rangle\right) = (-1)^{f_s(x)}|x_k\rangle \otimes \left|\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)\right\rangle$$
$$= \left(1 - 2|s\rangle\langle s|\right)|x_k\rangle \otimes \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = (\lambda_k|u\rangle - \mu_k|s\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$$

La deuxième unité est d'apparence plus complexe bien que l'effet global soit simple : elle correspond à l'opérateur, $2|x_0\rangle\langle x_0| - Id$, ce que l'on peut vérifier directement. Les calculs se résument comme suit :

$$|x_{k+1}\rangle = \lambda_{k+1}|u\rangle + \mu_{k+1}|s\rangle =$$
$$\left(2(\cos(\theta/2)|u\rangle + \sin(\theta/2)|s\rangle)(\cos(\theta/2)\langle u| + \sin(\theta/2)\langle s|) - Id\right) \cdot (\lambda_k|u\rangle - \mu_k|s\rangle)$$

d'où le système récurrent satisfait par λ_k et μ_k :

$$\lambda_{k+1} = \lambda_k \cos\theta - \mu_k \sin\theta \qquad \mu_{k+1} = \mu_k \cos\theta + \lambda_k \sin\theta$$
$$\lambda_0 = \cos(\theta/2) \qquad \mu_0 = \sin(\theta/2)$$

On trouve que le système complet se trouve, après k itérations, dans l'état :

$$|x_k\rangle = \cos[(k+1/2)\theta]|u\rangle + \sin[(k+1/2)\theta]|s\rangle$$

et le qubit de contrôle n'est toujours pas altéré.

On constate que lorsque k est l'entier le plus proche de ($\pi/2\theta$)-0.5, soit encore de l'ordre de ,

$$k \approx \frac{\pi}{4}\sqrt{N} - \frac{1}{2},$$

$|x_k\rangle$ se confond quasiment avec $|s\rangle$. On voit qu'en gros, $\sqrt{N} = 2^{n/2}$ passages sont nécessaires, un gain appréciable par rapport à l'algorithme classique qui en exigerait 2^n. Par exemple, si $N=10^6$, $\theta = 2\arcsin(1/1000)$, et la probabilité qu'une mesure du registre, à l'étape k=785, fournisse l'ensemble des qubits encodés par $|s\rangle$ vaut $|\langle x_k|s\rangle|^2 = 0.9999999584$.

Factorisation des entiers longs : algorithme de Shor.

On sait que la confiance que l'on porte à la méthode désormais classique de cryptographie RSA repose sur deux conjectures jamais démontrées : 1) que la brisure du code RSA est synonyme de factorisation et 2) que cette factorisation n'est pas possible par des procédures classiques en un temps polynomial.

Aucune méthode classique de factorisation, de la plus naïve (Erathostène, en $O(\sqrt{N})$) à la plus évoluée (Pollard-Strassen, en $O\{exp[(c\ell gN)^{1/3}(\ell g\ell gN)^{2/3}]\}$), ne résout le problème en un temps polynomial. Par contre, on sait, depuis 1994, qu'il existe un algorithme quantique, dû à Shor, qui est susceptible d'y parvenir à condition qu'un ordinateur quantique digne de ce nom voie jamais le jour. En 2001, le record est détenu par un groupe IBM qui a « réussi » à factoriser l'entier 15 mais ce n'est peut-être qu'un début. Il va donc de soi que cet algorithme ne menace pas immédiatement la cryptographie à clefs publiques à tel point que beaucoup pensent que la probabilité que l'ordinateur quantique devienne une réalité est bien moindre que celle d'une brisure du code RSA par une méthode différente de la factorisation. L'algorithme de Shor n'en vaut pas moins le détour.

La méthode de Shor déploie, en fait, une stratégie probabiliste qui lui assure de trouver un facteur premier de n'importe quel nombre composite avec une bonne probabilité. Toute tentative qui a échoué peut être recommencée jusqu'à ce qu'un facteur se dégage presque à coup sûr en un temps raisonnable. La méthode de Shor est un mélange de stratégies classique et quantique et il va de soi que les premières ne sont utilisées que lorsqu'elles sont effectives

en un temps polynomial. Nous commençons par l'exposé du principe de la méthode. Soit à trouver un facteur premier de l'entier N.

1) Tester la primalité de N. On connaît depuis 2002 un algorithme (AKS) effectif en un temps polynomial. Si N est premier le problème est résolu : il n'existe pas de facteur premier autre que lui-même.

2) Sinon, choisir un entier, a (1<a<N), au hasard. Calculer, par l'algorithme d'Euclide, le pgcd de a et de N. S'il est différent de 1 on a, par chance, trouvé un facteur premier de N et le problème est résolu. Sinon on poursuit comme suit.

3) On construit la suite, $s_k = Mod[a^k, N]$, inévitablement périodique de période, r, $s_{k+r} = s_k$.

4) Si r est impair ou si $Mod[a^{r/2}, N] = \pm 1$, la procédure est en échec et il y a lieu de la reprendre au point 2 sur base d'une nouvelle valeur de a.

5) Sinon, deux facteurs de N sont respectivement : $pgcd[a^{r/2} \pm 1, N]$.

Toutes ces étapes sont effectives en un temps polynomial sauf une : celle qui calcule la période, r, de la suite. Il est utile, à ce stade, de rappeler la méthode classique de détection de la période d'une suite. Elle est basée sur la transformée de Fourier discrète (TFD).

1) Extraction de la période d'une suite par transformée de Fourier discrète.

Soit une suite $s_k, \ (k = 0,1,\cdots,N-1)$, on définit ainsi sa TFD, $\tilde{s}_j, \ (j = 0,1,\cdots,N-1)$:

$$\tilde{s}_j = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} exp[2i\pi \frac{jk}{N}] s_k \qquad \Leftrightarrow \qquad s_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} exp[-2i\pi \frac{jk}{N}] \tilde{s}_j$$

Lorsque la suite s_k est réelle, sa TFD ne l'est en général pas à l'exception de son premier élément qui vaut toujours la moyenne arithmétique de ses éléments. Les autres éléments de la TFD sont reliés par la relation : $\tilde{s}_j = \tilde{s}^*_{N-j} \ (j = 1,\cdots,N-1)$. Cette propriété ne vaut plus si sa suite de départ est complexe.

Il existe un rapport étroit entre TFD et suites périodiques. On le met en évidence en considérant le prototype de la suite périodique, $s_k = exp[2i\pi k\nu]$, de fréquence, $\nu \in]0,1[$, ou, si l'on préfère, de période, $T = 1/\nu$ (>1). La TFD de cette suite vaut exactement :

$$s_k = exp[2i\pi k\nu] \qquad \Leftrightarrow \qquad \tilde{s}_j = \frac{1}{\sqrt{N}} \frac{sin(N\pi\nu)}{sin(\pi\nu + \pi j/N)} exp\left[i\pi[(N-1)\nu - \frac{j}{N}] \right].$$

Il existe un cas idéal où la suite $\tilde{s}_j$ est nulle partout sauf en un point : il suffit que Nν soit entier, $N\nu = \ell$, ou, ce qui revient au même, que la période de la suite, T, divise sa longueur, N. Dans ce cas, la TFD est nulle partout sauf au point, j, calculé comme suit :

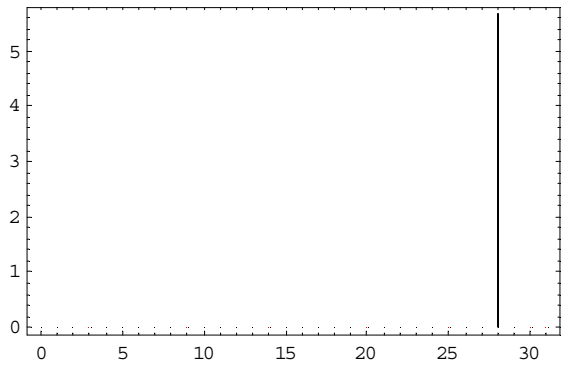
$$\sin(\pi\nu + \pi j / N) = 0 \quad \Rightarrow \quad j = N(1-\nu) \quad (\text{entier!}) \quad \Rightarrow \quad \tilde{s}_j = \sqrt{N} \neq 0 .$$

Voici le détail de la TFD de la suite de fréquence 1/8, échantillonnée 32 fois, suivie de son graphe qui est réel dans ce cas particulier :

```
Simplify[Table[1/Sqrt[32] Sum[Exp[2 i π/8 k] Exp[2 i π k j/32], {k, 0, 31}], {j, 0, 31}]]
```

```
{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 4 Sqrt[2], 0, 0, 0}
```

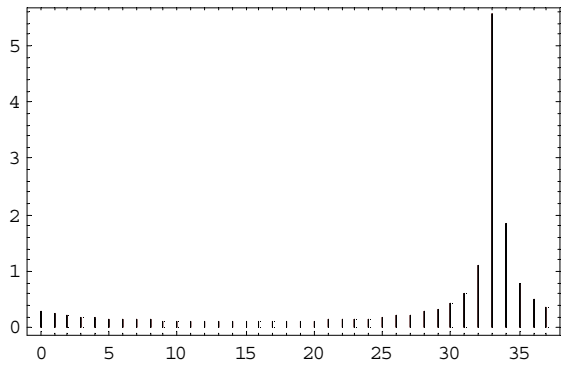
```
GeneralizedBarChart[Table[{j, Abs[Fourier[Table[Exp[2 i π/8 n], {n, 0, 31}]]][[j+1]], 0.0001}, {j, 0, 31}],
 PlotRange → All, Axes → False, Frame -> True]
```



($s_k = \exp[\frac{2i\pi}{8}k]$; ν = 1/8 d'où T = 8; N = 32)

Le graphe change lorsque N n'est plus un multiple de T : la suite, $\tilde{s}_j$, cesse d'être nulle presque partout. Toutefois, elle conserve un pic principal au voisinage de $N(1-\nu)$, qui a d'ailleurs cessé d'être un entier. Du fait que la TFD devient complexe on ne dessine que son module :

```
GeneralizedBarChart[Table[{j, Abs[Fourier[Table[Exp[2 i π/8 n], {n, 0, 37}]]][[j+1]], 0.0001}, {j, 0, 37}],
 PlotRange → All, Axes → False, Frame -> True]
```



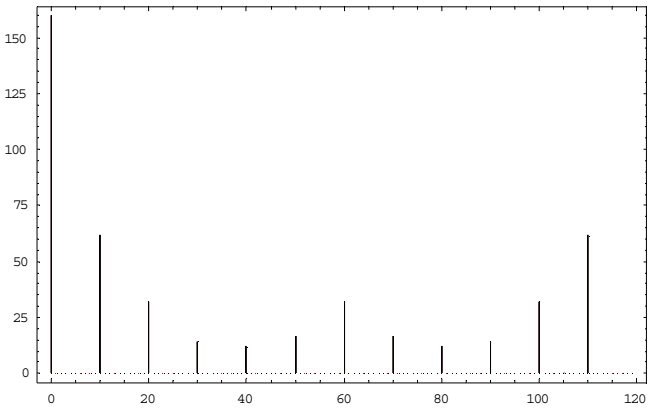
($s_k = \exp[\frac{2i\pi}{8}k]$; ν = 1/8 d'où T = 8; N = 38)

L'exemple considéré est très particulier. Une suite périodique quelconque possède une TFD plus compliquée. Considérons la suite, de période 12,

$$s_k = Mod[2^k, 35] = \{1,2,4,8,16,32,29,23,11,22,9,18,1,2,4,8,\cdots\}.$$

On constate, à nouveau, que si T divise N, la TFD continue d'être nulle sauf en quelques points isolés, onze dans l'exemple retenu en ignorant l'origine :

```
GeneralizedBarChart[Table[{j, Abs[Fourier[Table[Mod[2^n, 35], {n, 0, 119}]][[j + 1]], 0.0001}, {j, 0, 119}],
 PlotRange → All, Axes → False, Frame -> True]
```



($s_k = Mod[2^k, 35]$; T = 12 ; N = 120)

On explique cette démultiplication des pics en exprimant la suite comme combinaison linéaire de T exponentielles imaginaires du type, $exp(2i\pi \ell k \nu_0)$ $(\ell = 0,1,\cdots,T-1)$. Cette opération est toujours possible à condition d'égaler la valeur de ν_0 à l'inverse de la période de la suite considérée. On trouve un nombre de termes égal à la période, T, soit, dans l'exemple, la décomposition suivante :

$$s_k = Mod[2^k, 35] = \sum_{\ell=0}^{T-1} c_\ell \, exp[2i\pi \nu_0 \ell k] = c_0 + c_1 \, exp[\frac{2i\pi}{12} k] + c_2 \, exp[\frac{2i\pi}{12} 2k] + \cdots + c_{11} \, exp[\frac{2i\pi}{12} 11k]$$

Dans cette expression, le deuxième terme, $\ell = 1$, est toujours présent, c'est le fondamental de fréquence $\nu_0 = 1/T$. Les (T-2) termes qui suivent sont ses harmoniques, de fréquences, $\nu = \ell \nu_0 = \ell / T$ et il n'est pas exclu que certains harmoniques soient absents si la valeur du coefficient, c_j, qui lui est associée est nulle. Les c_j (j=0, …,11) se calculent simplement par inversion de la transformée de Fourier, soit, dans l'exemple :

```
c = Simplify[Table[1/12 Sum[Mod[2^j, 35] Exp[-2 i π/12 j k], {j, 0, 11}], {k, 0, 11}]]
```

$$\left\{\frac{175}{12}, -\frac{35}{24}\left((2-i)+\sqrt{3}\right), \frac{35}{12}(-1)^{1/3}, -\frac{7}{6}-\frac{7i}{12}, \frac{5}{24}\left(1+3i\sqrt{3}\right), \frac{35}{24}\left((-2+i)+\sqrt{3}\right),\right.$$
$$\left.-\frac{35}{12}, \frac{35}{24}\left((-2-i)+\sqrt{3}\right), \frac{5}{24}\left(1-3i\sqrt{3}\right), -\frac{7}{6}+\frac{7i}{12}, -\frac{35}{12}(-1)^{2/3}, -\frac{35}{24}(-1)^{1/6}\left((2-i)+\sqrt{3}\right)\right\}$$

En résumé, le terme constant est responsable de la contribution à l'origine et les autres termes contribuent chacun pour un pic dans la TFD, qu'on localise, en partant de la droite du graphe, aux positions, $j = N(1-\ell\nu_0)$ $(\ell = 1,2,\cdots,11)$. Les onze termes présentent, en effet, des fréquences, $\nu = \ell / T$, égales respectivement à : 1/12, 2/12, …, 11/12. On en

conclut que, dans l'exemple choisi, la TFD sera non nulle aux abscisses, 0, 10, 20, 30, …, 110. C'est bien ce qu'on observe.

En pratique on procède plutôt en sens inverse : on cherche la période de la suite sur base du graphe de sa TFD. On inverse donc le raisonnement et on associe à chaque pic situé en j une fréquence, ν, valant : $\boldsymbol{\nu = \ell \nu_0 = \frac{\ell}{T} = 1 - \frac{j}{N}}$.

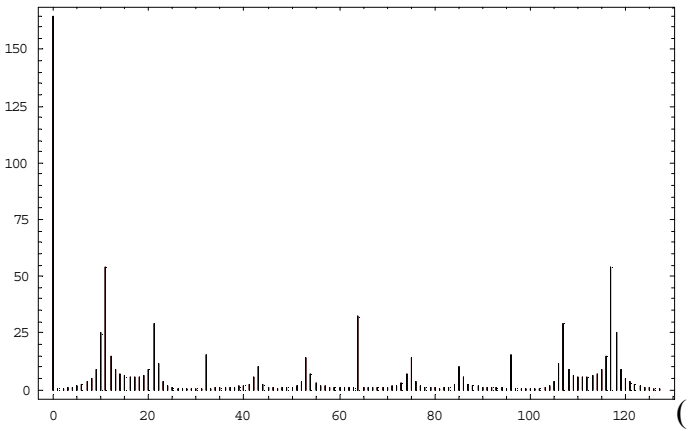
L'ensemble de ces fréquences forment la suite du fondamental et des harmoniques, soit dans l'exemple, {1/12, 2/12, 3/12, …, 11/12}, les pics étant lus de droite à gauche sur le graphe. La fréquence la plus basse, ν_0, correspond au dernier pic, obligatoirement toujours présent, et elle vaut l'inverse de la période cherchée, T=12, dans l'exemple.

Il résulte de ce qui précède que dans le cas particulier considéré où la longueur de la suite est un multiple de la période, la connaissance de la TFD d'une suite périodique renseigne immédiatement sur la valeur de sa période : il suffit de repérer la position, j, du dernier pic et d'appliquer la formule, $\boldsymbol{T = N/(N-j)}$. Si l'on applique la même formule en utilisant la valeur de j correspondant à un autre pic que le dernier, on trouve toujours un sous-multiple de la période.

Plusieurs problèmes subsistent cependant. Le premier est assez évident : on doit se fixer une longueur d'échantillonnage, N, pour la suite mais on ne connaît pas sa période, T, puisque précisément on la cherche. Sauf par chance extraordinaire, on se trouvera donc rarement dans le cas où T divise N et il convient de voir ce qu'on peut espérer de la méthode dans ce cas général.

Si T ne divise pas N, la TFD présente encore des pics principaux mais ils sont noyés dans un ensemble de pics ambiants plus petits d'où il résulte que le succès de l'analyse précédente n'est plus garanti. L'exemple suivant où la suite test est échantillonnée 128 fois le montre : le dernier pic principal se situe en j = 117. Or $\boldsymbol{T = N/(N-j)}$ = 128/(128-117) = 128/11 = 11.6364 ne livre certainement pas la période cherchée avec exactitude puisque ce n'est pas un entier mais elle n'en n'est pas très éloignée et une vérification de $\boldsymbol{s_T = s_0}$ est après tout toujours possible !

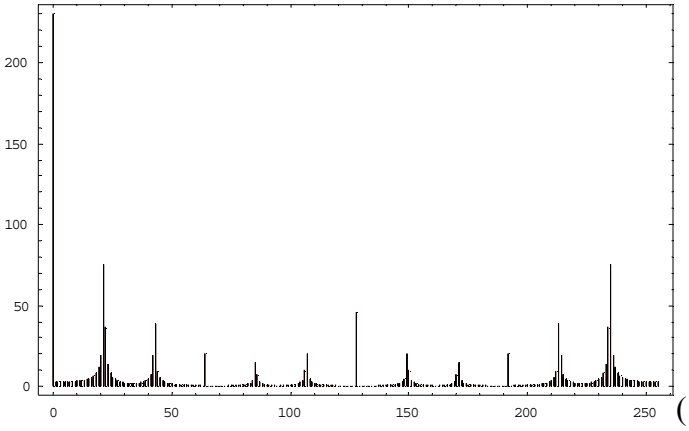
```
GeneralizedBarChart[Table[{j, Abs[Fourier[Table[Mod[2^n, 35], {n, 0, 127}]][[j + 1]], 0.00001}, {j, 0, 127}],
 PlotRange → All, Axes → False, Frame -> True]
```



(s_k = Mod[2^k,35] ; T = 12 ; N = 128)

On peut cependant remédier à cette situation de deux façons le cas échéant simultanément. La première consiste à allonger la suite. Voyons ce que devient l'exemple précédent si on double la longueur de l'échantillon, passant de N = 128 à 256 :

```
GeneralizedBarChart[Table[{j, Abs[Fourier[Table[Mod[2^n, 35], {n, 0, 255}]][[j + 1]], 0.0001}, {j, 0, 255}],
 PlotRange → All, Axes → False, Frame -> True]
```



(s_k = Mod[2^k,35] ; T = 12 ; N = 256)

On constate que les pics s'affinent et que la position du dernier d'entre eux, en j = 235, nous rapproche de la période cherchée : ***T = N/(N-j)*** = 256/(256-235) = 256/21 = 12.19. Si on localise par erreur le dernier pic principal en j = 234 ou 236, on trouvera une approximation de la période plus ou moins bonne, respectivement, 11.64 et 12.8. On voit que dans ce cas, la méthode cesse d'être sûre mais cela dit, il est toujours extrêmement facile de vérifier si on a bien, $s_T = s_0$, en essayant quelques valeurs qui encadrent l'approximation trouvée. Cette remarque peut paraître hors de propos dans la mesure où personne ne s'attend à commettre d'erreur dans le calcul d'une TFD. Elle ne l'est absolument pas dans l'optique d'une implémentation quantique de la TFD.

2) La transformée de Fourier quantique.

Une question préalable se pose toutefois : la procédure classique qui vient d'être décrite semble résoudre parfaitement le problème de l'extraction de la période d'une suite périodique. Dès lors pourquoi ne pas s'en contenter ? Le problème est que cette procédure n'est pas effective avec des ressources polynomiales. Le calcul complet d'une TFD sur un ordinateur classique requerrait un nombre d'opérations élémentaires de l'ordre du carré, N^2, du nombre que l'on veut factoriser (en fait, de l'ordre de $N\ell gN$ en recourant à l'algorithme rapide de Cooley & Tuckey) . A ce prix, autant recourir au crible d'Erathostène en $\sqrt{N}$!

Par contre l'ordinateur quantique fait beaucoup mieux car il est capable de calculer en parallèle tous les éléments d'une TFD. Considérons un registre, comprenant n qubits, qui évolue dans un espace de Hilbert à $N=2^n$ dimensions. Son vecteur d'état s'écrit :

$$|\psi\rangle = \sum_{j=0}^{2^n-1} c_j |j\rangle ,$$

dans la notation abrégée où les 2^n vecteurs de base, $|j\rangle$, sont ordonnés en suivant l'écriture binaire, de $|0\rangle = |00\cdots0\rangle$ à $|2^n-1\rangle = |11\cdots1\rangle$. Dans cette base, le vecteur d'état est complètement caractérisé par la suite des amplitudes $\{c_j\}$, de longueur, 2^n. On définit la transformée de Fourier quantique du vecteur d'état (TFQ), $|\tilde{\psi}\rangle$, comme le vecteur d'état,

$$|\tilde{\psi}\rangle = \sum_{k=0}^{2^n-1} \tilde{c}_k |k\rangle,$$

où la suite $\{\tilde{c}_k\}$ est la TFD, $\tilde{c}_k = \frac{1}{\sqrt{2^n}} \sum_{j=0}^{2^n-1} exp[2i\pi \frac{jk}{2^n}] c_j$, de la suite $\{c_j\}$.

Voici l'opérateur, $\hat{F}$, qui transforme $|\psi\rangle$ en $|\tilde{\psi}\rangle$, il est unitaire :

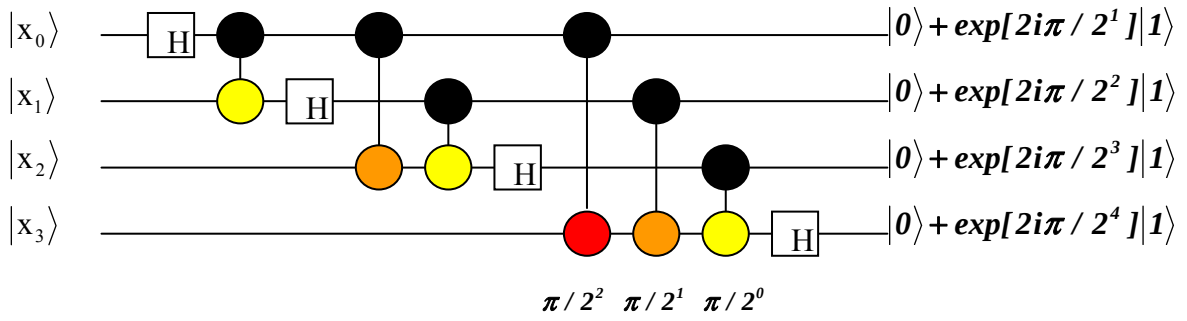
$$\hat{F} = \frac{1}{\sqrt{2^n}} \sum_{j=0}^{2^n-1} \sum_{k=0}^{2^n-1} exp[2i\pi \frac{jk}{2^n}] |k\rangle\langle j| \qquad \Rightarrow \qquad \hat{F}|\psi\rangle = |\tilde{\psi}\rangle.$$

Intéressons-nous au résultat de l'application de $\hat{F}$ à n'importe lequel des 2^n vecteurs de base, par exemple, $|\ell\rangle = |\ell_1 \ell_2 \cdots \ell_n\rangle$, on trouve que la TFQ se factorise comme suit :

$$\hat{F}|\ell\rangle =$$

$$|\tilde{\ell}\rangle = \frac{1}{\sqrt{2^n}} \left((|0\rangle + exp[2i\pi 0.\overline{\ell_n}|1\rangle) \otimes (|0\rangle + exp[2i\pi 0.\overline{\ell_{n-1}\ell_n}|1\rangle) \otimes \cdots \otimes (|0\rangle + exp[2i\pi 0.\overline{\ell_1\ell_2\cdots\ell_n}|1\rangle) \right)$$

Cette factorisation est essentielle pour une conception récursive du circuit quantique capable d'implémenter la QFT quel que soit le nombre, n, de qubits qui composent le registre. Le réseau quantique correspondant utilise n portes de Hadamard et n(n-1)/2 portes induisant sous contrôle des déphasages du type, $\pi/2^j$ (j=0,1,2,…), (n=4 dans l'exemple représenté) :



Dans ce schéma, la succession des portes s'écrit (attention à l'ordre inverse !) :

$$H_D \, c\Phi(\pi)_{CD} \, c\Phi(\pi/2)_{BD} \, c\Phi(\pi/4)_{AD} \, H_C \, c\Phi(\pi)_{BC} \, c\Phi(\pi/2)_{AC} \, H_B \, c\Phi(\pi)_{AB} \, H_A$$

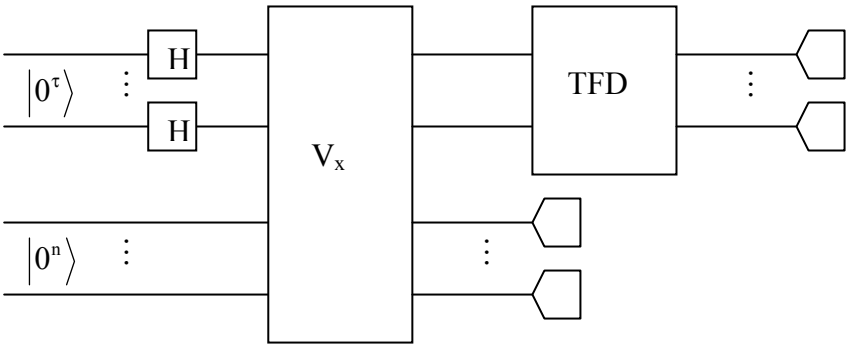
L'idée sur laquelle repose l'algorithme de Shor est de se servir de la TFQ pour calculer en un seul passage la totalité de la TFD. Toutefois ce calcul massivement parallèle a un prix, celui de ne révéler qu'une valeur particulière de la TFD lors de la lecture du registre de sortie

mais on verra que cela suffit à déterminer la période avec une bonne probabilité. Il est alors aisé de vérifier si $s_T = s_0$ et de recommencer la procédure si, par malchance, elle a échoué.

3) L'algorithme de Shor proprement dit.

Il existe plusieurs présentations plus ou moins économiques de l'algorithme mais nous avons préféré privilégier la clarté. Un exemple fera mieux comprendre comment il fonctionne. Soit à factoriser le nombre N=21. On choisit un entier, a<N, premier avec N, soit a=2, par exemple. La suite, $\{2^k \text{ Mod } 21\}$ vaut : {1,2,4,8,16,11,1,2,4,8,16,11} et elle est de période paire, r = 6. Deux facteurs premiers de 21 sont donc : PGCD$[2^{r/2} \pm 1,21]$ soit 3 et 7. Ce raisonnement a été facilité par le fait qu'une simple inspection de la suite a révélé la valeur de sa période mais cela cesse d'être possible lorsque N comporte quelques centaines de chiffres. Pour les besoins de l'exemple considéré, nous allons faire comme si r était effectivement hors d'atteinte immédiate.

Nous optons pour la présentation suivante, empruntée à Lavor et al., pas forcément la plus économique mais sans doute la plus claire. Nous avons besoin de deux registres. Le premier comprend τ qubits, τ, tel que $N^2 \le 2^\tau < 2N^2$: τ = 9 est la plus petite valeur qui convient dans l'exemple considéré. Le deuxième registre comprend n qubits où, $n = \lceil \ell gN \rceil$, représente le nombre de bits nécessaires à l'encodage classique de N. Les deux registres sont initialisés dans les états de base, $|0^\tau\rangle$ et $|0^n\rangle$, et le système est décrit par le vecteur d'état, $|\psi_0\rangle = |0^\tau\rangle|0^n\rangle$, soit, dans l'exemple, $|\psi_0\rangle = |000000000\rangle|00000\rangle$, que l'on abrège comme d'habitude en $|\psi_0\rangle = |0\rangle|0\rangle$.



τ portes de Hadamard commencent par transformer cet état en,

$$|\psi_1\rangle = \frac{1}{2^{\tau/2}} \sum_{j=0}^{2^\tau - 1} |j\rangle|0\rangle = \frac{1}{\sqrt{512}} \sum_{j=0}^{511} |j\rangle|0\rangle .$$

Ensuite, l'opérateur V_x entre en action qui effectue l'opération unitaire :

$$V_x|j\rangle|k\rangle = |j\rangle|(k + x^j) \mod N\rangle .$$

A la sortie de cette porte, le système se trouve dans l'état suivant :

$$|\psi_2\rangle = V_x|\psi_1\rangle = \frac{1}{2^{\tau/2}}\sum_{j=0}^{2^\tau-1}|j\rangle|x^j \mod N\rangle .$$

Dans l'exemple, cela donne :

$$\begin{aligned}|\psi_2\rangle = \frac{1}{\sqrt{512}}[&(|0\rangle+|6\rangle+|12\rangle+\cdots+|510\rangle)|1\rangle+(|1\rangle+|7\rangle+|13\rangle+\cdots+|511\rangle)|2\rangle+\\&(|2\rangle+|8\rangle+|14\rangle+\cdots+|506\rangle)|4\rangle+(|3\rangle+|9\rangle+|15\rangle+\cdots+|507\rangle)|8\rangle+\\&(|4\rangle+|10\rangle+|16\rangle+\cdots+|508\rangle)|16\rangle+(|5\rangle+|11\rangle+|17\rangle+\cdots+|509\rangle)|11\rangle]\end{aligned}$$

On note que les deux premiers termes contiennent 86 termes alors que les quatre derniers en contiennent 85. L'état, $|\psi_2\rangle$, a ceci de remarquable que grâce au phénomène de superposition quantique, il contient la totalité de l'information que constituent les diverses puissances de x modulo N. Evidemment cette information n'est pas directement accessible puisque toute mesure a pour effet de ne révéler qu'une seule de ces puissances en détruisant les autres par projection.

Le moment est venu de mesurer l'état du second registre qui, dans l'exemple, ne peut livrer que l'un des six résultats suivants, avec des probabilités d'ailleurs égales : {1,2,4,8,16,11}. Supposons qu'il s'agisse du résultat '2'. Le système est projeté sur l'état propre renormalisé correspondant :

$$|\psi_3\rangle = \frac{1}{\sqrt{86}}((|1\rangle+|7\rangle+|13\rangle+\cdots+|511\rangle)|2\rangle) = \frac{1}{\sqrt{86}}\sum_{\ell=0}^{85}|6\ell+1\rangle|2\rangle .$$

Le premier registre est à présent projeté dans une superposition d'états dont les numéros d'ordre forment une suite périodique de période précisément égale à r. Peu importe le détail des termes de la suite, r est le renseignement vital qu'il nous faut extraire. On y parvient en appliquant une TFD portant sur τ qubits :

$$|\psi_4\rangle = TFD|\psi_3\rangle = \frac{1}{\sqrt{2^\tau}}\sum_{j=0}^{2^\tau-1}\sum_{k=0}^{2^\tau-1}exp[2i\pi\frac{jk}{2^\tau}]|k\rangle|j\rangle|\psi_3\rangle ,$$

soit dans l'exemple :

$$|\psi_4\rangle = TFD|\psi_3\rangle = \frac{1}{\sqrt{512}}\frac{1}{\sqrt{86}}\sum_{j=0}^{512}\left[\left(\sum_{k=0}^{85}exp[2i\pi\frac{(6\ell+1)j}{512}]\right)|j\rangle\right]|2\rangle$$

Une mesure du premier registre, révèle la réponse, 'j', avec la probabilité,

$$p_j = \frac{1}{86\times512}\left|\sum_{\ell=0}^{85}exp[2i\pi\frac{(6\ell+1)j}{512}\right|^2 = \frac{1}{86\times512}\frac{\sin^2(258\pi j/256)}{\sin^2(3\pi j/256)} \qquad (j=0,1,\cdots,511) .$$

Dans cette dernière expression, une indétermination doit être levée en j=0 et 256, elle donne :

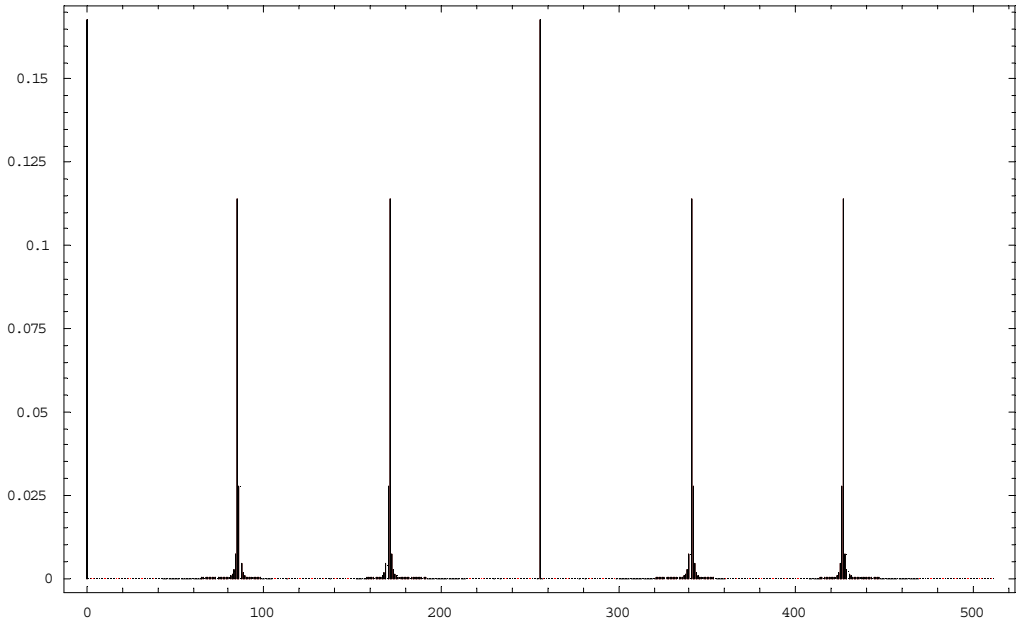
$$p_0 = p_{256} = \frac{86^2}{86 \times 512}$$

On vérifie que l'on a bien que la somme des probabilités vaut 1 : $\sum_{j=0}^{511} p_j = 1$.

Le graphe de p_j est celui d'une fonction qui ne s'écarte notablement de zéro qu'en cinq endroits bien définis (l'origine étant ignorée), aux voisinages respectifs de :

j=85	j=171	j=256	j=341	j=427
(p_{85}=11.4)	(p_{171}=11.4)	(p_{256}=16.8)	(p_{341}=11.4)	(p_{427}=11.4)

Ces cinq valeurs de j représentent à elles seules 62.4% des cas possibles.



Partout ailleurs (sauf en l'origine qui est inexploitable et qui est « tirée au sort par la mesure dans 16.8% des cas), la probabilité tombe rapidement en-dessous de 0.001. Cela signifie que la mesure du premier registre révélera le plus fréquemment une des 5 valeurs intéressantes, 85, 171, 256, 341 ou 427. Il reste à appliquer à ces 5 valeurs de j, prises dans l'ordre inverse, la formule bien connue, (pour rappel, N=512) :

$$\nu = \frac{\text{N-j}}{\text{N}} \qquad \Rightarrow \qquad \nu_0 = \frac{85}{512}; \nu_1 = \frac{171}{512}; \nu_2 = \frac{256}{512}; \nu_3 = \frac{341}{512}; \nu_4 = \frac{427}{512};$$

La période cherchée vaut l'inverse de la fréquence la plus basse, $1/\nu_0$= 512/85=6.023, soit plus que probablement la valeur cherchée, T = r = 6.

En pratique l'extraction de la période ou d'un de ses sous-multiples si la mesure a révélé autre chose que le dernier pic, se fait sur base du développement en fractions continues de la valeur trouvée pour la fréquence. Le développement de ν_4 donne :

$$\nu_4 = \frac{427}{512} = \frac{k}{r} = \cfrac{1}{1+\cfrac{1}{5+\cfrac{1}{42+\cfrac{1}{2}}}}$$

soit la suite des approximants, 1/1, **5/6**, 211/253,427/512.

On trouve la période cherchée, r, comme le dénominateur de l'approximant le plus précis dont le dénominateur n'excède toutefois pas N (en effet, r<21). On trouve le période vaut très probablement 6.

Le même calcul effectué sur ν_1 donne :

$$\nu_1 = \frac{171}{512} = \frac{k}{r} = \cfrac{1}{2+\cfrac{1}{1+\cfrac{1}{170}}}$$

soit la suite des approximants, 1/2, **1/3**, 171/512. Le meilleur dénominateur est 3 qui n'est pas la période mais un de ses sous-multiples. Cette technique d'extraction est tout à fait remarquable car on montre qu'elle continue de fonctionner même si l'on s'écarte modérément du pic principal. Par exemple, il suffit que la mesure révèle une valeur de j comprise entre 163 et 178 pour que le développement en fraction continue fournisse toujours 3 comme facteur probable de la période ainsi qu'en attestent les listes des approximants pour des valeurs de j allant de 162 à 179 :

```
Table[Convergents[ContinuedFraction[j/512]],{j,162,179}]
```

$$\left\{\left\{0, \frac{1}{3}, \frac{6}{19}, \frac{25}{79}, \frac{81}{256}\right\}, \left\{0, \frac{1}{3}, \frac{7}{22}, \frac{78}{245}, \frac{163}{512}\right\}, \left\{0, \frac{1}{3}, \frac{8}{25}, \frac{41}{128}\right\},\right.$$
$$\left\{0, \frac{1}{3}, \frac{9}{28}, \frac{10}{31}, \frac{29}{90}, \frac{68}{211}, \frac{165}{512}\right\}, \left\{0, \frac{1}{3}, \frac{11}{34}, \frac{12}{37}, \frac{83}{256}\right\}, \left\{0, \frac{1}{3}, \frac{15}{46}, \frac{76}{233}, \frac{167}{512}\right\},$$
$$\left\{0, \frac{1}{3}, \frac{21}{64}\right\}, \left\{0, \frac{1}{3}, \frac{33}{100}, \frac{34}{103}, \frac{169}{512}\right\}, \left\{0, \frac{1}{3}, \frac{85}{256}\right\}, \left\{0, \frac{1}{2}, \frac{1}{3}, \frac{171}{512}\right\}, \left\{0, \frac{1}{2}, \frac{1}{3}, \frac{43}{128}\right\},$$
$$\left\{0, \frac{1}{2}, \frac{1}{3}, \frac{24}{71}, \frac{25}{74}, \frac{74}{219}, \frac{173}{512}\right\}, \left\{0, \frac{1}{2}, \frac{1}{3}, \frac{17}{50}, \frac{35}{103}, \frac{87}{256}\right\}, \left\{0, \frac{1}{2}, \frac{1}{3}, \frac{13}{38}, \frac{27}{79}, \frac{175}{512}\right\},$$
$$\left\{0, \frac{1}{2}, \frac{1}{3}, \frac{11}{32}\right\}, \left\{0, \frac{1}{2}, \frac{1}{3}, \frac{9}{26}, \frac{28}{81}, \frac{177}{512}\right\}, \left\{0, \frac{1}{2}, \frac{1}{3}, \frac{8}{23}, \frac{89}{256}\right\}, \left\{0, \frac{1}{2}, \frac{1}{3}, \frac{7}{20}, \frac{43}{123}, \frac{179}{512}\right\}\right\}$$

La procédure échoue en j=162 ou 179 car l'algorithme suggérerait une période erronée valant 19 ou 20. Toutefois ces événements malencontreux sont extrêmement peu probables, p_{162} = 0.000126 ou p_{179} = 0.0002245. En recommençant toute la procédure un nombre suffisant de fois, on fait apparaître les facteurs probables de r. Le ppcm de ces facteurs permet de remonter à r avec une probabilité proche de 1. L'incertitude qui subsiste est facile à dissiper : il suffit de vérifier que la période convient et que l'on a bien $s_T = s_0$.

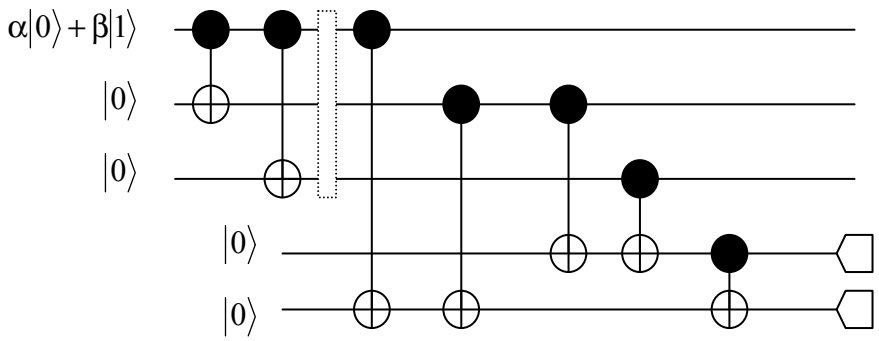
Corrections d'erreurs.

Les énormes difficultés qu'il y a à préserver tout registre quantique des influences décohérentes du milieu environnant représentent l'entrave majeure à la réalisation d'un prototype d'ordinateur quantique. Le problème est si aigu que la solution n'est nullement écartée de l'adoption d'un protocole massif de corrections d'erreurs. En d'autres termes, plutôt que d'essayer de maintenir à tout prix la cohérence des registres, on envisage de mettre en place, à tous les stades du calcul, des transformation qui passent les registres en revue et les restaurent automatiquement s'ils sont corrompus. Bien entendu de telles procédures consomment des ressources de qubits supplémentaires mais on estime généralement que ce coût reste modéré par rapport à ce que représenterait la mise en place d'une cohérence parfaite.

La correction automatique d'erreurs se fait sur le même principe qu'en théorie classique de l'information en recourant à des (qu)bits supplémentaires qui créent la redondance nécessaire. Toutefois le problème se complique en théorie quantique du fait que les vecteurs d'états ne sont pas forcément dans l'état logique '0' ou '1' mais qu'ils peuvent être en superposition des deux. Il faut, en particulier, pouvoir corriger des erreurs de déphasages sans perdre de vue que le clonage pur et simple est impossible. Nous ne faisons qu'effleurer ce sujet vaste et complexe.

Rappelons le principe du code correcteur classique le plus élémentaire qui soit : il est question d'envoyer un bit au travers d'un canal de transmission bruité de telle manière que la probabilité qu'il soit malencontreusement inversé soit égale à p. Le correspondant ne reçoit le bit correct qu'avec la probabilité (1-p). Une stratégie élémentaire consiste à envoyer trois copies du même bit que le receveur décodera à la majorité simple. Si $p<1/2$, on constate que la probabilité de décodage erroné tombe de p à $3p^2-2p^3$. Ce n'est pas parfait mais il y a un progrès. Si l'on veut faire beaucoup mieux, il y a lieu de mettre des procédures plus compliquées.

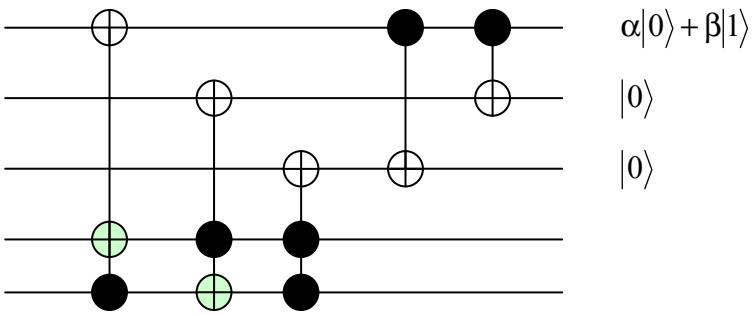
Cette procédure n'est pas applicable telle qu'elle à la transmission d'un qubit lorsque celui-ci est en état de superposition inconnu, $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$. D'une part, le théorème de non-clonage interdit de réaliser les deux copies nécessaires de $|\psi\rangle$ et de plus, mesurer les trois qubits envoyés afin de décoder à la majorité n'aurait aucun sens puisque la mesure a pour effet de détruire l'état mesuré. En fait, dans le montage qui suit, quatre qubits supplémentaires, préparés dans l'état de base, $|0\rangle$, sont utilisés à l'émission :



Supposons qu'on veuille protéger un qubit inconnu, $\alpha|0\rangle + \beta|1\rangle$, contre l'inversion accidentelle, '0' $\leftrightarrow$ '1'. On commence par utiliser deux portes CNot afin de l'encoder dans un registre à trois qubits, sous la forme, $|\psi_0\rangle = \alpha|000\rangle + \beta|111\rangle$. Rappelons que sur ce genre de figure, l'axe horizontal, lu de gauche à droite, représente la ligne du temps. Le rectangle pointillé représente une zone temporelle où une corruption d'inversion menace de se produire avec une probabilité que nous n'avons pas besoin de connaître. A la sortie de cette zone, le registre se trouvera dans l'un des quatre états suivants dont trois sont altérés :

$$Id \otimes Id \otimes Id|\psi_0\rangle \qquad Not \otimes Id \otimes Id|\psi_0\rangle \qquad Id \otimes Not \otimes Id|\psi_0\rangle \qquad Id \otimes Id \otimes Not|\psi_0\rangle .$$

Plutôt que d'essayer d'empêcher cette corruption, on préfère l'accepter mais on ajoute un circuit, composé de cinq portes CNot correctement agencées, qui ont pour effet de restaurer le registre dans les quatre cas de figure. Avec un peu de patience on peut se convaincre que le circuit fonctionne comme suit : les trois qubits parviennent à Bob dans l'état éventuellement altéré, $|\psi_0\rangle = \alpha|000\rangle + \beta|111\rangle$. Les deux derniers qubits ne sont que des auxiliaires qui peuvent être mesurés par Bob. Celui-ci trouvera, dans tous les cas, le numéro binaire du qubit altéré (00 = pas d'altération, 01 = altération du premier qubit, 10 = altération du deuxième qubit, 11 = altération du troisième qubit). Selon la valeur du syndrome trouvé, il reste à Bob à réappliquer une porte Not au qubit altéré puis à inverser la manœuvre réalisée par Alice afin d'extraire $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ de $|\psi_0\rangle = \alpha|000\rangle + \beta|111\rangle$. Ces deux manœuvres peuvent être faites automatiquement sans même qu'une mesure soit effectuée. Il suffit que Bob utilise le circuit de réception suivant, où les trois premières portes CNot corrigent l'erreur (le qubit de contrôle gris est inversé : le contrôle ne commande l'instruction Not que s'il est mis à '0') et les deux dernières portes extraient $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ de $|\psi_0\rangle = \alpha|000\rangle + \beta|111\rangle$:



Au bilan ce réseau répare l'éventuelle inversion isolée d'un qubit. Hélas l'histoire ne s'arrête pas là. L'inversion d'un qubit n'est qu'un avatar parmi d'autres que peut subir le qubit. L'altération la plus générale revêt nécessairement la forme suivante

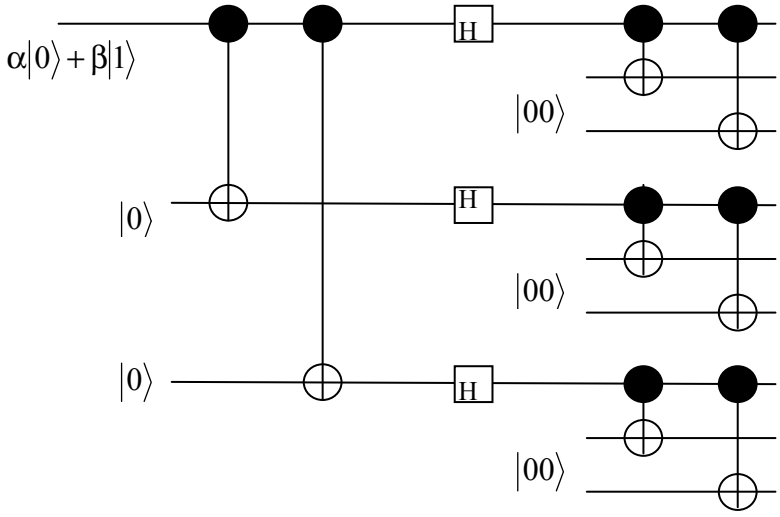
$$|\psi_{alt}\rangle = (e_1 Id + e_2\sigma_x + e_3\sigma_y + e_4\sigma_z)|\psi\rangle,$$

où les coefficients, e_i, peuvent être complexes. Dans l'exemple précédent, l'inversion correspondait au cas particulier, Not = σ_x. $\sigma_z = \Phi(\pi)$ correspondrait à une inversion de phase. Or les réseaux qui précèdent sont impuissants à corriger ne serait-ce qu'une inversion de phase et ce qu'il nous faut c'est une procédure effective dans tous les cas.

Il n'est pas question d'étudier ici les détails de cette procédure générale. On peut se faire une idée des complications auxquelles il faut s'attendre en contemplant la solution que Shor a trouvée pour régler le cas pas encore général où on veut corriger à la fois une inversion de bit et de phase. Une solution à ce problème nécessite un encodage préalable du qubit à protéger, $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, sous la forme d'un registre à 9 qubits :

$$|\psi_0\rangle = \alpha\frac{(|000\rangle+|111\rangle)(|000\rangle+|111\rangle)(|000\rangle+|111\rangle)}{2\sqrt{2}} + \beta\frac{(|000\rangle-|111\rangle)(|000\rangle-|111\rangle)(|000\rangle-|111\rangle)}{2\sqrt{2}}$$

que l'on obtient en utilisant le réseau suivant :



Ce réseau n'est que la première étape, celle de l'encodage redondant. Il faut encore installer le circuit correcteur puis celui de désencodage qui extrait $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ de $|\psi_0\rangle$ dans tous les cas d'une inversion de bit ou de phase. On voit que le réseau commence à prendre de l'ampleur exigeant un nombre croissant de qubits auxiliaires. La situation s'aggraverait évidemment dans le cas d'une altération supplémentaire par σ_y ou dans le cas d'une altération de plusieurs qubits. Toutefois, malgré le caractère inquiétant de ces complications, on peut montrer que le nombre de portes nécessaires ne croît que polynomialement ce qui permet de ne pas disqualifier la méthode.

Bob, Alice et les autres.

Les applications qui suivent impliquent, de près ou de loin, une communication à distance selon un canal de transmission. On ne s'étonnera pas que le photon soit considéré, dans ces cas, comme le système idéal d'encodage des qubits.

Nous poserons systématiquement le problème de la manière suivante : deux correspondants, traditionnellement appelés, Alice et Bob, désirent échanger de l'information. Alice est par convention l'émettrice et Bob le receveur. Nous admettrons que le canal de transmission n'est pas bruité ou, si ce n'est pas le cas, que toutes les précautions ont été prises en termes de protocoles de corrections d'erreurs.

La théorie prévoit dans certains cas qui privilégient la confidentialité l'intervention d'un troisième larron, baptisé Eve (!), qui a pour mission d'espionner passivement les canaux de transmission. L'espion actif, qui intercepte et falsifie la transmission dans un but malveillant se nomme habituellement Oscar : ses interventions sont nettement plus redoutables que celles d'Eve et elles nécessitent la mise en œuvre de protocoles évolués. Nettement plus fréquentable est Walter, qui est chargé de certifier le bon déroulement des transactions effectuées par Alice et Bob, par exemple, du type commerciales à distance.

Les applications qui impliquent une communication à distance sont nombreuses et nous n'en retiendrons que trois : la cryptographie quantique, le codage dense et la téléportation. Elles représentent les espoirs les plus sensés de l'informatique quantique de l'an 2000. Le fait est que la communication n'implique que la maîtrise au compte-gouttes de photons isolés, faciles à préparer et à mesurer. La technologie sous-jacente est nettement moins intimidante que celle qui déboucherait sur un ordinateur quantique.

Distribution quantique des clefs.

Nous avons vu par ailleurs qu'aucune méthode cryptographique classique n'est sûre. Toutes les méthodes imaginables aussi tordues soient-elles contiennent l'information cachée et sont de ce fait exposées à être dévoilées. La seule perspective qui s'offre aux encrypteurs est de compliquer la tâche d'espions éventuels afin d'allonger tellement démesurément le temps de décodage qu'il en devient prohibitif. La méthode RSA, dite à clefs publiques, ne voit sa sécurité garantie que par la lenteur de l'ordinateur classique et par l'acte de foi que le problème de la factorisation des entiers longs est incontournable et non polynomial.

Il nous faut quand même tempérer l'affirmation qu'aucune méthode cryptographique classique n'est sûre : il existe bien une méthode classique fiable à 100% mais elle exige d'utiliser une clef de (dé)chiffrement aléatoire aussi longue que le texte à encrypter et de ne l'utiliser qu'une seule fois ! Si cette possibilité paraît ridicule, c'est évidemment que cette clef doit être échangée entre les correspondants par un canal sûr et que dans ces conditions on a aussi vite fait d'utiliser ce canal sûr pour échanger le message lui-même ! Elle a pourtant été mise à l'essai en utilisant des porteurs de confiance, le texte crypté n'étant échangé que lorsqu'il était certain que la clef était parvenue, sans encombres, à destination. Ces essais n'ont pas survécu aux coûts de la manœuvre.

La théorie quantique de l'information rend cependant une nouvelle jeunesse à cette méthode. Bennett et Brassard ont en effet mis au point un protocole peu coûteux qui assure une transmission inviolable des clefs. Plus exactement les deux correspondants, Alice et Bob, ont la possibilité d'échanger une clef en ayant la certitude que si elle est interceptée par un espion, Eve, ils en seront informés. Ce n'est que lorsqu'ils ont la certitude de n'avoir pas été espionnés qu'il peuvent échanger en toute quiétude le message crypté sur un canal qui n'a même pas besoin d'être sûr.

La méthode exige deux canaux de communication entre Alice et Bob. Le premier canal achemine au compte-gouttes des photons, dans divers états de polarisation linéaire, le long d'une fibre optique par exemple. Le second est une voie de communication classique dont nous préciserons l'usage ultérieurement. Voyons d'abord comment les choses se passent dans le cas idéal où aucun espion n'est présent. Alice envoie à Bob des photons qu'elle

prépare individuellement et aléatoirement dans un des quatre états de polarisation linéaire, x, y ou à 45° dans les deux sens, soit schématiquement, {-, | }∈mode(+) et { /, \}∈mode(X). Alice associe, à sa convenance, les valeurs '0' et '1' à chaque état complémentaire et elle ne change jamais de convention. Par exemple, elle envoie les photons suivants, en respectant l'encodage, (- ou \ →'0' et | ou /→'1') :

- | / - \ / \ \ \ \ / - / / - - \ \ | / | - -	(suite aléatoire sur l'alphabet, - | / \)
0 1 1 0 0 1 0 0 0 0 1 0 1 1 0 0 0 0 1 1 1 0 0	(sa traduction en '0' et en '1')
+ ++ XX XX ++ + ++ X + + +X ++ X+ X X	(bases choisies aléatoirement par Bob)
- || / \ / \ -| - - - / | - - \ - | / | \ /	(résultats des mesures faites par Bob)
0 1 1 1 0 1 0 0 1 0 0 0 1 1 0 0 0 0 1 1 1 1 0	(leur traduction en '0' et '1')

Bob est parfaitement au courant de l'orientation des axes x et y (les deux correspondants peuvent se mettre d'accord lors d'un échange préparatoire d'information banale) ainsi que de l'encodage des bits adopté par Alice (en l'occurrence, - et \ pour encoder '0' et | et / pour encoder '1') mais il ignore totalement la suite (aléatoire) des orientations des polariseurs choisies par Alice lors de l'encodage. Il procède néanmoins à un décodage en variant lui aussi aléatoirement l'orientation de ses propres polariseurs selon les bases + ou X. Il va de soi qu'il se trompe d'orientation en moyenne une fois sur deux (les choix erronés ont été soulignés dans l'exemple cité).

Lorsque cette opération est terminée, les deux correspondants prennent contact par une voie téléphonique quelconque qui n'a pas besoin d'être sécurisée et dressent la liste des numéros d'ordre des photons pour lesquels ils ont fortuitement adopté la même orientation des polariseurs. Les bits restants, en gros la moitié des bits transmis, constituent une clef secrète potentiellement valable. Dans l'exemple, la clef qui subsisterait s'écrirait : 0 1 0 1 0 0 1 0 0 0 1 1 1. Toutefois il ne serait pas raisonnable de l'utiliser telle quelle. Les correspondants doivent, en effet, absolument contrôler l'absence d'intervention d'Eve. Pour ce faire, il existe une méthode élégante qui fait le prix de la cryptographie quantique : il suffit que Bob et Alice échangent par téléphone une partie de la clef obtenue de part et d'autre, par exemple les bits impairs ou multiples de 4 et qu'ils les jettent à la poubelle par mesure de précaution. Si ce sondage révèle une coïncidence parfaite c'est que le canal de transmission est resté inviolé. Par contre que faut-il penser du cas où Eve a espionné le canal de transmission ?

Posons-nous la question autrement : de quels moyens Eve dispose-t-elle pour espionner ce canal ? Au pis, elle pourrait : 1) être informée des orientations utilisées, + et X, 2) intercepter les photons envoyés par Alice et procéder à leur mesure enfin, 3) renvoyer les photons vers Bob dans l'état où sa mesure les a projeté. Par contre, elle ne connaît aucune des suites aléatoires des orientations des polariseurs choisies par Alice et par Bob. Elle en est donc réduite à se choisir sa propre suite mais cela va altérer gravement l'état des photons que Bob va recevoir. Alice et Bob vont immanquablement s'en apercevoir lors de leur sondage.

En effet, Eve va se tromper dans l'orientation des polariseurs en moyenne une fois sur deux. Les photons qu'elle va retransmettre à Bob vont provoquer un taux d'erreur sur la clef de 25% ce dont Alice et Bob vont inévitablement se rendre compte lors de la phase de

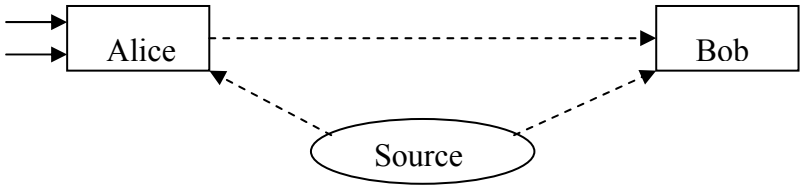
contrôle. S'il s'avère que la transmission a été interceptée, Alice et Bob interrompent de commun accord leur échange sinon Alice envoie le message crypté selon la clef définie.

Il existe des stratégies d'espionnage plus subtiles où Eve peut intriquer les photons interceptés en provenance d'Alice avec des photons ancillaires de sa fabrication avant de les renvoyer vers Bob dans leur forme altérée. Quelle que soit la méthode retenue par Eve, il lui est impossible de faire descendre le taux d'erreurs en-dessous de 15%, une valeur largement suffisante pour être détectée lors du sondage de contrôle. Entre les cas extrêmes où le taux d'erreur est nul (où l'échange du message codé peut se faire en toute sécurité) et celui où il est supérieur à 15% (où l'échange doit être interrompu), les choses se compliquent du fait qu'il est impossible de savoir si les erreurs constatées sont dues à un bruit dans le canal de transmission ou à un espion qui tenterait de se camoufler en n'interceptant qu'une partie des photons. Dans de tels cas il faut recourir à des techniques plus sophistiquées qui préservent la confidentialité. Ces techniques débordent d'un exposé élémentaire.

Il existe une multitude d'attaques variées possibles et de réponses adaptées qu'il est impossible de détailler dans un exposé élémentaire. Cela étant, la distribution quantique des clefs n'est plus une fiction : les premiers essais, entrepris à Genève dès 1995, banques obligent !, sont plus qu'encourageants. Plusieurs sociétés, américaines (NEC et NiCT) et japonaises (JST) ont commencé à développer un produit commercial et quelques banques s'intéressent à la publicité que représenterait la protection définitive des données sensibles.

Codage dense.

L'intrication permet à Alice de communiquer deux bits d'informations à Bob en ne lui envoyant qu'un seul photon, cela s'appelle le codage dense. Cette performance ne contredit en rien l'affirmation selon laquelle tout qubit ne peut révéler in fine qu'un seul bit d'information au sens classique du terme : le codage dense implique effectivement deux photons appartenant à une même paire EPR. Mais le canal qui relie Alice à Bob n'a à en acheminer qu'un seul. Voici le principe du protocole utilisé.



Une source EPR émet deux photons intriqués sous la forme,

$$|\psi_0\rangle = \frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right),$$

l'un vers Alice et l'autre vers Bob. Alice reçoit deux bits classiques autorisant l'encodage de 4 messages distincts, numérotés de 0 à 3, qu'elle veut pouvoir envoyer à Bob en n'envoyant qu'un seul photon. Selon le message à transmettre, elle soumet le photon reçu à l'une des quatre transformations unitaires suivantes et renvoie le résultat à Bob :

$$\mathbf{0} \qquad |\psi_0'\rangle = Id \otimes Id\,|\psi_0\rangle \quad \Rightarrow \quad |\psi_0'\rangle = \frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right)$$

$$\mathbf{1} \qquad |\psi_0'\rangle = X \otimes Id\,|\psi_0\rangle \quad \Rightarrow \quad |\psi_0'\rangle = \frac{1}{\sqrt{2}}\left(|10\rangle + |01\rangle\right)$$

$$\mathbf{2} \qquad |\psi_0'\rangle = Y \otimes Id\,|\psi_0\rangle \quad \Rightarrow \quad |\psi_0'\rangle = \frac{1}{\sqrt{2}}\left(-|10\rangle + |01\rangle\right)$$

$$\mathbf{3} \qquad |\psi_0'\rangle = Z \otimes Id\,|\psi_0\rangle \quad \Rightarrow \quad |\psi_0'\rangle = \frac{1}{\sqrt{2}}\left(|10\rangle - |11\rangle\right)$$

où les opérateurs réels, I, X, Y et Z sont apparentés aux matrices de Pauli :

$$Id = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \qquad X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \qquad Y = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \qquad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

Naturellement Alice n'a pu altérer que le premier qubit, celui qu'elle a reçu physiquement, l'autre est resté intact.

Lorsque Bob reçoit le photon réémis par Alice, il soumet la paire réunie dans son laboratoire à une porte CNot. Le calcul montre que Bob peut mesurer le second qubit sans altérer le premier, en effet :

$$\mathbf{0} \qquad |\psi_0''\rangle = cNot\,|\psi_0'\rangle \quad \Rightarrow \quad |\psi_0''\rangle = \frac{1}{\sqrt{2}}\left(|00\rangle + |10\rangle\right)$$

$$\mathbf{1} \qquad |\psi_0''\rangle = cNot\,|\psi_0'\rangle \quad \Rightarrow \quad |\psi_0''\rangle = \frac{1}{\sqrt{2}}\left(|11\rangle + |01\rangle\right)$$

$$\mathbf{2} \qquad |\psi_0''\rangle = cNot\,|\psi_0'\rangle \quad \Rightarrow \quad |\psi_0''\rangle = \frac{1}{\sqrt{2}}\left(-|11\rangle + |01\rangle\right)$$

$$\mathbf{3} \qquad |\psi_0''\rangle = cNot\,|\psi_0'\rangle \quad \Rightarrow \quad |\psi_0''\rangle = \frac{1}{\sqrt{2}}\left(|00\rangle - |10\rangle\right)$$

Si le second qubit est mesuré par Bob à la valeur '0' (resp. '1'), c'est qu'on est dans les cas 0 ou 3 (resp. 1 ou 2). Bob peut maintenant appliquer une porte de Hadamard au premier qubit et la mesure permet de trouver le message d'origine, en effet :

$$\mathbf{0} \qquad |\psi_0'''\rangle = H\frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right)|0\rangle \quad \Rightarrow \quad |\psi_0'''\rangle = |00\rangle$$

$$\mathbf{1} \qquad |\psi_0'''\rangle = H\frac{1}{\sqrt{2}}\left(|1\rangle + |0\rangle\right)|1\rangle \quad \Rightarrow \quad |\psi_0'''\rangle = |01\rangle$$

$$\mathbf{2} \qquad |\psi_0'''\rangle = H\frac{1}{\sqrt{2}}\left(-|1\rangle + |0\rangle\right)|1\rangle \quad \Rightarrow \quad |\psi_0'''\rangle = |11\rangle$$

$$\mathbf{3} \qquad |\psi_0'''\rangle = H\frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right)|0\rangle \quad \Rightarrow \quad |\psi_0'''\rangle = |01\rangle$$

Téléportation.

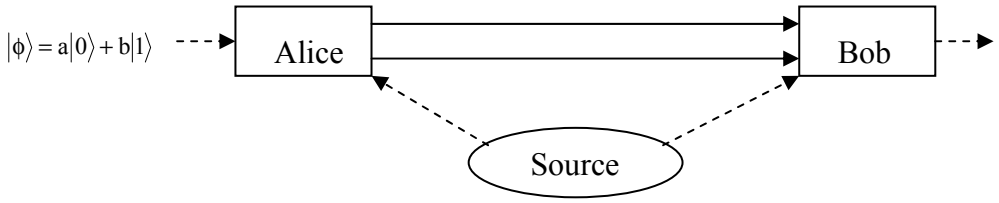
La téléportation est en un certain sens, que l'on va préciser, l'opération inverse du codage dense. Il s'agit de transmettre, par voie classique, l'information suffisante pour être en mesure de reconstruire de toute pièce et à distance un état quantique inconnu mais donné. Il va de soi que le prix à payer pour cette téléportation est la destruction de l'état quantique source sinon on aurait contrevenu au « no cloning theorem ».

Imaginons qu'Alice dispose d'un état, ϕ, qu'elle ne connaît pas mais qu'elle veut transmettre à Bob par un canal classique,

$$|\phi\rangle = a|0\rangle + b|1\rangle$$

Elle dispose pour ce faire d'un des photons d'une paire EPR dont l'autre est en possession de Bob. L'état de la paire est décrit par :

$$|\psi_0\rangle = \frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right).$$



Alice et Bob vont en fait inverser l'ordre des manoeuvres effectuées lors du codage dense : Alice commence par combiner les états , ϕ et ψ_0, sous la forme,

$$|\phi\rangle \otimes |\Psi_0\rangle = \frac{1}{\sqrt{2}}\left(a|000\rangle + a|011\rangle + b|100\rangle + b|111\rangle\right).$$

En se souvenant qu'elle ne contrôle que les deux premiers bits, elle soumet cet état aux transformations successives, cNot⊗Id puis H⊗Id⊗Id :

$$(H \otimes Id \otimes Id)(cNot \otimes Id)(|\phi\rangle \otimes |\Psi_0\rangle) =$$

$$\frac{1}{2}\left(|00\rangle(a|0\rangle + b|1\rangle) + |01\rangle(a|1\rangle + b|0\rangle) + |10\rangle(a|0\rangle - b|1\rangle) + |11\rangle(a|1\rangle - b|0\rangle)\right)$$

Ensuite elle mesure les deux premiers bits, trouvant, avec des probabilités égales, l'une des quatre possibilités, '00', '01', '10 ou '11' mais détruisant par là même l'état à transmettre. Toutefois l'information suffisante est sauvée qui va permettre à Bob de le reconstruire à distance. En effet celui-ci va recevoir par un canal classique le résultat de la mesure d'Alice, '01' par exemple. Or il connaît la table de reconversion des bits reçus :

‘00’ $\Rightarrow$ $|\phi\rangle = Id(a|0\rangle + b|1\rangle)$

‘01’ $\Rightarrow$ $|\phi\rangle = X(a|1\rangle - b|0\rangle)$

‘10’ $\Rightarrow$ $|\phi\rangle = Z(a|0\rangle - b|1\rangle)$

‘11’ $\Rightarrow$ $|\phi\rangle = Y(a|1\rangle - b|0\rangle)$

Il suffit à Bob d’appliquer la bonne transformation au photon qu’il a reçu de la source EPR pour qu’il le projette dans l’état demandé, $|\phi\rangle = a|0\rangle + b|1\rangle$, dans tous les cas.