

QUANTUM

UN PEU DE MATHÉMATIQUES
POUR L'INFORMATIQUE QUANTIQUE

ARNAUD BODIN

ALGORITHMES ET MATHÉMATIQUES



Un peu de mathématiques pour l'informatique quantique

Les ordinateurs quantiques sont parmi nous ! Enfin presque. . . Dans ce livre vous découvrirez l'informatique quantique et apprendrez à programmer sur un vrai ordinateur quantique. Même s'ils sont encore balbutiants et ne sont pas disponibles chez vous, vous avez accès en ligne à des machines quantiques pour tester de petits programmes.

La physique quantique est l'une des révolutions du vingtième siècle. Cela reste une matière difficile à étudier et encore plus à comprendre tant certains phénomènes quantiques contredisent notre perception du monde physique classique. Cependant la théorie quantique est validée par de nombreuses expériences et a des applications dans notre quotidien.

Depuis quelques années il existe des ordinateurs quantiques effectuant des calculs sur des « qubits ». Un qubit stocke l'information quantique : soit l'information 0, notée $|0\rangle$, soit l'information 1, notée $|1\rangle$, soit d'une certaine manière les deux en même temps ! Un qubit correspond à l'état d'une particule qui peut osciller entre un état au repos et un état excité.

C'est là qu'interviennent les mathématiques ! La physique quantique est difficile à comprendre et les ordinateurs quantiques sont compliqués à réaliser mais heureusement les mathématiques nécessaires pour s'initier à l'informatique quantique sont simples. Par exemple un qubit s'exprime en fait par l'expression mathématique :

$$\alpha |0\rangle + \beta |1\rangle .$$

C'est cette combinaison des deux états $|0\rangle$ et $|1\rangle$ qu'on vulgarise par la phrase mystérieuse « prendre à la fois la valeur 0 et la valeur 1 ». Il est cependant délicat de trouver un sens physique à cette superposition dans le monde classique et c'est encore plus ardu de maîtriser une particule qui réalise un qubit. Les mathématiques sont le langage idéal pour exprimer la physique et l'informatique quantique. Nous expliquons ici les notions (superposition, intrication, non-clonage quantique, . . .) comme des concepts mathématiques en se permettant de s'affranchir de l'univers physique délicat qui se cache derrière.

Ce livre offre une introduction douce à l'informatique quantique et aux mathématiques afin d'être en mesure de présenter l'algorithme de Shor. Cet algorithme a fait découvrir au monde la révolution que pourrait apporter un ordinateur quantique. Les communications sur internet sont pour la plupart sécurisées par un chiffrement qui repose sur la difficulté de factoriser de très grands entiers, même avec des ordinateurs très puissants. L'algorithme de Shor montre que sur un ordinateur quantique (plus gros que ceux qui existent actuellement) ce problème deviendrait simple à résoudre.

Pour démarrer l'étude de l'informatique quantique avec ce livre, vous n'avez pas besoin de connaître la physique quantique, vous n'avez pas non plus besoin de compétences avancées en programmation (un peu de *Python*). Les mathématiques de ce cours sont d'un niveau première année d'études supérieures, avec des incursions vers la deuxième année. Toutes les notions de bases sont introduites, en particulier les nombres complexes jouent un rôle important (d'ailleurs les nombre α et β ci-dessus sont des nombres complexes) ainsi que les vecteurs et les matrices.

L'informatique quantique est un monde déconcertant mais bien réel. À vous de le découvrir !

Le cours est aussi disponible en vidéos :

[Youtube](#) : « Quantum ».

L'intégralité des codes *Python* ainsi que tous les fichiers sources sont sur la page *GitHub* d'Exo7 :

« [GitHub](#) : quantum-exo7 ».

Sommaire

I Premiers pas quantiques	1
1 Découverte de l'informatique quantique	2
2 Utiliser un ordinateur quantique (avec Qiskit)	21
3 Nombres complexes	33
4 Vecteurs et matrices	47
5 Informatique classique	66
6 Physique quantique	74
7 Téléportation quantique	85
II Algorithmes quantiques	95
8 Un premier algorithme quantique	96
9 Portes quantiques	104
10 Algorithme de Deutsch–Jozsa	118
11 Algorithme de Grover	125
III Algorithme de Shor	142
12 Arithmétique	143
13 Algorithme de Shor	155
14 Compléments d'arithmétique	169
15 Transformée de Fourier discrète	184
IV Vivre dans un monde quantique	201
16 Cryptographie quantique	202

17	Code correcteur	208
18	Avantage quantique	217
	Index	

Résumé des chapitres

Découverte de l'informatique quantique

Ce chapitre donne un aperçu des calculs avec les qubits et est une introduction détaillée des chapitres suivants dans lesquels plusieurs notions seront revues : nombres complexes, vecteurs, matrices, portes logiques, physique quantique. Ce chapitre se termine par une application assez difficile : le codage super-dense.

Utiliser un ordinateur quantique (avec Qiskit)

Le but est de programmer des circuits quantiques et de simuler les résultats. Mais nous allons aussi utiliser un véritable ordinateur quantique.

Nombres complexes

Les nombres complexes sont les coefficients naturels des qubits. Nous détaillons les calculs avec les nombres complexes ainsi que sur les qubits.

Vecteurs et matrices

Un qubit est un vecteur et les opérations sur les qubits sont codées par des matrices. Nous étudions ici le calcul sur les vecteurs, les matrices et leur lien avec les qubits.

Informatique classique

Nous rappelons quelques principes de base du fonctionnement d'un ordinateur classique avec les notions de bits, de portes logiques et de complexité d'un algorithme.

Physique quantique

L'objectif est de comprendre les notions de base de la physique quantique.

Téléportation quantique

La téléportation quantique permet de transmettre un qubit d'un point A à un point B .

Un premier algorithme quantique

Nous commençons par étudier une version simple de l'algorithme de Deutsch–Jozsa afin de nous familiariser avec les objets, les techniques et les types d'algorithmes que nous découvrirons dans la seconde partie du livre.

Portes quantiques

Nous approfondissons nos connaissances théoriques des portes quantiques en étudiant ce qu'elles peuvent réaliser (ou pas !).

Algorithme de Deutsch–Jozsa

Nous expliquons et prouvons l'algorithme de Deutsch–Jozsa dans le cas général. C'est notre premier algorithme quantique qui supprime les algorithmes classiques et c'est aussi l'occasion d'introduire plusieurs notions utiles pour la suite.

Algorithme de Grover

L'algorithme de Grover est un algorithme de recherche d'un élément dans une liste qui est plus efficace que les algorithmes classiques. Son principe est simple, même si sa mise en œuvre est un peu complexe. L'algorithme de Grover ne fournit pas un résultat sûr à 100 %, mais une réponse qui a de grandes chances d'être la bonne.

Arithmétique

La sécurité des communications sur internet est basée sur l'arithmétique et en particulier sur le système de cryptographie RSA qui repose sur la difficulté de factoriser de très grands entiers avec un ordinateur classique. Nous présentons dans ce chapitre les notions essentielles d'arithmétique afin de comprendre plus tard l'algorithme de Shor qui permet de factoriser rapidement un entier à l'aide d'un ordinateur quantique.

Algorithme de Shor

Nous détaillons le circuit et les calculs qui permettent une factorisation rapide des entiers à l'aide d'un ordinateur quantique.

Compléments d'arithmétique

Nous apportons des compléments à l'algorithme de Shor en étudiant chacune des hypothèses.

Transformée de Fourier discrète

Nous revenons sur l'outil principal de l'algorithme de Shor : la transformée de Fourier. Nous expliquons comment elle est construite, comment la réaliser par un circuit quantique et quelles sont ses autres applications.

Cryptographie quantique

Nous étudions le protocole BB84 qui permet le partage d'un secret commun entre deux personnes grâce à la physique quantique.

Code correcteur

Lors de la transmission d'un qubit il peut y avoir des erreurs. Les codes correcteurs permettent de détecter et corriger ces erreurs.

Avantage quantique

Quand est-ce qu'un ordinateur quantique sera plus performant qu'un ordinateur classique ?

PREMIÈRE PARTIE

| 0 >

| 0 >

PREMIERS PAS QUANTIQUES

Découverte de l'informatique quantique

Chapitre

1

Vidéo ■ partie 1.1. Un qubit

Vidéo ■ partie 1.2. Portes quantiques

Vidéo ■ partie 1.3. Les 2-qubits (partie 1)

Vidéo ■ partie 1.3. Les 2-qubits (partie 2)

Vidéo ■ partie 1.4. Plus de qubits

Vidéo ■ partie 1.5. Communication par codage super-dense

Plongeons directement au cœur de l'informatique quantique en abordant la notion de qubit et les circuits quantiques fondamentaux.

Ce chapitre donne un aperçu des calculs avec les qubits et est une introduction détaillée des chapitres suivants dans lesquels plusieurs notions seront revues : nombres complexes, vecteurs, matrices, portes logiques, physique quantique. Ce chapitre se termine par une application assez difficile : le codage super-dense.

1. Un qubit

Pour un ordinateur classique l'unité d'information est le **bit** représenté soit par 0, soit par 1. Avec plusieurs bits on peut coder un entier, par exemple 19 est codé en binaire par 1.0.0.1.1 ; on peut aussi coder des caractères, par exemple le code ASCII de « A » est 1.0.0.0.0.0.1.

1.1. Un qubit est un vecteur

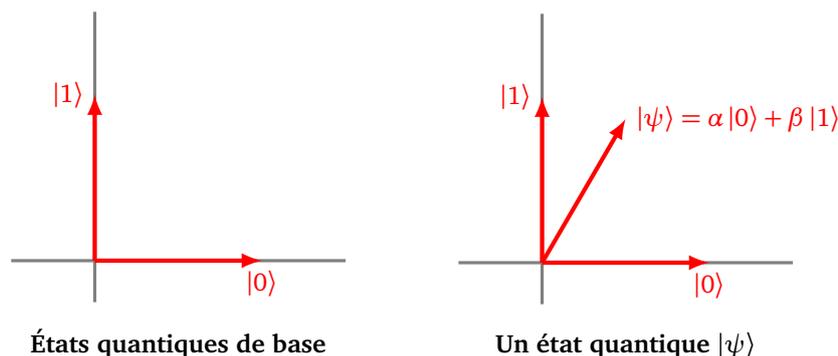
En informatique quantique on part aussi de deux **états quantiques de base** :

$$|0\rangle \quad \text{et} \quad |1\rangle.$$

La notation est un peu bizarre (elle sera justifiée ultérieurement). En fait $|0\rangle$ et $|1\rangle$ sont deux vecteurs :

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \text{et} \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

Ces deux vecteurs forment une base orthonormée du plan.



Ce qui est nouveau et fondamental est que l'on peut *superposer* ces deux états $|0\rangle$ et $|1\rangle$. Un *qubit* est un *état quantique* obtenu par combinaison linéaire :

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

Ainsi, un qubit est représenté par un vecteur :

$$|\psi\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}.$$

En effet :

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \alpha \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \beta \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}.$$

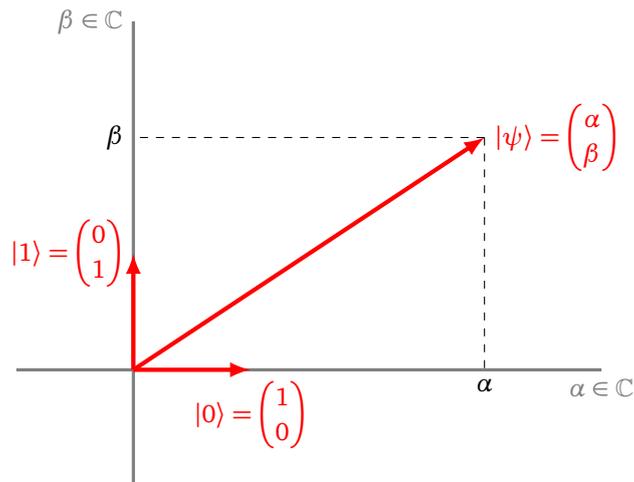
Vocabulaire.

- Les états $|0\rangle$ et $|1\rangle$ se lisent « ket zéro » et « ket un » (« ket » se prononce comme le mot « quête »).
- ψ est la lettre grecque « psi », ainsi $|\psi\rangle$ se lit « ket psi ».

Là où cela se complique un peu, c'est que les coefficients α et β ne sont pas des nombres réels mais des nombres complexes :

$$\alpha \in \mathbb{C} \quad \text{et} \quad \beta \in \mathbb{C}$$

Ainsi $|\psi\rangle$ est un vecteur de \mathbb{C}^2 , défini par ses deux coordonnées complexes α et β .



Sur la figure ci-dessus, on a représenté un vecteur à coordonnées complexes comme un vecteur du plan. Cette figure aide à la compréhension mais ne correspond pas tout à fait à la réalité. Comme chacun des axes correspond à une coordonnée complexe (de dimension 2), un dessin réaliste nécessiterait quatre dimensions.

Exemple.

- $|\psi\rangle = (3 + 4i)|0\rangle + (2 - 8i)|1\rangle$. On rappelle que i est le nombre complexe tel que $i^2 = -1$.
- $|\psi\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{i}{\sqrt{2}}|1\rangle$.
- On peut superposer des états par addition, par exemple :

$$(2|0\rangle + (1 + i)|1\rangle) + (i|0\rangle + (2 - 3i)|1\rangle) = (2 + i)|0\rangle + (3 - 2i)|1\rangle,$$

ce qui correspond à additionner deux vecteurs :

$$\begin{pmatrix} 2 \\ 1 + i \end{pmatrix} + \begin{pmatrix} i \\ 2 - 3i \end{pmatrix} = \begin{pmatrix} 2 + i \\ 3 - 2i \end{pmatrix}.$$

Remarque.

- Si on souhaitait définir $|\psi\rangle$ uniquement avec des nombres réels, alors on pourrait écrire $\alpha = \alpha_1 + i\alpha_2$, $\beta = \beta_1 + i\beta_2$ et dire qu'un état quantique est défini par 4 nombres réels $\alpha_1, \alpha_2, \beta_1, \beta_2$. Cependant ce n'est pas le bon état d'esprit pour la suite.
- Attention $|0\rangle$ n'est pas le vecteur nul $\begin{pmatrix} 0 \\ 0 \end{pmatrix}$, mais bien le vecteur $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$.

1.2. Norme

États de norme 1. On va principalement considérer les états $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$ dont la **norme est égale à 1**, c'est-à-dire :

$$|\alpha|^2 + |\beta|^2 = 1$$

où $|\alpha|$ et $|\beta|$ sont les modules des coefficients complexes. On rappelle que si $z = a + ib$ est un nombre complexe (avec $a, b \in \mathbb{R}$), alors son **module** $|z|$ est le nombre réel positif défini par $|z|^2 = a^2 + b^2$.

Exemple.

- $|\psi\rangle = \frac{1}{\sqrt{2}} |0\rangle + \frac{1}{\sqrt{2}} |1\rangle$.
Alors

$$|\alpha|^2 + |\beta|^2 = \left| \frac{1}{\sqrt{2}} \right|^2 + \left| \frac{1}{\sqrt{2}} \right|^2 = \frac{1}{2} + \frac{1}{2} = 1.$$

Ainsi cet état $|\psi\rangle$ est bien de norme 1.

- $|\psi\rangle = (3 + 4i) |0\rangle + (2 - 8i) |1\rangle$.

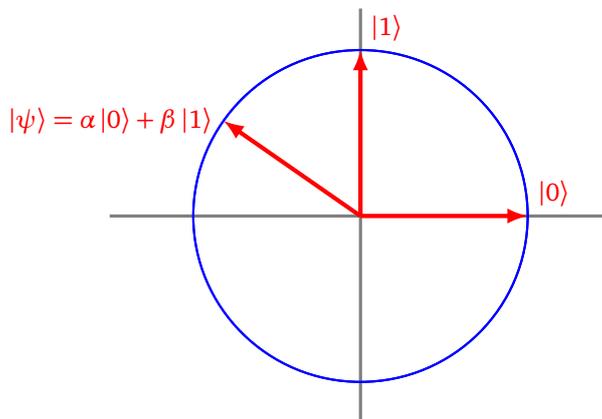
$$|\alpha|^2 + |\beta|^2 = |3 + 4i|^2 + |2 - 8i|^2 = 25 + 68 = 93.$$

Ainsi la norme de $|\psi\rangle$ est $\sqrt{|\alpha|^2 + |\beta|^2} = \sqrt{93}$ et n'est pas égale à 1. En divisant par la norme, on transforme facilement $|\psi\rangle$ en un état $|\psi'\rangle$ de norme 1 :

$$|\psi'\rangle = \frac{3 + 4i}{\sqrt{93}} |0\rangle + \frac{2 - 8i}{\sqrt{93}} |1\rangle.$$

Remarque.

On peut schématiser de façon imparfaite les états de norme 1 par le dessin du cercle ci-dessous.



Cependant ceci est un dessin où l'on considère que les coefficients α et β sont des nombres réels, ce qui n'est pas le cas en général. La « sphère de Bloch » fournira une représentation plus fidèle, voir le chapitre « Nombres complexes ».

1.3. Mesure et probabilités

Un des aspects fondamentaux mais troublants de la physique quantique est que l'on ne peut pas mesurer les coefficients α et β de l'état quantique $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$. Partons d'un état quantique de norme 1 :

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \quad \text{avec} \quad |\alpha|^2 + |\beta|^2 = 1.$$

La **mesure** de l'état quantique $|\psi\rangle$ va renvoyer l'un des bits classiques 0 ou 1 :

- 0 avec une probabilité $|\alpha|^2$
- 1 avec une probabilité $|\beta|^2$

Noter que, comme nous sommes partis d'un état de norme 1, nous avons bien la somme des probabilités $|\alpha|^2 + |\beta|^2$ qui vaut 1.

Exemple.

Considérons l'état quantique :

$$|\psi\rangle = \frac{1-i}{\sqrt{3}}|0\rangle + \frac{1+2i}{\sqrt{15}}|1\rangle.$$

Alors

$$|\alpha|^2 = \left| \frac{1-i}{\sqrt{3}} \right|^2 = \frac{2}{3}$$

et

$$|\beta|^2 = \left| \frac{1+2i}{\sqrt{15}} \right|^2 = \frac{5}{15} = \frac{1}{3}.$$

On a bien $|\alpha|^2 + |\beta|^2 = 1$. Si on mesure $|\psi\rangle$ alors on obtient 0 avec une probabilité $\frac{2}{3}$ et 1 avec une probabilité $\frac{1}{3}$.

Autrement dit, si je peux répéter 100 fois l'expérience « je prépare l'état initial $|\psi\rangle$, puis je le mesure », alors pour environ 66 cas sur 100 j'obtiendrai pour mesure 0 et pour environ 33 cas sur 100 j'obtiendrai 1.

La mesure d'un état quantique $|\psi\rangle$ le perturbe de façon irrémédiable. C'est un phénomène physique appelé « réduction du paquet d'onde ». Si la mesure a donné le bit 0, alors l'état $|\psi\rangle$ est devenu $|0\rangle$, si la mesure a donné le bit 1 alors $|\psi\rangle$ est devenu $|1\rangle$. Autrement dit, une fois qu'il est mesuré, un qubit ne sert plus à grand chose !

Remarque.

Bien évidemment la mesure de $|0\rangle$ donne 0 avec une probabilité 1 (l'événement est presque sûr). De même la mesure de $|1\rangle$ donne 1 avec une probabilité 1. Dans ce cours nous faisons le choix qu'une mesure renvoie un bit classique 0 ou 1. Une autre convention serait de décider qu'une mesure renvoie un des états de base $|0\rangle$ ou $|1\rangle$.

Bilan. On retient qu'à partir d'un état $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ avec $\alpha, \beta \in \mathbb{C}$ tels que $|\alpha|^2 + |\beta|^2 = 1$:

- on ne peut pas mesurer les coefficients α et β ;
- la mesure de $|\psi\rangle$ renvoie soit 0 avec une probabilité $|\alpha|^2$, soit 1 avec une probabilité $|\beta|^2$;
- la mesure transforme le qubit $|\psi\rangle$ en $|0\rangle$ ou en $|1\rangle$, les coefficients α et β ont disparu après mesure.

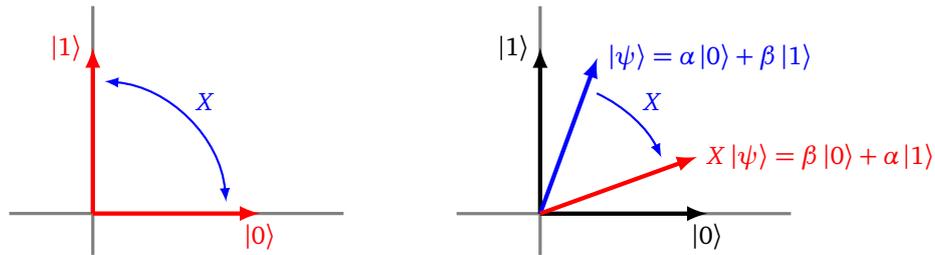
2. Porte avec une entrée

Un ordinateur quantique produit des qubits et leur applique des transformations, qui dans un circuit s'appellent des « portes ». Nous commençons par transformer un seul qubit.

2.1. Porte X

La porte X s'appelle aussi porte NON (ou NOT) et est la transformation qui échange les deux états quantiques de base :

$$|0\rangle \xrightarrow{X} |1\rangle \quad \text{et} \quad |1\rangle \xrightarrow{X} |0\rangle$$



Porte X

La transformation est de plus linéaire, ce qui fait que la porte X échange les deux coefficients d'un état quantique :

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \xrightarrow{X} \beta|0\rangle + \alpha|1\rangle.$$

Par exemple l'état $|\psi\rangle = 2|0\rangle + (1-i)|1\rangle$ est transformé par la porte X en l'état $X(|\psi\rangle) = (1-i)|0\rangle + 2|1\rangle$.

En termes de vecteurs cette transformation s'écrit :

$$\begin{pmatrix} 1 \\ 0 \end{pmatrix} \xrightarrow{X} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad \begin{pmatrix} 0 \\ 1 \end{pmatrix} \xrightarrow{X} \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \xrightarrow{X} \begin{pmatrix} \beta \\ \alpha \end{pmatrix}$$

La matrice de la porte X est donc :

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

car

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

Note. La notion de matrice n'est pas indispensable pour ce premier chapitre, elle sera développée dans le chapitre « Vecteurs et matrices ».

2.2. Porte H de Hadamard

La porte H de Hadamard est la transformation linéaire définie par :

$$|0\rangle \xrightarrow{H} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \quad \text{et} \quad |1\rangle \xrightarrow{H} \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle).$$

Ainsi, si $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ alors

$$H(|\psi\rangle) = \alpha \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) + \beta \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle).$$

On regroupe les coefficients selon les termes $|0\rangle$ et $|1\rangle$, pour obtenir :

$$H(|\psi\rangle) = \frac{\alpha + \beta}{\sqrt{2}}|0\rangle + \frac{\alpha - \beta}{\sqrt{2}}|1\rangle.$$

Par exemple l'état $|\psi\rangle = i|0\rangle + (2+i)|1\rangle$ est transformé en $H(|\psi\rangle) = \frac{2+2i}{\sqrt{2}}|0\rangle - \frac{2}{\sqrt{2}}|1\rangle$.

En termes de vecteurs cette transformation s'écrit sur les vecteurs de base :

$$\begin{pmatrix} 1 \\ 0 \end{pmatrix} \xrightarrow{H} \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad \begin{pmatrix} 0 \\ 1 \end{pmatrix} \xrightarrow{H} \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

La matrice de la porte H est donc :

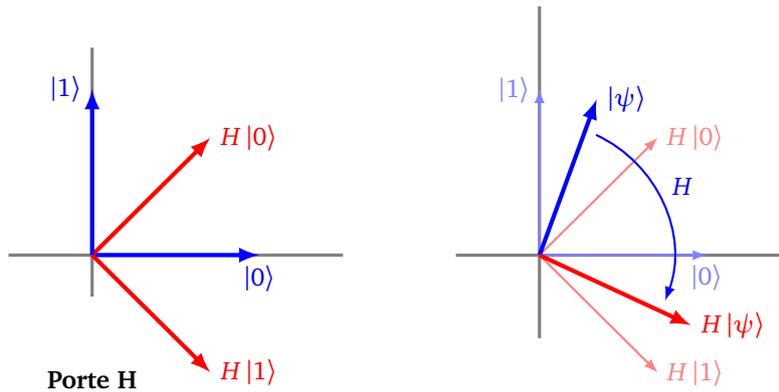
$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

car la multiplication

$$\frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$

redonne bien le vecteur correspondant à $H(|\psi\rangle)$.

Géométriquement la base $(|0\rangle, |1\rangle)$ est transformée en une autre base orthonormée $(H(|0\rangle), H(|1\rangle))$.



Remarque. Il est fréquent de rencontrer les notations suivantes :

$$|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \quad \text{et} \quad |-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$$

même si nous éviterons de les utiliser ici.

2.3. Mesure

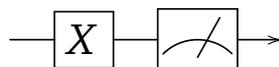
C'est un élément fondamental d'un circuit quantique. C'est le seul moment où l'on peut obtenir une information sur un état quantique $|\psi\rangle$, mais c'est aussi la fin du qubit, car la mesure ne renvoie que 0 ou 1 et perturbe irrémédiablement l'état quantique.

2.4. Exemples de circuit

Un **circuit** est composé d'une succession de portes. Il se lit de gauche à droite.

Exemple.

Voici un circuit composé d'une porte X (c'est-à-dire une porte NON) suivie d'une porte mesure symbolisée par un petit cadran.



- Si l'entrée est $|0\rangle$, alors $X(|0\rangle) = |1\rangle$, la sortie mesurée vaut donc 1 (avec une probabilité 1) :

$$|0\rangle \xrightarrow{X} \text{mesure} \rightarrow 1$$

- Par contre si l'entrée est $|1\rangle$, alors $X(|1\rangle) = |0\rangle$ et la sortie mesurée vaut 0 :

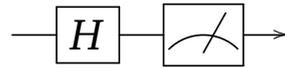
$$|1\rangle \xrightarrow{X} \text{mesure} \rightarrow 0$$

- Si l'entrée est l'état $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ (avec $|\alpha|^2 + |\beta|^2 = 1$), alors $X(|\psi\rangle) = \beta|0\rangle + \alpha|1\rangle$. La mesure donne donc 0 avec une probabilité $|\beta|^2$ et 1 avec une probabilité $|\alpha|^2$.

$$\alpha|0\rangle + \beta|1\rangle \xrightarrow{X} \text{mesure} \rightarrow 0 \text{ ou } 1$$

Exemple.

Ce circuit est formé d'une porte H de Hadamard, suivi d'une mesure :



- Si l'entrée est $|0\rangle$, alors $H(|0\rangle) = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$, la mesure donne donc le bit 0 avec une probabilité $\frac{1}{2}$ et le bit 1 avec une probabilité $\frac{1}{2}$.
- Si l'entrée est $|1\rangle$, alors $H(|1\rangle) = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$ et les mesures conduisent aux mêmes résultats que précédemment.
- Par contre si l'entrée est $|\psi\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$, alors :

$$\begin{aligned} H(|\psi\rangle) &= H\left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle\right) \\ &= \frac{1}{\sqrt{2}}H(|0\rangle) + \frac{1}{\sqrt{2}}H(|1\rangle) \\ &= \frac{1}{\sqrt{2}}\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) + \frac{1}{\sqrt{2}}\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \\ &= \frac{1}{2}|0\rangle + \frac{1}{2}|0\rangle + \frac{1}{2}|1\rangle - \frac{1}{2}|1\rangle \\ &= |0\rangle \end{aligned}$$

Ainsi pour cette entrée, le circuit renvoie, après mesure, 0 avec une quasi-certitude.

- Exercice : trouver $|\psi\rangle$ tel que la mesure donne 1 avec une quasi-certitude.

2.5. Portes X, Y et Z de Pauli

Nous avons déjà rencontré la porte X (dite aussi porte NOT), qui fait partie d'une famille de trois portes, dites *portes de Pauli*. Les voici définies par leur action sur les états quantiques de base $|0\rangle$ et $|1\rangle$, et également par leur matrice.

Porte X

$$\xrightarrow{X} \quad \begin{cases} |0\rangle \mapsto |1\rangle \\ |1\rangle \mapsto |0\rangle \end{cases} \quad X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Porte Y

$$\xrightarrow{Y} \quad \begin{cases} |0\rangle \mapsto i|1\rangle \\ |1\rangle \mapsto -i|0\rangle \end{cases} \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$$

Porte Z

$$\xrightarrow{Z} \quad \begin{cases} |0\rangle \mapsto |0\rangle \\ |1\rangle \mapsto -|1\rangle \end{cases} \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Exercice.

On considère la porte \sqrt{NOT} définie par

$$\xrightarrow{\sqrt{NOT}} \quad \begin{cases} |0\rangle \mapsto \frac{1+i}{2}|0\rangle + \frac{1-i}{2}|1\rangle \\ |1\rangle \mapsto \frac{1-i}{2}|0\rangle + \frac{1+i}{2}|1\rangle \end{cases} \quad \text{c'est-à-dire} \quad M = \frac{1}{2} \begin{pmatrix} 1+i & 1-i \\ 1-i & 1+i \end{pmatrix}$$

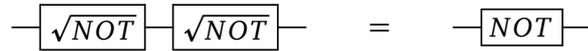
1. Pour l'entrée $|0\rangle$, que donne une mesure après la porte \sqrt{NOT} ? Même question avec $|1\rangle$.



2. Pour l'entrée $|\psi\rangle = \frac{1}{2}|0\rangle + i\frac{\sqrt{3}}{2}|1\rangle$, que donne la sortie après la porte \sqrt{NOT} ? Que donne ensuite une mesure?



3. Montrer que le circuit suivant, qui consiste à enchaîner deux portes \sqrt{NOT} , équivaut à une seule porte NOT (notée aussi porte X).



Autrement dit, il s'agit de montrer que :

$$\sqrt{NOT}(\sqrt{NOT}(|\psi\rangle)) = NOT(|\psi\rangle)$$

Indication. On peut faire les calculs avec un qubit général $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$. Mais on peut aussi seulement vérifier que cette affirmation est vraie pour les deux états de bases $|0\rangle$ et $|1\rangle$, ce qui est suffisant par linéarité. Une autre technique serait d'utiliser les matrices.

3. Les 2-qubits

Nous allons maintenant réunir deux qubits pour obtenir un 2-qubit. C'est la version quantique de l'union de deux bits.

3.1. Superposition

Deux qubits réunis sont dans un état quantique $|\psi\rangle$, appelé **2-qubit**, défini par la superposition :

$$|\psi\rangle = \alpha|0.0\rangle + \beta|0.1\rangle + \gamma|1.0\rangle + \delta|1.1\rangle$$

où $\alpha, \beta, \gamma, \delta \in \mathbb{C}$, avec souvent la convention de normalisation :

$$|\alpha|^2 + |\beta|^2 + |\gamma|^2 + |\delta|^2 = 1.$$

La mesure d'un 2-qubit, donne deux bits classiques :

- 0.0 avec probabilité $|\alpha|^2$,
- 0.1 avec probabilité $|\beta|^2$,
- 1.0 avec probabilité $|\gamma|^2$,
- 1.1 avec probabilité $|\delta|^2$.

Notons déjà la différence remarquable avec l'informatique classique : avec deux bits classiques, on encode un seul des quatre états 0.0, 0.1, 1.0 ou 1.1, mais avec un 2-qubit on encode en quelque sorte les quatre états en même temps!

Que représentent $|0.0\rangle, |0.1\rangle, \dots$? Il s'agit de nouveaux vecteurs d'une base mais cette fois en dimension 4 :

$$|0.0\rangle = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad |0.1\rangle = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \quad |1.0\rangle = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \quad |1.1\rangle = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Ainsi $|\psi\rangle$ est un vecteur de \mathbb{C}^4 :

$$|\psi\rangle = \alpha \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \beta \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} + \gamma \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} + \delta \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} \alpha \\ \beta \\ \gamma \\ \delta \end{pmatrix}.$$

Exemple.

$$|\psi\rangle = \frac{1}{\sqrt{6}} |0.0\rangle + \frac{i}{\sqrt{6}} |1.0\rangle + \frac{1+i}{\sqrt{3}} |1.1\rangle$$

est un 2-qubit de norme 1. Sa mesure donne :

- 0.0 avec probabilité 1/6,
- 0.1 avec probabilité 0,
- 1.0 avec probabilité 1/6,
- 1.1 avec probabilité 2/3.

On peut aussi noter les états de base par des formules de multiplications :

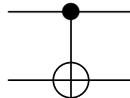
$$|0.0\rangle = |0\rangle \cdot |0\rangle \quad |0.1\rangle = |0\rangle \cdot |1\rangle \quad |1.0\rangle = |1\rangle \cdot |0\rangle \quad |1.1\rangle = |1\rangle \cdot |1\rangle$$

On note aussi ce produit par le symbole \otimes :

$$|0.1\rangle = |0\rangle \otimes |1\rangle = \begin{matrix} |0\rangle \\ \otimes \\ |1\rangle \end{matrix}$$

3.2. Porte CNOT

La porte *CNOT* est une porte qui prend en entrée deux qubits et renvoie deux qubits en sortie.



Voici la règle sur les quatre états quantiques de bases :



Autrement dit le premier qubit reste inchangé. C'est différent pour le second qubit :

- si le premier qubit est $|0\rangle$ alors le second qubit est inchangé,
- si le premier qubit est $|1\rangle$ alors le second qubit est changé selon la règle d'une porte *X* : $|0\rangle \mapsto |1\rangle$ et $|1\rangle \mapsto |0\rangle$.

On peut interpréter cette porte comme une instruction « si ..., sinon ... » : si le premier qubit est $|0\rangle$ faire ceci, sinon faire cela.

Voici la règle reformulée avec la notation des 2-qubits :

$$|0.0\rangle \xrightarrow{CNOT} |0.0\rangle \quad |0.1\rangle \xrightarrow{CNOT} |0.1\rangle \quad |1.0\rangle \xrightarrow{CNOT} |1.1\rangle \quad |1.1\rangle \xrightarrow{CNOT} |1.0\rangle$$

Voici cette même règle présentée à l'aide de vecteurs :

$$\begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \mapsto \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \mapsto \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \quad \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \mapsto \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \quad \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \mapsto \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

La matrice de la transformation de *CNOT* est donc la matrice 4×4 :

$$M = \left(\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{array} \right).$$

La porte *CNOT* transforme un vecteur représentant un 2-qubit par multiplication par cette matrice *M* :

$$\begin{pmatrix} \alpha \\ \beta \\ \gamma \\ \delta \end{pmatrix} \xrightarrow{CNOT} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \\ \gamma \\ \delta \end{pmatrix} = \begin{pmatrix} \alpha \\ \beta \\ \delta \\ \gamma \end{pmatrix}.$$

3.3. L'état de Bell

À l'aide de la porte *CNOT* nous allons obtenir un des états les plus importants pour deux qubits : l'état de Bell :

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}} |0.0\rangle + \frac{1}{\sqrt{2}} |1.1\rangle$$

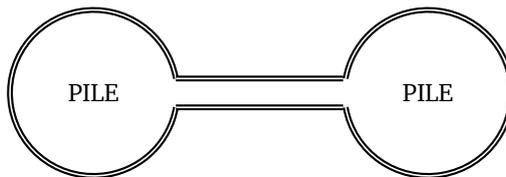
Une mesure de cet état conduit à :

- 0.0 avec une probabilité $\frac{1}{2}$,
- 1.1 avec une probabilité $\frac{1}{2}$,
- les deux autres sorties 0.1 et 1.0 ayant une probabilité nulle.

Remarque.

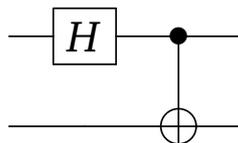
En physique quantique il est toujours aventureux de faire des analogies avec le monde tel qu'on le connaît. Permettons-nous un petit écart :

- Un qubit, c'est un peu comme une pièce de monnaie lancée en l'air. Tant que la pièce tourne dans l'air, « pile » et « face » ont les mêmes chances de se produire. Ce n'est que lorsque la pièce est retombée que l'on peut lire le résultat (c'est la partie « mesure ») et ensuite le résultat est définitivement figé à « pile » ou bien à « face ».
- Un 2-qubit, c'est-à-dire la réunion de deux qubits, c'est comme deux pièces de monnaie en train d'être lancées en l'air en même temps. Les quatre résultats « pile/pile », « pile/face », « face/pile » ou encore « face/face » sont possibles.
- L'état de Bell, c'est comme deux pièces liées entre elles lancées en l'air. Le résultat ne peut être que « pile/pile » ou bien « face/face ». Ce phénomène s'appelle « l'intrication quantique ».



Obtention de l'état de Bell.

Considérons le circuit suivant, composé d'une porte de Hadamard, suivie d'une porte *CNOT* :



Alors, à partir de l'entrée $|0.0\rangle$, l'état de Bell $|\Phi^+\rangle$ est obtenu en sortie.

$$\begin{matrix} |0\rangle \\ \otimes \\ |0\rangle \end{matrix} \begin{matrix} \xrightarrow{H} \\ \bullet \\ \oplus \end{matrix} \begin{matrix} \frac{1}{\sqrt{2}} |0\rangle \\ \otimes \\ |0\rangle \end{matrix} + \begin{matrix} \frac{1}{\sqrt{2}} |1\rangle \\ \otimes \\ |1\rangle \end{matrix}$$

Reprenons le calcul en détails (en adoptant la notation verticale) à partir de l'entrée

$$|0.0\rangle = \begin{array}{c} |0\rangle \\ \otimes \\ |0\rangle \end{array}$$

Tout d'abord le premier qubit (celui du haut) passe par une porte H , le second qubit reste inchangé :

$$\begin{array}{c} |0\rangle \\ \otimes \\ |0\rangle \end{array} \xrightarrow{H} \begin{array}{c} H(|0\rangle) \\ \otimes \\ |0\rangle \end{array} = \begin{array}{c} \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle \\ \otimes \\ |0\rangle \end{array} = \frac{1}{\sqrt{2}} \begin{array}{c} |0\rangle \\ \otimes \\ |0\rangle \end{array} + \frac{1}{\sqrt{2}} \begin{array}{c} |1\rangle \\ \otimes \\ |0\rangle \end{array}$$

Ensuite ce résultat intermédiaire passe par la porte $CNOT$. On regarde d'abord indépendamment les deux termes de la somme obtenue :

$$\begin{array}{c} |0\rangle \\ \otimes \\ |0\rangle \end{array} \xrightarrow{CNOT} \begin{array}{c} |0\rangle \\ \otimes \\ |0\rangle \end{array} \quad \text{et} \quad \begin{array}{c} |1\rangle \\ \otimes \\ |0\rangle \end{array} \xrightarrow{CNOT} \begin{array}{c} |1\rangle \\ \otimes \\ |1\rangle \end{array}$$

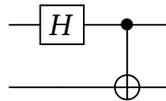
Ainsi par linéarité, la porte $CNOT$ a pour action :

$$\begin{array}{c} H(|0\rangle) \\ \otimes \\ |0\rangle \end{array} \xrightarrow{CNOT} \frac{1}{\sqrt{2}} \begin{array}{c} |0\rangle \\ \otimes \\ |0\rangle \end{array} + \frac{1}{\sqrt{2}} \begin{array}{c} |1\rangle \\ \otimes \\ |1\rangle \end{array}$$

qui est bien l'état de Bell $|\Phi^+\rangle$.

Exercice.

Reprenons le même circuit :



1. Quelle est la sortie produite pour l'entrée $|1.0\rangle$?
2. Trouver où insérer une porte X dans le circuit, de sorte que l'entrée $|0.0\rangle$ conduise à la sortie $\frac{1}{\sqrt{2}}|0.1\rangle + \frac{1}{\sqrt{2}}|1.0\rangle$.

3.4. Calculs algébriques avec un ou deux qubits

Il faut savoir faire des calculs algébriques avec les qubits, même si pour vraiment comprendre ces opérations il faudra attendre le produit tensoriel qui sera expliqué dans le chapitre « Vecteurs et matrices ».

Addition.

L'addition se fait coefficient par coefficient et ne pose pas de problème, par exemple si

$$|\phi\rangle = (1 + 3i)|0\rangle + 2i|1\rangle \quad \text{et} \quad |\psi\rangle = 3|0\rangle + (1 - i)|1\rangle$$

alors

$$|\phi\rangle + |\psi\rangle = (4 + 3i)|0\rangle + (1 + i)|1\rangle.$$

Ou encore pour des 2-qubits :

$$(|1.0\rangle + |0.1\rangle) + (|1.0\rangle - |0.1\rangle) = 2|1.0\rangle.$$

Multiplication.

On peut multiplier deux 1-qubits pour obtenir un 2-qubit. Les calculs se font comme des calculs algébriques à l'aide des règles de bases $|0\rangle \cdot |0\rangle = |0.0\rangle$, $|0\rangle \cdot |1\rangle = |0.1\rangle$,...

Par exemple :

$$\begin{aligned} (3|0\rangle + 2i|1\rangle) \cdot ((1 + i)|0\rangle - |1\rangle) &= 3(1 + i)|0\rangle \cdot |0\rangle - 3|0\rangle \cdot |1\rangle + 2i(1 + i)|1\rangle \cdot |0\rangle - 2i|1\rangle \cdot |1\rangle \\ &= (3 + 3i)|0.0\rangle - 3|0.1\rangle + (-2 + 2i)|1.0\rangle - 2i|1.1\rangle. \end{aligned}$$

On a utilisé l'identité $i^2 = -1$ et fait attention que la multiplication des « ket » n'est pas commutative : $|0\rangle \cdot |1\rangle \neq |1\rangle \cdot |0\rangle$. En particulier on a la relation $(k|a\rangle) \cdot |b\rangle = |a\rangle \cdot (k|b\rangle) = k|a \cdot b\rangle$ pour $k \in \mathbb{C}$. Cette relation a été utilisée précédemment sans le dire pour la porte $CNOT$: $(\frac{1}{\sqrt{2}}|0\rangle) \cdot |0\rangle = \frac{1}{\sqrt{2}}|0 \cdot 0\rangle$. On a aussi la relation de développement/factorisation $|(a+b) \cdot c\rangle = |a \cdot c\rangle + |b \cdot c\rangle$. Par exemple : $|(0+1) \cdot 1\rangle = |0 \cdot 1\rangle + |1 \cdot 1\rangle$.

Norme.

- Pour un nombre réel x , $|x|$ est sa valeur absolue.
- Pour un nombre complexe $z = a + ib$, $|z| = \sqrt{a^2 + b^2}$ est son module.
- Pour un qubit $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, $\|\psi\| = \sqrt{|\alpha|^2 + |\beta|^2}$ est sa norme.
- Pour un 2-qubit $|\psi\rangle = \alpha|0.0\rangle + \beta|0.1\rangle + \gamma|1.0\rangle + \delta|1.1\rangle$, sa norme est $\|\psi\| = \sqrt{|\alpha|^2 + |\beta|^2 + |\gamma|^2 + |\delta|^2}$.
- La normalisation d'un qubit $|\psi\rangle$ est $\frac{|\psi\rangle}{\|\psi\|}$ qui est un qubit de norme 1.

Exercice.

Soit $|\phi\rangle = \frac{1}{\sqrt{3}}|0\rangle + \frac{\sqrt{2}}{\sqrt{3}}i|1\rangle$ et $|\psi\rangle = \frac{2+i}{\sqrt{10}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$. Calculer la norme de $|\phi\rangle$, $|\psi\rangle$, $|\phi\rangle + |\psi\rangle$ et $|\phi\rangle \cdot |\psi\rangle$. Conclusion : on note que la somme de deux qubits de norme 1 n'est pas nécessairement de norme 1, par contre le produit de deux qubits de norme 1 est encore un qubit de norme 1.

Exercice.

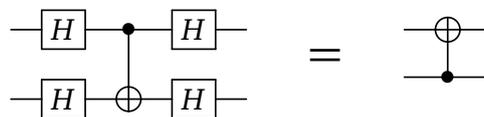
Dans une porte $CNOT$ les deux entrées ne jouent pas des rôles symétriques.



Sur la figure à droite, est dessinée une porte $CNOT$ renversée pour laquelle c'est le premier qubit qui change (ou non) en fonction du second qubit.

Cependant on peut construire la porte $CNOT$ renversée à partir de la porte $CNOT$ classique et de quatre portes H de Hadamard.

Montrer que les circuits suivants sont équivalents :

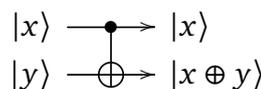


Indication. Il suffit de vérifier que l'affirmation est vraie pour les quatre états de base $|0.0\rangle, |0.1\rangle, |1.0\rangle, |1.1\rangle$.

La porte $CNOT$. Revisitons la porte $CNOT$ d'une manière un peu plus abstraite. La transformation associée à cette porte s'écrit aussi :

$$|x \cdot y\rangle \xrightarrow{CNOT} |x \cdot y \oplus x\rangle$$

c'est-à-dire :



où x et y ont pour valeurs 0 ou 1 et où « \oplus » représente l'addition usuelle sur un bit (comme une porte XOR) :

$$0 \oplus 0 = 0 \quad 1 \oplus 0 = 1 \quad 0 \oplus 1 = 1 \quad \text{et} \quad 1 \oplus 1 = 0.$$

Par exemple :

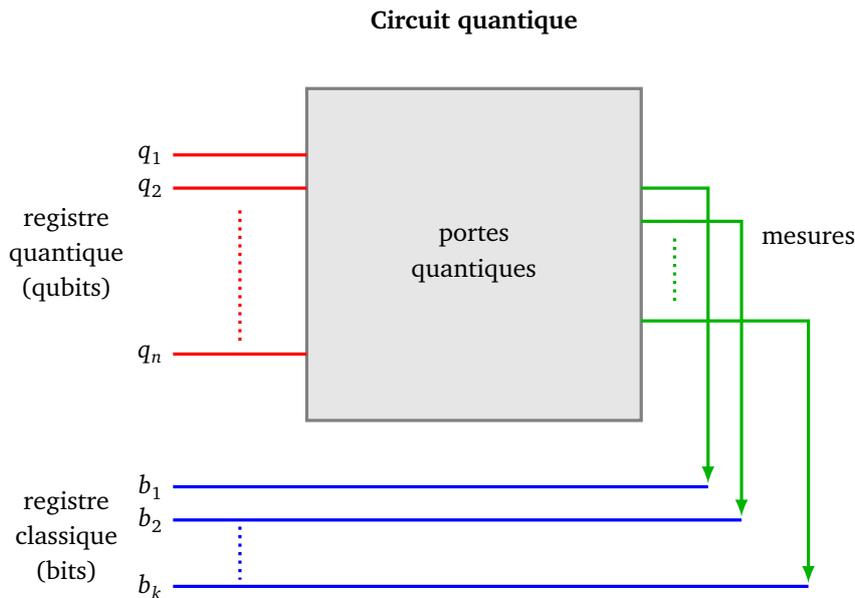
$$CNOT(|1.1\rangle) = |1.(1 \oplus 1)\rangle = |1.0\rangle.$$

4. Plus de qubits

4.1. Circuit quantique

D'une façon générale, le regroupement de plusieurs qubits conduit à un « n -qubit ». Voici le schéma de principe d'un circuit quantique :

- en entrée : n qubits dont la superposition représente un n -qubit ;
- une succession de portes quantiques, chacune agissant sur un ou plusieurs qubits ;
- le circuit est terminé par un certain nombre de mesures, qui renvoient des bits classiques.



4.2. Les n -qubits

Un n -qubit est un état quantique :

$$|\psi\rangle = \alpha_0 |0.0 \dots 0.0\rangle + \alpha_1 |0.0 \dots 0.1\rangle + \dots + \alpha_{2^n-1} |1.1 \dots 1.1\rangle.$$

- Un n -qubit possède donc 2^n coefficients. C'est toute la puissance de l'informatique quantique : la réunion de n qubits conduit à la superposition de 2^n états de base. Travailler avec un n -qubit correspond à travailler sur tous les 2^n n -bits classiques $0.0 \dots 0.0, 0.0 \dots 0.1, \dots, 1.1 \dots 1.1$ en même temps, alors que l'informatique classique ne s'occupe que d'un seul n -bit à la fois.

Par exemple, l'écriture d'un 3-qubit est la superposition de 8-états de base :

$$|\psi\rangle = \alpha_0 |0.0.0\rangle + \alpha_1 |0.0.1\rangle + \alpha_2 |0.1.0\rangle + \alpha_3 |0.1.1\rangle + \alpha_4 |1.0.0\rangle + \alpha_5 |1.0.1\rangle + \alpha_6 |1.1.0\rangle + \alpha_7 |1.1.1\rangle.$$

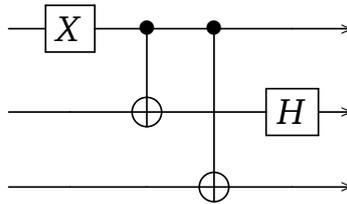
- Un n -qubit correspond donc au vecteur :

$$\begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_{2^n-1} \end{pmatrix} \in \mathbb{C}^{2^n}$$

- On impose souvent la condition de normalisation $\sum_{i=0}^{2^n-1} |\alpha_i|^2 = 1$.
- La mesure d'un n -qubit de norme 1 produit un n -bit classique : $0.0 \dots 0.0$ avec la probabilité $|\alpha_0|^2$, $0.0 \dots 0.1$ avec la probabilité $|\alpha_1|^2, \dots, 1.1 \dots 1.1$ avec la probabilité $|\alpha_{2^n-1}|^2$.

Exercice.

Voici un exemple de circuit avec 3 qubits en entrée.

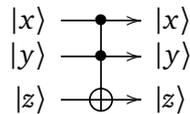


Pour chacune des entrées, correspondant à un état de base $|0.0.0\rangle, |0.0.1\rangle, \dots, |1.1.1\rangle$, calculer la sortie produite.

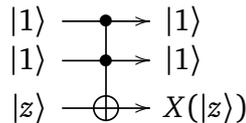
Exemple. Pour $|0.0.0\rangle$ la sortie est $\frac{1}{\sqrt{2}}|1.0.1\rangle - \frac{1}{\sqrt{2}}|1.1.1\rangle$.

Exercice.

La **porte de Toffoli** est un exemple de porte qui nécessite 3 qubits en entrée. Si l'état des deux premiers qubits est $|1\rangle$ alors la porte échange $|0\rangle$ et $|1\rangle$ pour le troisième qubit, sinon elle conserve le troisième qubit. C'est une généralisation de la porte *CNOT* qui se note aussi *CCNOT*. Autrement dit, si $(x, y) \neq (1, 1)$ alors :



Mais pour le cas particulier $x = 1$ et $y = 1$:



On suppose que les qubits en entrée sont :

- $|\psi_1\rangle = |0\rangle + |1\rangle$
- $|\psi_2\rangle = |0\rangle + 2i|1\rangle$
- $|\psi_3\rangle = 2|0\rangle - 3|1\rangle$

Calculer les trois qubits de sortie.

Indication. On pourra commencer en développant $|\psi_1\rangle \cdot |\psi_2\rangle \cdot |\psi_3\rangle$ (voir la section 3.4).

Note. La matrice associée à la porte de Toffoli est la matrice 8×8 suivante :

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

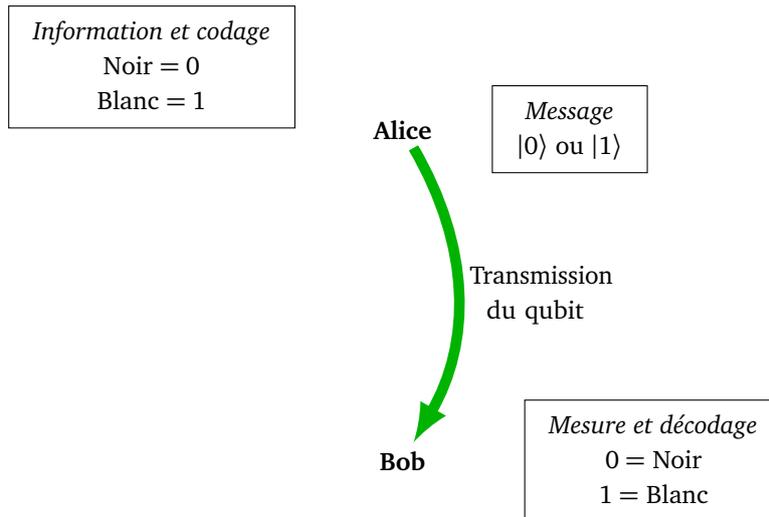
5. Communication par codage super-dense

Le codage super-dense est un protocole quantique permettant à deux personnes d'échanger de l'information.

5.1. Motivation

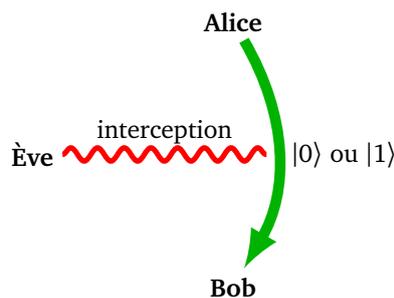
On commence par une situation très simple.

Transmission. Alice souhaite envoyer un message à Bob, par exemple « Noir » codé par 0 ou « Blanc » codé par 1. Elle peut envoyer le qubit $|0\rangle$ à Bob qui le mesure, obtient 0 et sait donc que le message est « Noir ». Si Alice envoie le qubit $|1\rangle$ à Bob, sa mesure donne 1 et le message est « Blanc ».

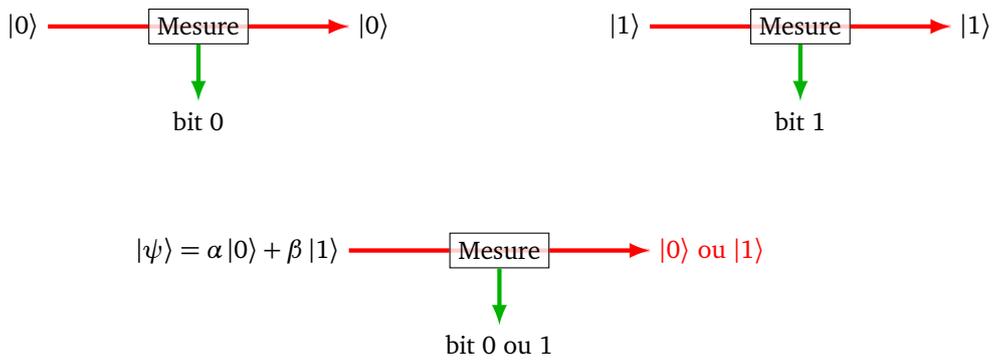


Avec cette technique, un seul bit classique d'information est transmis pour chaque qubit envoyé. Ne pourrait-on pas mieux faire ?

Interception. De plus cette technique n'est pas sûre, si l'espionne Ève intercepte le qubit transmis, alors elle peut mesurer le qubit sans changer son état. Elle récupère l'information et Bob ne s'aperçoit de rien !



En effet, mesurer le qubit $|0\rangle$ donne 0 mais ne change pas son état, idem pour le qubit $|1\rangle$. Ce ne serait pas le cas pour les autres états. Lorsque, par exemple, le qubit $|\psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ est mesuré en 0 ou 1 (une chance sur deux), il change d'état en $|0\rangle$ ou en $|1\rangle$.



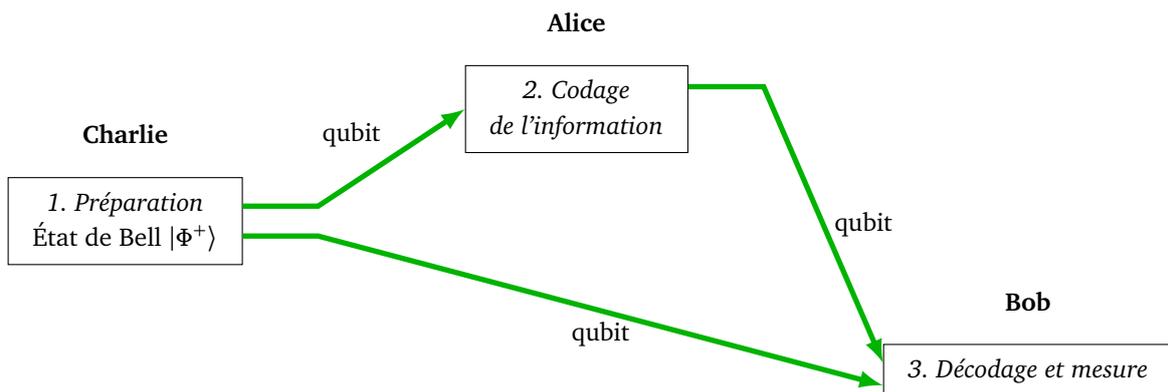
Note. Alice, Bob et Ève (pour *eavesdropper*, espionne) sont les noms habituels utilisés en cryptographie !

5.2. Schéma général du protocole

Le reste de la section est consacré au protocole appelé « codage super-dense ». Alice souhaite transmettre de façon sécurisée à Bob une information constituée de deux bits classiques, en envoyant un seul qubit.

Voici les trois étapes de ce protocole :

1. préparation de l'état de Bell,
2. codage de l'information par Alice,
3. décodage par Bob.



5.3. Préparation de l'état de Bell

Le protocole commence par un travail de préparation externe : une troisième personne, Charlie, prépare l'état de Bell.

C'est très facile : partant de l'état quantique $|0.0\rangle$, l'action d'une porte H suivi d'une porte $CNOT$ conduit à l'état de Bell :

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}|0.0\rangle + \frac{1}{\sqrt{2}}|1.1\rangle.$$

$$\begin{array}{c}
 |0\rangle \\
 \otimes \\
 |0\rangle
 \end{array}
 \xrightarrow{\begin{array}{c} \text{H} \\ \oplus \end{array}}
 \begin{array}{c}
 \frac{1}{\sqrt{2}} \begin{array}{c} |0\rangle \\ \otimes \\ |0\rangle \end{array} + \frac{1}{\sqrt{2}} \begin{array}{c} |1\rangle \\ \otimes \\ |1\rangle \end{array} = |\Phi^+\rangle
 \end{array}$$

Les calculs ont été expliqués dans la section 3.3, les voici refaits rapidement :

$$|0.0\rangle \xrightarrow{H} \left| \frac{1}{\sqrt{2}}(0+1).0 \right\rangle = \frac{1}{\sqrt{2}}(|0.0\rangle + |1.0\rangle) \xrightarrow{CNOT} \frac{1}{\sqrt{2}}(|0.0\rangle + |1.1\rangle)$$

Pour clarifier l'exposé et distinguer ce qui est à destination d'Alice et ce qui est à destination de Bob, on note l'état de Bell sous la forme :

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}|0_A.0_B\rangle + \frac{1}{\sqrt{2}}|1_A.1_B\rangle.$$

Ensuite Charlie envoie :

- un premier qubit $|\psi_A\rangle = \frac{1}{\sqrt{2}}|0_A\rangle + \frac{1}{\sqrt{2}}|1_A\rangle$ à Alice,
- un second qubit $|\psi_B\rangle = \frac{1}{\sqrt{2}}|0_B\rangle + \frac{1}{\sqrt{2}}|1_B\rangle$ à Bob.

Intrication quantique. Attention ces deux qubits $|\psi_A\rangle$ et $|\psi_B\rangle$ sont *intriqués*, c'est-à-dire liés entre eux, même une fois séparés. Si on mesure $|\psi_A\rangle$ et que l'on obtient 0, alors la mesure de $|\psi_B\rangle$ donne aussi 0 et, bien entendu, si la mesure de $|\psi_A\rangle$ donne 1 alors la mesure de $|\psi_B\rangle$ donne aussi 1.

Cela s'explique par le fait que ces deux qubits sont issus de l'état de Bell, qui lors de sa mesure ne peut conduire qu'à 0.0 ou 1.1. L'intrication quantique est un des aspects les plus troublants de la mécanique quantique. Deux particules intriquées, même distantes, continuent de partager des propriétés communes.

5.4. Transformation d'Alice

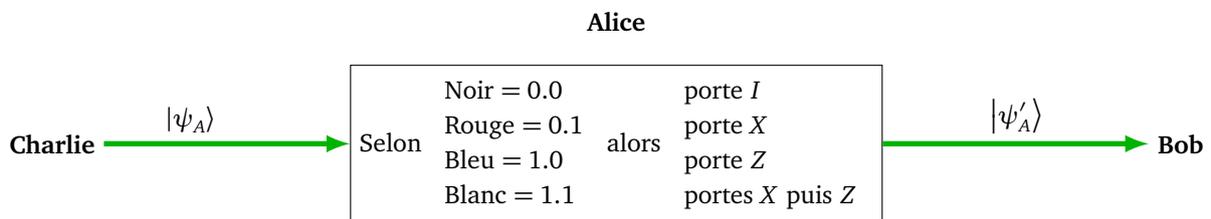
Alice souhaite envoyer un des quatre messages suivants à Bob, codés chacun par une couleur ou deux bits classiques.

- « Noir » ou 0.0,
- « Rouge » ou 0.1,
- « Bleu » ou 1.0,
- « Blanc » ou 1.1.

Elle reçoit de Charlie le qubit $|\psi_A\rangle = \frac{1}{\sqrt{2}}|0_A\rangle + \frac{1}{\sqrt{2}}|1_A\rangle$ et lui applique une des quatre transformations en fonction de l'information qu'elle souhaite transmettre :

- Si elle veut transmettre l'information « Noir/0.0 » elle applique l'identité I (elle ne fait rien et conserve $|\psi_A\rangle$).
- Si elle veut transmettre « Rouge/0.1 », elle applique la porte X à $|\psi_A\rangle$.
- Si elle veut transmettre « Bleu/1.0 », elle applique la porte Z à $|\psi_A\rangle$.
- Si elle veut transmettre « Blanc/1.1 », elle applique la porte X , suivie de la porte Z à $|\psi_A\rangle$.

Ensuite elle transmet le qubit transformé $|\psi'_A\rangle$ à Bob.



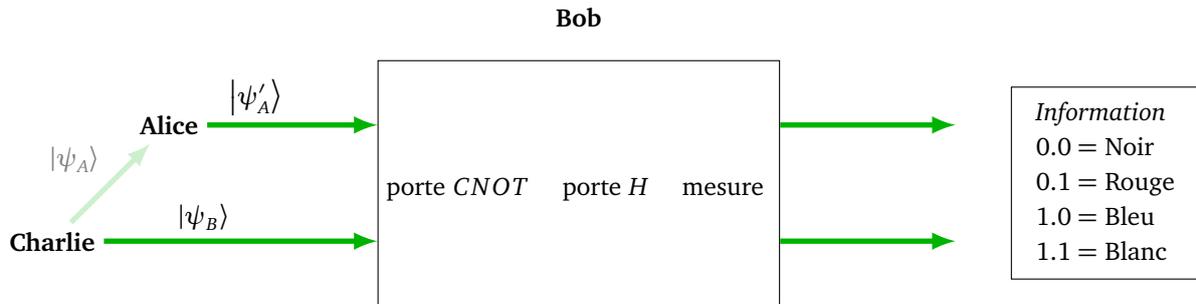
5.5. Décodage de Bob

Bob reçoit deux qubits :

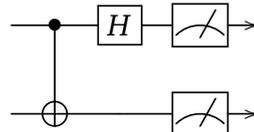
- le qubit transformé $|\psi'_A\rangle$ envoyé par Alice,
- le qubit $|\psi_B\rangle = \frac{1}{\sqrt{2}}|0_B\rangle + \frac{1}{\sqrt{2}}|1_B\rangle$ préparé par Charlie.

Mais attention, ces deux qubits sont toujours liés par intrication.

Bob a suffisamment d'informations pour retrouver le message d'Alice. Dans la pratique, il applique une porte $CNOT$ suivi d'une porte H (c'est l'opération inverse de la préparation de Charlie). Puis Bob mesure les deux qubits. Nous allons vérifier que la mesure redonne exactement l'information que voulait transmettre Alice : 0.0, 0.1, 1.0, 1.1 (pour Noir, Rouge, Bleu, Blanc).



Voici le circuit quantique du décodage de Bob :



On reprend pour chaque cas le codage d'Alice et le décodage de Bob. Ainsi Alice reçoit le qubit $|\psi_A\rangle = \frac{1}{\sqrt{2}}(|0_A\rangle + |1_A\rangle)$. Elle applique ensuite une transformation.

Cas de « Noir/0.0 ». Dans ce cas Alice ne fait rien (porte identité I sur le premier qubit), elle envoie donc directement $|\psi_A\rangle = \frac{1}{\sqrt{2}}(|0_A\rangle + |1_A\rangle)$ à Bob. Bob reçoit aussi $|\psi_B\rangle = \frac{1}{\sqrt{2}}(|0_B\rangle + |1_B\rangle)$ de Charlie. Mais n'oublions pas que ces deux qubits sont intriqués. Ainsi Bob a en main le 2-qubit $\frac{1}{\sqrt{2}}(|0_A \cdot 0_B\rangle + |1_A \cdot 1_B\rangle)$. Il applique ensuite une porte $CNOT$:

$$\frac{1}{\sqrt{2}}(|0_A \cdot 0_B\rangle + |1_A \cdot 1_B\rangle) \xrightarrow{CNOT} CNOT\left(\frac{1}{\sqrt{2}}|0_A \cdot 0_B\rangle\right) + CNOT\left(\frac{1}{\sqrt{2}}|1_A \cdot 1_B\rangle\right) = \frac{1}{\sqrt{2}}|0_A \cdot 0_B\rangle + \frac{1}{\sqrt{2}}|1_A \cdot 0_B\rangle.$$

Bob continue et applique une porte H sur le premier qubit (indexé par A) :

$$\xrightarrow{H_A} \frac{1}{\sqrt{2}} \left| \frac{1}{\sqrt{2}}(0_A + 1_A) \cdot 0_B \right\rangle + \frac{1}{\sqrt{2}} \left| \frac{1}{\sqrt{2}}(0_A - 1_A) \cdot 0_B \right\rangle = \frac{1}{2}(|0_A \cdot 0_B\rangle + |1_A \cdot 0_B\rangle + |0_A \cdot 0_B\rangle - |1_A \cdot 0_B\rangle) = |0_A \cdot 0_B\rangle.$$

Il ne reste plus que la mesure qui donne bien évidemment 0.0, ce qui est exactement le message d'Alice.

Cas de « Rouge/0.1 ». Alice applique la porte X au premier qubit de l'état de Bell, elle transforme son qubit $\frac{1}{\sqrt{2}}(|0_A\rangle + |1_A\rangle)$ en $\frac{1}{\sqrt{2}}(|1_A\rangle + |0_A\rangle)$. Mais pour l'état de Bell $\frac{1}{\sqrt{2}}(|0_A \cdot 0_B\rangle + |1_A \cdot 1_B\rangle)$ initial, cette transformation correspond au nouvel état $\frac{1}{\sqrt{2}}(|1_A \cdot 0_B\rangle + |0_A \cdot 1_B\rangle)$. Ainsi Bob reçoit le 2-qubit $|\psi\rangle = \frac{1}{\sqrt{2}}(|1_A \cdot 0_B\rangle + |0_A \cdot 1_B\rangle)$. Bob applique une porte $CNOT$, suivie d'une porte H sur le premier qubit :

$$\begin{aligned} \frac{1}{\sqrt{2}}(|1_A \cdot 0_B\rangle + |0_A \cdot 1_B\rangle) &\xrightarrow{CNOT} \frac{1}{\sqrt{2}}|1_A \cdot 1_B\rangle + \frac{1}{\sqrt{2}}|0_A \cdot 1_B\rangle \\ &\xrightarrow{H_A} \frac{1}{2} |(0_A - 1_A) \cdot 1_B\rangle + \frac{1}{2} |(0_A + 1_A) \cdot 1_B\rangle = |0_A \cdot 1_B\rangle. \end{aligned}$$

Ainsi Bob mesure 0.1 ce qui est le message d'Alice.

Cas de « Bleu/1.0 ». Alice applique la porte Z au premier qubit de l'état de Bell, Bob reçoit donc $|\psi\rangle = \frac{1}{\sqrt{2}}(|0_A \cdot 0_B\rangle - |1_A \cdot 1_B\rangle)$. Bob applique une porte $CNOT$, suivie d'une porte H sur le premier qubit :

$$\begin{aligned} \frac{1}{\sqrt{2}}(|0_A \cdot 0_B\rangle - |1_A \cdot 1_B\rangle) &\xrightarrow{CNOT} \frac{1}{\sqrt{2}}|0_A \cdot 0_B\rangle - \frac{1}{\sqrt{2}}|1_A \cdot 0_B\rangle \\ &\xrightarrow{H_A} \frac{1}{2} |(0_A + 1_A) \cdot 0_B\rangle - \frac{1}{2} |(0_A - 1_A) \cdot 0_B\rangle = |1_A \cdot 0_B\rangle. \end{aligned}$$

Ainsi Bob mesure 1.0 ce qui est le message d'Alice.

Cas de « Blanc/1.1 ». À partir de l'état de Bell, Alice applique la porte X sur le premier qubit, ce qui donne $\frac{1}{\sqrt{2}}(|1_A \cdot 0_B\rangle + |0_A \cdot 1_B\rangle)$, puis une porte Z sur le premier qubit. Ainsi Bob reçoit $|\psi\rangle = \frac{1}{\sqrt{2}}(-|1_A \cdot 0_B\rangle + |0_A \cdot 1_B\rangle)$. Bob applique une porte $CNOT$, suivie d'une porte H sur le premier qubit :

$$\begin{aligned} \frac{1}{\sqrt{2}}(-|1_A \cdot 0_B\rangle + |0_A \cdot 1_B\rangle) &\xrightarrow{CNOT} -\frac{1}{\sqrt{2}}|1_A \cdot 1_B\rangle + \frac{1}{\sqrt{2}}|0_A \cdot 1_B\rangle \\ &\xrightarrow{H_A} -\frac{1}{2} |(0_A - 1_A) \cdot 1_B\rangle + \frac{1}{2} |(0_A + 1_A) \cdot 1_B\rangle = |1_A \cdot 1_B\rangle. \end{aligned}$$

Ainsi Bob mesure 1.1 ce qui est le message d'Alice.

5.6. Bilan

Alice transmet une information composée de deux bits à Bob, mais elle ne lui a envoyé qu'un seul qubit (même si Bob reçoit globalement deux qubits). De plus c'est un protocole de transmission sécurisé. En effet, si Ève intercepte le qubit qu'Alice envoie à Bob alors elle ne peut en tirer aucune information car ce qubit est de la forme $\frac{1}{\sqrt{2}}(\pm |0\rangle \pm |1\rangle)$ et donc sa mesure donne 0 ou 1 et ne permet pas à Ève de conclure quoi que ce soit sur l'information que souhaitait transmettre Alice.

Utiliser un ordinateur quantique (avec Qiskit)

Chapitre 2

Vidéo ■ partie 2.1. Un premier circuit quantique

Vidéo ■ partie 2.2 Un qubit

Vidéo ■ partie 2.3 Deux qubits

Vidéo ■ partie 2.4 Utiliser un vrai ordinateur quantique

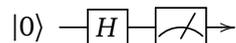
Le but est de programmer des circuits quantiques et de simuler les résultats. Mais nous allons aussi utiliser un véritable ordinateur quantique.

1. Un premier circuit quantique

On se jette l'eau et on réalise notre premier circuit quantique. Nous utilisons le langage de programmation *Python* et la librairie *qiskit* fournie par IBM.

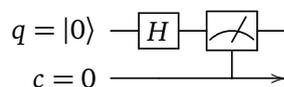
1.1. Le circuit

Il s'agit de programmer le circuit suivant.



On part donc de l'état initial $|0\rangle$, on applique une porte de Hadamard, l'état quantique devient donc $\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$. On termine par une mesure qui renvoie un bit classique 0 ou 1 avec ici chacun la probabilité $1/2$.

Pour mieux représenter la réalité de ce circuit, on l'écrit sur deux lignes. La première ligne correspond au qubit, noté q et à sa transformation, la seconde ligne correspond au bit classique, noté c , qui sert à stocker la mesure du qubit.



Remarque importante. Noter la différence avec les circuits rencontrés dans le premier chapitre : les circuits sont ici initialisés avec un état initial, $|0\rangle$ pour les qubits et 0 pour les bits classiques.

1.2. Le programme

```
import qiskit as q
from qiskit_aer import QasmSimulator
```

```
### Partie A. Préparation
```

```

# On simule un ordinateur quantique
simulator = QasmSimulator()

### Partie B. Construction du circuit

# Circuit quantique avec un qubit et une mesure
circuit = q.QuantumCircuit(1, 1)

# Une porte de Hadamard
circuit.h(0)

# Mesure du qubit (donne un bit classique)
circuit.measure(0, 0)

# Affichage du circuit
print(circuit.draw(output='text'))

### Partie C. Exécution

# Lancer de 1000 simulations
tcircuit = q.transpile(circuit, simulator)
job = simulator.run(tcircuit, shots=1000)

### Partie D. Résultats et visualisation

result = job.result()

# Comptage
counts = result.get_counts(tcircuit)
print("Nombre de '0' et de '1' :", counts)

# Diagramme en barres
import matplotlib.pyplot as plt
q.visualization.plot_histogram(counts)
plt.show()

```

1.3. Explications et résultats

On reprend pas à pas le programme ci-dessus avec des explications et les résultats.

Partie A. Préparation

- Le module *Python* à importer au préalable est le module *qiskit*, on abrège son nom par la seule lettre *q*.
- Pour l'instant on n'utilise pas un véritable ordinateur quantique, mais on transforme notre machine en un simulateur avec l'option 'QasmSimulator'.

Partie B. Construction du circuit

On commence par déclarer l'architecture du circuit :

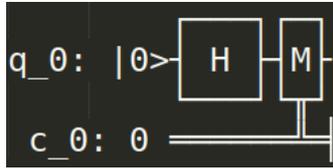
```
circuit = q.QuantumCircuit(1, 1)
```

L'instruction définit un circuit quantique, nommé *circuit*, et déclare le nombre de bits quantiques (ici 1) suivi du nombre de bits classiques (ici 1 également, pour la mesure).

On construit ensuite le circuit de la gauche vers la droite :

- on ajoute une porte H de Hadamard pour le qubit numéro 0 : `circuit.h(0)`.
- on mesure le qubit numéro 0 et on envoie le résultat sur le bit classique numéro 0 : `circuit.measure(0, 0)`.

Le programme affiche ensuite une version texte de notre circuit, ce qui permet de vérifier que tout est bien en place.



Comme l'entrée n'a pas été précisée, c'est, comme mentionné sur l'affichage ci-dessus, la valeur par défaut qui sera utilisée, à savoir $|0\rangle$.

Partie C. Exécution

On transforme notre circuit en un circuit adapté à une machine quantique via une opération appelée *transpilation* :

```
tcircuit = q.transpile(circuit, simulator)
```

Le travail de simulation commence. Un test du circuit conduit à une mesure, avec une sortie 0 ou 1 (la valeur exacte du qubit dans le circuit n'est pas accessible). Une seule valeur ne permet pas de conclure sur la nature du circuit, c'est pourquoi on effectue une simulation avec un grand nombre de lancers (*shots*).

```
job = simulator.run(tcircuit, shots=1000)
```

Partie D. Résultats et visualisation

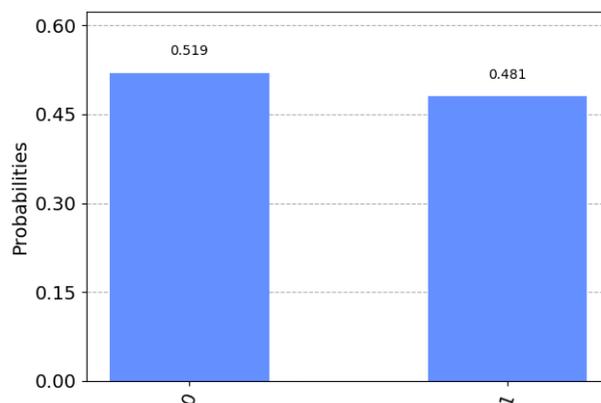
Voici un exemple de résultat renvoyé.

```
Nombre de '0' et de '1': '0': 519, '1': 481
```

Bien sûr, chaque simulation a une part d'aléatoire et conduit à des résultats différents. Cependant, selon la loi des grands nombres, par exemple avec 1000 lancers, la proportion de 0 et de 1 obtenue doit se rapprocher de la probabilité attendue. Ici les proportions de 0 et de 1 sont effectivement proches de la probabilité $\frac{1}{2}$ attendue :

$$p_0 = \frac{519}{1000} = 0.519 \quad \text{et} \quad p_1 = \frac{481}{1000} = 0.481.$$

On dispose aussi d'un affichage graphique.



En conclusion, notre circuit, qui réalise la mesure du qubit $\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$, fonctionne bien comme attendu et renvoie 0 ou 1 avec chacun une probabilité $\frac{1}{2}$.

2. Un qubit

2.1. Nombres complexes avec *Python*

Le nombre complexe $z = \frac{\sqrt{3}}{2} + \frac{1}{2}i$ s'affiche avec *Python* de la manière suivante :

```
0.8660254037844386-0.5j
```

Le nombre i est représenté par j (ou plus exactement par $1j$). Malheureusement *Python* ne fait pas de calculs exacts, il utilise des nombres flottants pour la partie réelle et pour la partie imaginaire.

Voici comment définir et afficher ce nombre complexe :

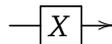
```
import numpy as np
z = np.sqrt(3)/2 + 1/2j
print(z)
print(abs(z))
print(1/z)
```

On manipule les nombres complexes comme les nombres réels : $z1+z2$, $z1*z2$, $z**2$, $1/z$.

Le module du nombre complexe z est 1, la valeur renvoyée par la fonction `abs(z)` est `0.99999...`

2.2. Circuit

Nous allons programmer un circuit encore plus simple que précédemment :



Deux différences majeures cependant :

- Cette fois nous allons initialiser l'entrée par un qubit $|\psi\rangle$ quelconque (et non plus $|0\rangle$).
- Nous utiliserons un « faux » ordinateur quantique afin d'obtenir les états quantiques. En effet un « vrai » circuit quantique n'est pas pratique pour l'apprentissage car il ne permet d'obtenir que des probabilités approchées et pas les états quantiques de qubits.

2.3. Le programme

```
from qiskit_aer import AerSimulator

### Partie A. Préparation

simulator = AerSimulator(method="statevector")

### Partie B. Construction du circuit

circuit = q.QuantumCircuit(1)

# Initialisation à la main : écriture algébrique
alpha0 = 3+1j
beta0 = 1-2j
norme = np.sqrt(abs(alpha0)**2 + abs(beta0)**2)
alpha, beta = alpha0/norme, beta0/norme
etat_initial = [alpha,beta]
qubit_initial = circuit.initialize(etat_initial, [0])
```

```
# Circuit : une porte X
circuit.x(0)

# Sauvegarder l'état du qubit
circuit.save_statevector()

### Partie C. Exécution
tcircuit = q.transpile(circuit, simulator)
job = simulator.run(tcircuit)

### Partie D. Résultats

result = job.result()

coefficients = result.get_statevector()
print("Coefficient alpha:", coefficients[0])
print("Coefficient beta :", coefficients[1])
```

2.4. Explications et résultats

- Cette fois le simulateur appelé est 'statevector_simulator', ce qui nous permet de récupérer les états quantiques en plus de leur mesure. C'est pratique pour vérifier la validité de nos circuits mais cela ne correspond pas à la réalité physique !
- Cette fois, notre circuit est défini avec un seul qubit (indexé par 0). La valeur par défaut de ce qubit est $|0\rangle$.
- Mais on va changer cette valeur, on souhaite comme qubit initial :

$$|\psi\rangle = (3 + i)|0\rangle + (1 - 2i)|1\rangle.$$

Pour cela on définit $\alpha_0 = 3 + i$, $\beta_0 = 1 - 2i$.

Afin que l'entrée soit acceptée, il faut normaliser le qubit $|\psi\rangle$. On calcule donc la norme $\|\psi\| = \sqrt{|\alpha_0|^2 + |\beta_0|^2}$. Et on définit $\alpha = \alpha_0/\|\psi\|$ et $\beta = \beta_0/\|\psi\|$. Ce qui nous permet de définir le qubit initial à l'aide de la commande `initialise([alpha,beta], [0])`.

- On rajoute une porte de Pauli X (qui échange les coefficients α et β du qubit).
- On récupère les coefficients du qubit de sortie par la fonction `get_statevector()`.
- Résultats. Notre qubit normalisé en entrée est :

$$|\psi'\rangle = (0.7746 + 0.2582i)|0\rangle + (0.2582 - 0.5164i)|1\rangle$$

La sortie obtenue :

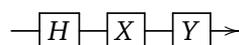
$$(0.2582 - 0.5164i)|0\rangle + (0.7746 + 0.2582i)|1\rangle$$

est bien $X(|\psi'\rangle)$.

Toutes les portes classiques sont disponibles : la porte H de Hadamard, les portes de Pauli X , Y , Z ...

Exercice.

On considère le circuit :



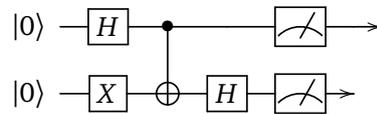
et le qubit d'entrée

$$|\psi\rangle = \frac{\sqrt{3}}{2}|0\rangle + \frac{1-i}{2\sqrt{2}}|1\rangle$$

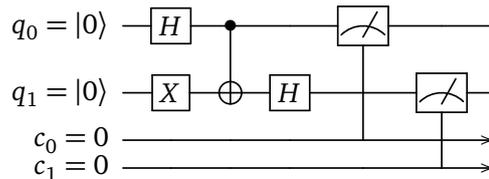
qui est un qubit de norme 1. Calculer la sortie à l'aide de la machine et vérifier vos calculs à la main.

3. Deux qubits

3.1. Le circuit



Voici une représentation plus réaliste de ce circuit, avec les deux lignes pour les bits classiques qui servent de stockage pour les mesures :



3.2. Le programme

```

### Partie A. Préparation
simulator = QasmSimulator()

### Partie B. Construction du circuit
circuit = q.QuantumCircuit(2, 2) # 2 qubits et 2 mesures

circuit.h(0) # Porte de Hadamard sur le premier qubit
circuit.x(1) # Porte X sur le second qubit
circuit.cx(0, 1) # CNOT
circuit.h(1) # Porte de Hadamard sur le second qubit
circuit.measure([0,1], [0,1]) # Mesure (q0->c0, q1->c1)

# Affichage graphique du circuit
img_circuit = circuit.draw(output='mpl')
img_circuit.show()

### Partie C. Exécution
tcircuit = q.transpile(circuit, simulator)
job = simulator.run(tcircuit, shots=1000)

### Partie D. Résultats et visualisation
result = job.result()
counts = result.get_counts(tcircuit)
print("Nombre de '00', '01', '10' et de '11':", counts)

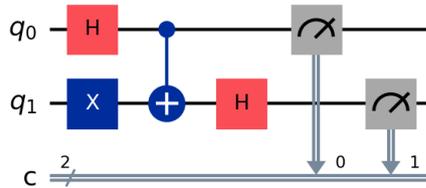
# Diagramme en barres
q.visualization.plot_histogram(counts)
plt.show()

```

3.3. Commentaires

Voici les quelques points nouveaux.

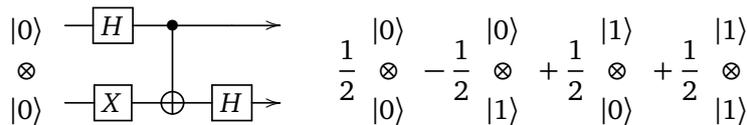
- Il faut préciser le numéro de ligne du circuit lorsque l'on ajoute une porte, par exemple `circuit.h(i)`. L'ajout se fait toujours de la gauche vers la droite.
- Pour une porte *CNOT*, il faut bien sûr préciser deux numéros de lignes.
- Pour la mesure, on précise d'abord la liste des lignes de qubits à lire et ensuite la liste des destinations.
- Ici on propose un affichage graphique du circuit construit.



3.4. Résultats

Vérifier que le qubit de sortie attendu (avant mesure) est :

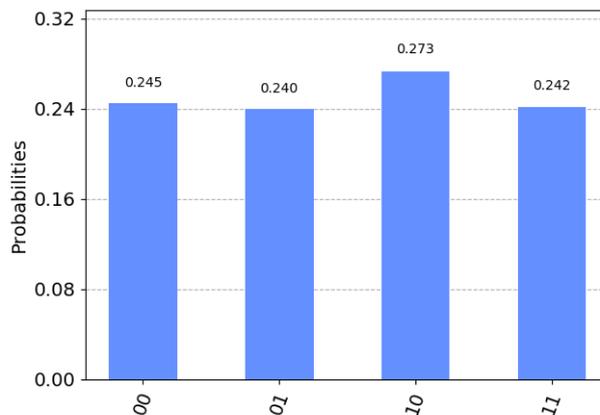
$$|\psi\rangle = \frac{1}{2} |0.0\rangle - \frac{1}{2} |0.1\rangle + \frac{1}{2} |1.0\rangle + \frac{1}{2} |1.1\rangle .$$



Expérimentalement, voici un exemple de ce que renvoie le programme :

'00': 245, '01': 240, '10': 273, '11': 242

L'affichage graphique met en évidence que chacune des 4 mesures possibles 0.0, 0.1, 1.0, 1.1 est équiprobable.

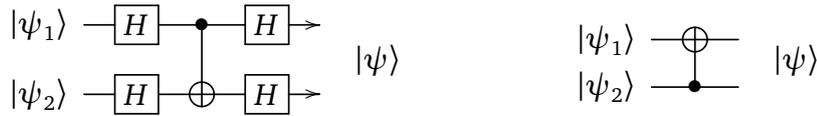


Piège! Il y a une inversion entre notre notation $|0.1\rangle$ (qui se mesure 0.1) et celle de *qiskit* '10'. De même $|1.0\rangle$ correspond à '01'. En effet, *qiskit* adopte la convention d'écriture des nombres binaires dans laquelle les bits sont écrits de droite à gauche.

Écriture d'un qubit	$ c_0.c_1 \dots c_k\rangle$
Écriture de sa mesure	$c_0.c_1 \dots c_k$
Notation <i>qiskit</i>	' $c_k \dots c_1 c_0$ '

Exercice.

On souhaite vérifier expérimentalement que les deux circuits suivants sont équivalents (des entrées égales donnent des sorties égales).



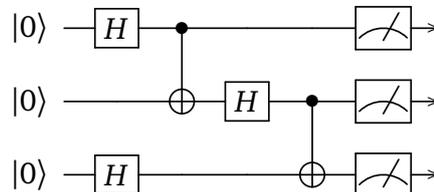
Voici les étapes.

1. Utiliser le moteur AerSimulator (method='statevector').
2. Définir au hasard un qubit de norme 1, $|\psi_1\rangle$. Pour cela, on suit la méthode utilisée dans le programme de la section 2.3 :
 - définir des nombres complexes α_0 et β_0 (choisis au hasard),
 - puis calculer la norme $n = \sqrt{|\alpha_0|^2 + |\beta_0|^2}$,
 - puis $\alpha = \alpha_0/n$ et $\beta = \beta_0/n$ définissant $|\psi_1\rangle = \alpha|0\rangle + \beta|1\rangle$ de norme 1,
 - puis utiliser la fonction Initialize() pour définir ce qubit.
3. Définir de même $|\psi_2\rangle$.
4. Définir le circuit circuit1 suivant le premier schéma et calculer le 2-qubit $|\psi\rangle$ de sortie.
5. Définir le circuit circuit2 suivant le second schéma et calculer le 2-qubit $|\psi'\rangle$ de sortie.
6. Comparer les sorties $|\psi\rangle$ et $|\psi'\rangle$.

On doit avoir $|\psi\rangle = |\psi'\rangle$ quel que soit le choix des entrées $|\psi_1\rangle$ et $|\psi_2\rangle$.

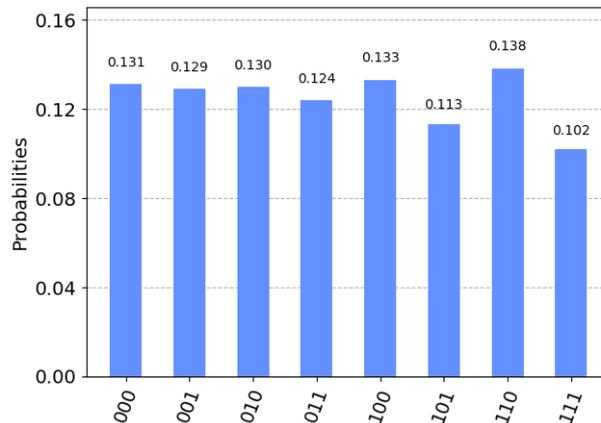
3.5. Plus de qubits

On peut bien sûr avoir davantage de qubits en entrée. Voici un exemple de circuit avec trois qubits.



```
circuit = q.QuantumCircuit(3, 3)
circuit.h(0)      # Porte de Hadamard
circuit.h(2)      # Porte de Hadamard
circuit.cx(0, 1)  # CNOT
circuit.h(1)      # Porte de Hadamard
circuit.cx(1, 2)  # CNOT
circuit.measure([0,1,2], [0,1,2]) # Mesures
```

Voici un exemple de résultat :



4. Utiliser un vrai ordinateur quantique

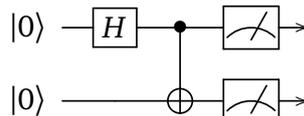
4.1. Accès à un ordinateur quantique

Un des intérêts de *qiskit* est que l'on peut exécuter ses programmes sur un ordinateur quantique ! En effet, IBM met à disposition du temps de calculs sur des véritables ordinateurs quantiques. Cet accès est gratuit et ouvert à tous. Il faut cependant s'inscrire et patienter dans une file d'attente pour lancer son programme. Voici les étapes préalables.

- Se créer un compte sur le site quantum-computing.ibm.com.
- Récupérer son code d'accès (*token*) qui est un long mot de passe du genre 'ce5a6210bb21...'. Ce code n'est nécessaire que pour la première connexion.

4.2. Programme

Voici un circuit.



Et voici le programme qui s'exécute à distance sur un vrai ordinateur quantique.

```
import qiskit as q
from qiskit_ibm_provider import IBMProvider
import matplotlib.pyplot as plt

### Partie A. Préparation

# Clé à donner une fois seulement, ensuite commenter cette ligne
IBMProvider.save_account(token='ce5a6210bb21...')

provider = IBMProvider()

print(provider.backends()) # Affiche les ordinateurs disponibles

backend = provider.get_backend('ibm_kyoto') # Choix d'un ordinateur dispo

### Partie B. Construction du circuit
```

```

circuit = q.QuantumCircuit(2, 2)
circuit.h(0)
circuit.cx(0, 1)
circuit.measure([0,1], [0,1])

### Partie C. Exécution

tcircuit = q.transpile(circuit, backend)
job = backend.run(tcircuit, shots=1000)

### Partie D. Résultats et visualisation

result = job.result()
counts = result.get_counts(tcircuit)
print("Nombre de '00', '01', '10' et de '11' :", counts)

q.visualization.plot_histogram(counts)
plt.show()

```

4.3. Explications

Partie A. Préparation. La préparation consiste à donner son code d'accès (une seule fois), à choisir un ordinateur quantique en accès (ici 'ibmq_kyoto').

Partie B. Construction du circuit. Comme d'habitude !

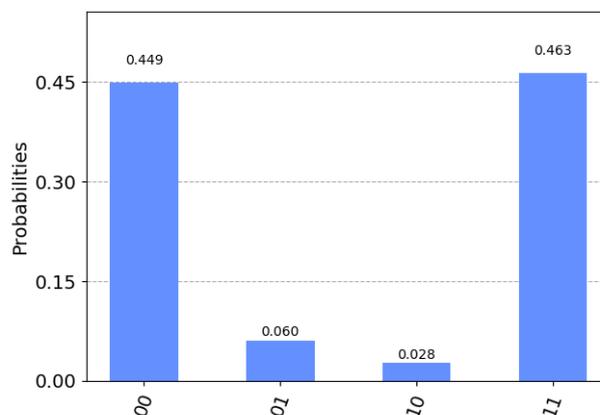
Partie C. Exécution. Presque comme d'habitude sauf qu'il faut être un peu plus patient pour disposer de l'accès et attendre la fin des calculs. On retrouve aussi les résultats sur la page de son compte.

4.4. Résultats

Voici un exemple de résultats sous forme numérique :

'00': 449, '01': 60, '10': 28, '11': 463

et sous forme graphique :



Noter que l'état quantique de sortie (avant mesure) est $|\Phi^+\rangle = \frac{1}{\sqrt{2}}|0.0\rangle + \frac{1}{\sqrt{2}}|1.1\rangle$. Les résultats devraient donc être uniquement mesurés en 0.0 ou 1.1 (avec des probabilités proches de 1/2). Mais sur un véritable

ordinateur quantique il y a des erreurs, qui ici produisent certaines mesures impossibles en théorie (ici 0.1 et 1.0).

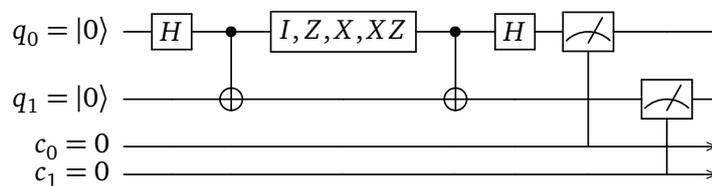
On distingue deux causes qui font que l'on n'obtient pas exactement moitié/moitié pour les mesures 0.0 et 1.1 :

- une cause probabiliste : il peut y avoir un écart entre les mesures et les probabilités théoriques attendues, écart dû au caractère aléatoire d'une mesure (comme le lancer d'une pièce de monnaie). Avec un grand nombre de lancers, cet écart diminue.
- une cause d'erreur physique : un ordinateur quantique n'est pas parfait, il peut y avoir des erreurs.

5. Codage super-dense

5.1. Circuit

Il s'agit de transmettre une information classique composée de deux bits 0.0 ou 0.1 ou 1.0 ou 1.1 en transmettant un seul qubit. On renvoie à la fin du chapitre « Découverte de l'informatique quantique » pour les explications. On rappelle le circuit en jeu :



5.2. Programme

```
import qiskit as q
from qiskit_aer import QasmSimulator

### Partie A. Préparation

simulator = QasmSimulator()

### Partie B. Construction du circuit

circuit = q.QuantumCircuit(2, 2)

## B.1 Préparation de l'état de Bell
circuit.h(0) # Porte de Hadamard
circuit.cx(0, 1) # CNOT

message_alice = '01' # choix entre '00', '01', '10', '11'

## B.2 Porte d'Alice selon message à transmettre
if message_alice == '00':
    circuit.iden(0) # identité
elif message_alice == '01':
    circuit.z(0) # porte Z
elif message_alice == '10':
    circuit.x(0) # porte X
elif message_alice == '11':
```

```

circuit.x(0)      # porte X
circuit.z(0)      # suivi de porte Z

## B.3 Décodage
circuit.cx(0, 1) # CNOT
circuit.h(0)     # Porte de Hadamard

## B.4 Mesures
circuit.measure([0,1], [0,1]) # Mesures

print(circuit.draw(output='text'))

### Partie C. Exécution

# Lancer de 1000 simulations
tcircuit = q.transpile(circuit, simulator)
job = simulator.run(tcircuit, shots=1000)

# Partie D. Résultats

result = job.result()

# Comptage
counts = result.get_counts(tcircuit)
print("Nombre de '00', '01', '10' '11' :", counts)

```

5.3. Résultats

Le programme s'utilise en choisissant le message qu'Alice envoie à Bob, par exemple `message_alice = '01'`. Dans ce cas, Alice ajoute une porte Z au circuit. La sortie mesurée est alors '01' (dans 100% des simulations). Testez les autres messages !

On rappelle la subtilité : si la mesure du qubit q_0 est 1 et celle du qubit q_1 est 0 alors on note globalement la mesure 1.0 (correspondant à l'état quantique $|1.0\rangle$) mais *qiskit* écrit cette même mesure '01' (de la droite vers la gauche). Faites un choix pour votre programme et tenez-vous-y !

Nombres complexes

Vidéo ■ partie 3.1. Écriture algébrique

Vidéo ■ partie 3.2. Qubit

Vidéo ■ partie 3.3. Module - Argument

Vidéo ■ partie 3.4. Écriture trigonométrique des qubits

Vidéo ■ partie 3.5. Sphère de Bloch

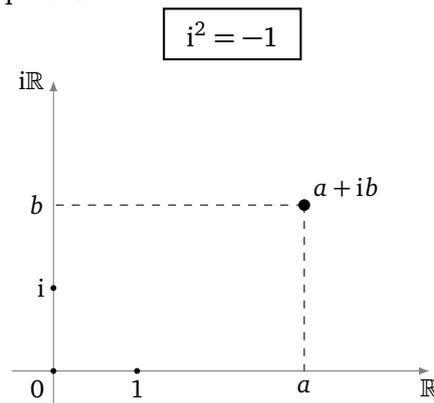
Les nombres complexes sont les coefficients naturels des qubits. Nous détaillons les calculs avec les nombres complexes ainsi que sur les qubits.

1. Écriture algébrique

Les nombres complexes étendent les nombres réels de façon à pouvoir résoudre les équations du type $x^2 = -1$.

1.1. Définition

- Un **nombre complexe** est un couple $(a, b) \in \mathbb{R}^2$ que l'on notera $a + ib$.
- Exemple avec $a = 2$ et $b = 3$: $z = 2 + 3i$.
- Le nombre complexe i vérifie l'équation :



- **Addition.** $(a + ib) + (a' + ib') = (a + a') + i(b + b')$
- **Multiplication :** $(a + ib) \times (a' + ib') = (aa' - bb') + i(ab' + ba')$. Ainsi on développe, en suivant les règles usuelles de la multiplication et en utilisant la règle $i^2 = -1$.

Exemple.

Soit $z_1 = 2 + 3i$ et $z_2 = 5 - 4i$.

Alors

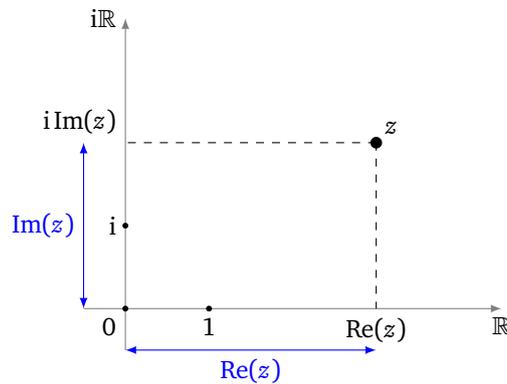
$$z_1 + z_2 = (2 + 3i) + (5 - 4i) = 7 - i.$$

Et

$$\begin{aligned}
 z_1 \times z_2 &= (2 + 3i) \times (5 - 4i) \\
 &= 10 - 8i + 15i - 12i^2 \\
 &= 10 - 8i + 15i + 12 \\
 &= 22 + 7i.
 \end{aligned}$$

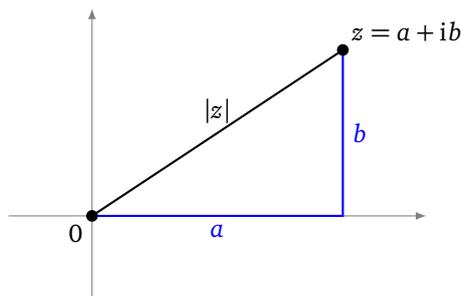
1.2. Partie réelle et imaginaire

Soit $z = a + ib$ un nombre complexe, sa *partie réelle* est le réel a et on la note $\operatorname{Re}(z)$; sa *partie imaginaire* est le réel b et on la note $\operatorname{Im}(z)$.



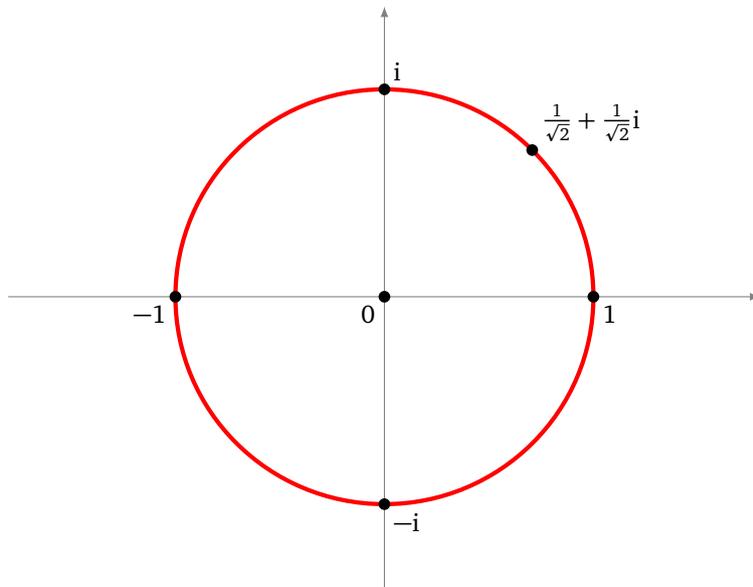
1.3. Module

Module. Le *module* de $z = a + ib$ est le réel positif $|z| = \sqrt{a^2 + b^2}$. Il mesure la distance du point (a, b) à l'origine $(0, 0)$.



Exemple : $|5 - 2i| = \sqrt{5^2 + (-2)^2} = \sqrt{29}$.

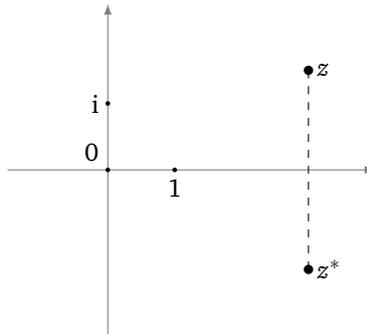
Nombres complexes de module 1. On peut représenter l'ensemble des nombres complexes de module 1 par le cercle de rayon 1 centré à l'origine.



Exemples : 1, i et $\frac{1}{\sqrt{2}} + \frac{1}{\sqrt{2}}i$ sont des nombres complexes de module 1.

On peut transformer un nombre complexe quelconque (non nul) en un nombre complexe de module 1 en le divisant par son module. Par exemple $z = 5 - 2i$ a pour module $|z| = \sqrt{29}$, donc $\frac{z}{|z|} = \frac{5}{\sqrt{29}} - \frac{2}{\sqrt{29}}i$ est de module 1.

Conjugué. Le *conjugué* de $z = a + ib$ est $z^* = a - ib$, autrement dit $\text{Re}(z^*) = \text{Re}(z)$ et $\text{Im}(z^*) = -\text{Im}(z)$. Le point z^* est le symétrique du point z par rapport à l'axe réel. Comme $z \times z^* = (a + ib)(a - ib) = a^2 + b^2$ alors le module vaut aussi $|z| = \sqrt{zz^*}$.



Notation. Une écriture plus classique pour le conjugué est \bar{z} , mais nous préférons ici la notation z^* plus adaptée pour la suite du cours.

Inverse. L'*inverse* : si $z \neq 0$, il existe un unique $z' \in \mathbb{C}$ tel que $zz' = 1$ (où $1 = 1 + i \times 0$).

$$z' = \frac{1}{z} = \frac{a - ib}{a^2 + b^2} = \frac{z^*}{|z|^2}$$

2. Qubit

2.1. Définition

Rappelons la définition des qubits à partir des deux *états quantiques de base* $|0\rangle$ et $|1\rangle$. Un *1-qubit*, appelé aussi simplement *qubit*, est un *état quantique* obtenu par combinaison linéaire :

$$|\psi\rangle = \alpha |0\rangle + \beta |1\rangle \quad \text{avec } \alpha \in \mathbb{C} \quad \text{et} \quad \beta \in \mathbb{C}$$

avec souvent la condition de normalisation :

$$|\alpha|^2 + |\beta|^2 = 1.$$

Un qubit est donc défini par deux nombres complexes, $\alpha = a_1 + ib_1$ et $\beta = a_2 + ib_2$. Il faut ainsi 4 nombres réels a_1, b_1, a_2, b_2 pour définir un qubit.

Deux qubits réunis sont dans un état quantique $|\psi\rangle$, appelé **2-qubit**, défini par la superposition :

$$|\psi\rangle = \alpha |0.0\rangle + \beta |0.1\rangle + \gamma |1.0\rangle + \delta |1.1\rangle \quad \text{avec } \alpha, \beta, \gamma, \delta \in \mathbb{C}$$

avec souvent la convention de normalisation :

$$|\alpha|^2 + |\beta|^2 + |\gamma|^2 + |\delta|^2 = 1.$$

Il faudrait donc 8 nombres réels pour définir un 2-qubit.

2.2. Opérations

Addition. L'addition de deux qubits se fait coefficient par coefficient, il s'agit donc d'additionner des paires de nombres complexes. Par exemple si

$$|\phi\rangle = (1 + 3i)|0\rangle + 2i|1\rangle \quad \text{et} \quad |\psi\rangle = 3|0\rangle + (1 - i)|1\rangle$$

alors

$$|\phi\rangle + |\psi\rangle = (4 + 3i)|0\rangle + (1 + i)|1\rangle.$$

Ou encore pour des 2-qubits :

$$(|1.0\rangle + |0.1\rangle) + (|1.0\rangle - |0.1\rangle) = 2|1.0\rangle.$$

Multiplication. On peut multiplier deux 1-qubits pour obtenir un 2-qubit. Les calculs se font comme des calculs algébriques à l'aide des règles de bases $|0\rangle \cdot |0\rangle = |0.0\rangle$, $|0\rangle \cdot |1\rangle = |0.1\rangle$,... Pour les coefficients, on utilise la multiplication des nombres complexes, avec bien sûr toujours la relation $i^2 = -1$.

Par exemple avec

$$|\phi\rangle = (1 + 3i)|0\rangle + 2i|1\rangle \quad \text{et} \quad |\psi\rangle = 3|0\rangle + (1 - i)|1\rangle$$

on a

$$\begin{aligned} |\phi\rangle \cdot |\psi\rangle &= ((1 + 3i)|0\rangle + 2i|1\rangle) \times (3|0\rangle + (1 - i)|1\rangle) \\ &= (1 + 3i) \cdot 3 \cdot |0\rangle \cdot |0\rangle + (1 + 3i) \cdot (1 - i) \cdot |0\rangle \cdot |1\rangle + 2i \cdot 3 \cdot |1\rangle \cdot |0\rangle + 2i \cdot (1 - i) \cdot |1\rangle \cdot |1\rangle \\ &= (3 + 9i)|0.0\rangle + (4 + 2i)|0.1\rangle + 6i|1.0\rangle + (2 + 2i)|1.1\rangle \end{aligned}$$

où on a utilisé $(1 + 3i) \cdot (1 - i) = 1 - i + 3i - 3i^2 = 4 + 2i$ et $2i \cdot (1 - i) = 2i - 2i^2 = 2 + 2i$.

2.3. Norme

Norme. La norme d'un qubit est un nombre réel $\|\psi\|$.

- Pour un qubit $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, $\|\psi\| = \sqrt{|\alpha|^2 + |\beta|^2}$ est sa norme.
- Pour un 2-qubit $|\psi\rangle = \alpha|0.0\rangle + \beta|0.1\rangle + \gamma|1.0\rangle + \delta|1.1\rangle$, sa norme est $\|\psi\| = \sqrt{|\alpha|^2 + |\beta|^2 + |\gamma|^2 + |\delta|^2}$.
- La normalisation d'un qubit $|\psi\rangle$ est $\frac{|\psi\rangle}{\|\psi\|}$, qui est un qubit de norme 1.

Exemple : pour $|\psi\rangle = (3 + 4i)|0\rangle + (2 - i)|1\rangle$ alors la norme au carré vaut :

$$\begin{aligned} \|\psi\|^2 &= |3 + 4i|^2 + |2 - i|^2 \\ &= (3^2 + 4^2) + (2^2 + (-1)^2) \\ &= 30. \end{aligned}$$

Donc $\|\psi\| = \sqrt{30}$.

Exercice.

Vérifier que la norme de

$$|\psi\rangle = (1 + i)|0.0\rangle + (1 - 2i)|0.1\rangle + (3 - 4i)|1.0\rangle + 2i|1.1\rangle$$

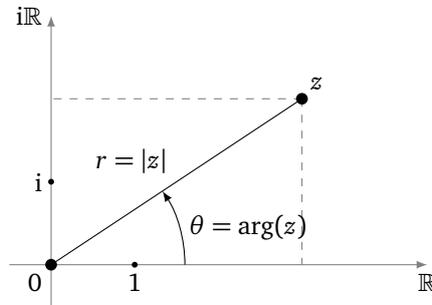
est $\|\psi\| = 6$. Que vaut la normalisation de $|\psi\rangle$?

3. Écriture trigonométrique

3.1. Module et argument

Un nombre complexe $z \in \mathbb{C}$, admet l'écriture trigonométrique :

$$z = r \cos \theta + ir \sin \theta \quad \text{avec} \quad r \in \mathbb{R}_+ \quad \text{et} \quad \theta \in \mathbb{R}$$



- r est en fait le module de z : $r = |z|$,
- θ est un **argument** de z , on le note $\arg(z)$.

L'argument n'est pas unique : si θ est un argument alors $\theta + 2k\pi$ ($k \in \mathbb{Z}$) aussi. Pour rendre l'argument unique, on peut imposer la condition $\theta \in]-\pi, \pi]$ (ou encore $\theta \in [0, 2\pi[$). Si on impose $\theta \in]-\pi, \pi]$ alors pour un nombre complexe z non nul, l'écriture $z = r \cos \theta + ir \sin \theta$ est unique.

On dira que $\arg(z)$ est « défini modulo 2π » et l'écriture $\theta \equiv \theta' \pmod{2\pi}$ signifie que $\theta = \theta' + 2k\pi$ pour un certain entier $k \in \mathbb{Z}$.

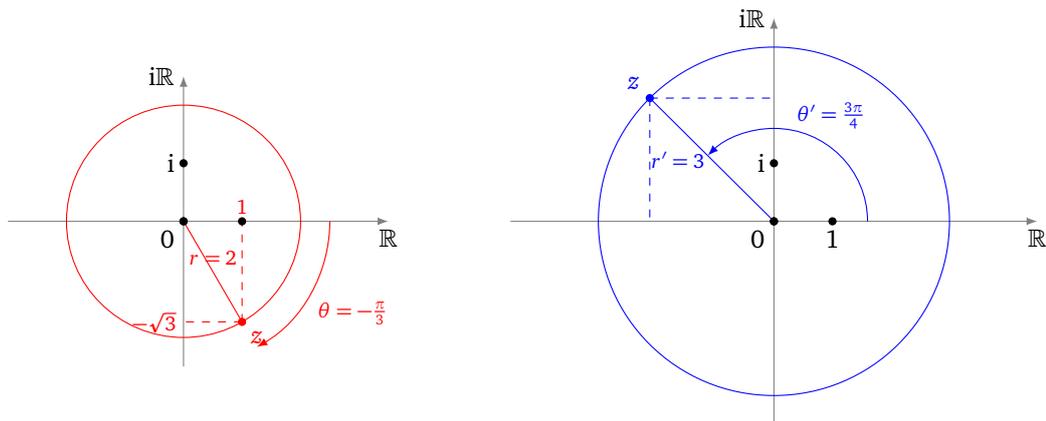
Exemple.

- Soit $z = 1 - \sqrt{3}i$. Alors $r = |z| = 2$ et $\theta = -\frac{\pi}{3}$, car alors

$$r \cos \theta = 2 \cos\left(-\frac{\pi}{3}\right) = 2 \times \frac{1}{2} = 1 = \operatorname{Re}(z)$$

et

$$r \sin \theta = 2 \sin\left(-\frac{\pi}{3}\right) = -2 \times \frac{\sqrt{3}}{2} = -\sqrt{3} = \operatorname{Im}(z).$$



- Le nombre complexe de module $r = 3$ et d'argument $\theta = \frac{3\pi}{4}$ est

$$z' = r \cos \theta + ir \sin \theta = -\frac{3\sqrt{2}}{2} + \frac{3\sqrt{2}}{2}i = \frac{3\sqrt{2}}{2}(-1 + i).$$

L'écriture module-argument facilite le calcul des multiplications. Les modules se multiplient, les arguments s'additionnent.

Proposition 1.

Soient z et z' deux nombres complexes. Alors

$$|zz'| = |z| \cdot |z'| \quad \text{et} \quad \arg(zz') \equiv \arg(z) + \arg(z') \pmod{2\pi}$$

Démonstration.

$$\begin{aligned} zz' &= |z|(\cos \theta + i \sin \theta) |z'|(\cos \theta' + i \sin \theta') \\ &= |zz'|(\cos \theta \cos \theta' - \sin \theta \sin \theta' + i(\cos \theta \sin \theta' + \sin \theta \cos \theta')) \\ &= |zz'|(\cos(\theta + \theta') + i \sin(\theta + \theta')) \end{aligned}$$

donc $|zz'| = |z| \cdot |z'|$ et $\arg(zz') \equiv \arg(z) + \arg(z') \pmod{2\pi}$. □

3.2. Notation exponentielle

Nous définissons la *notation exponentielle* par

$$e^{i\theta} = \cos \theta + i \sin \theta$$

et donc tout nombre complexe s'écrit :

$$z = r e^{i\theta}$$

où $r = |z|$ est son module et $\theta = \arg(z)$ est un de ses arguments.

Exemples : $e^{i\frac{\pi}{2}} = i$, $e^{i\pi} = -1$, $e^{2i\pi} = e^0 = 1$.

Avec la notation exponentielle, les calculs s'effectuent avec les lois habituelles pour les puissances. Par exemple :

$$(e^{i\theta})^n = e^{in\theta}$$

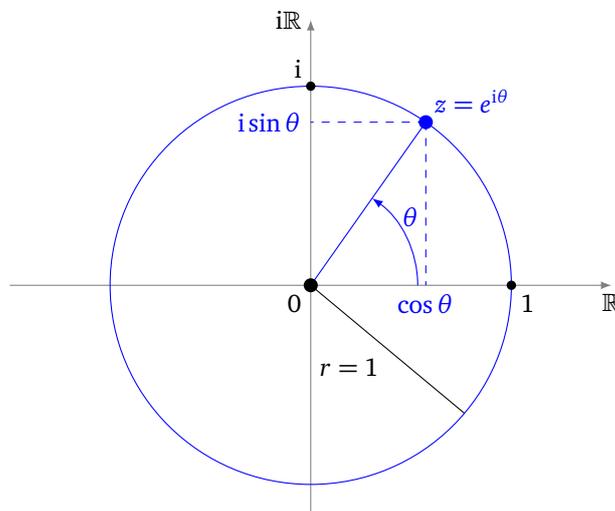
Il s'agit en fait de la *formule de Moivre* qui s'écrit en version étendue :

$$(\cos \theta + i \sin \theta)^n = \cos(n\theta) + i \sin(n\theta).$$

De façon plus générale, pour $z = r e^{i\theta}$ et $z' = r' e^{i\theta'}$, on peut écrire :

- $zz' = r r' e^{i\theta} e^{i\theta'} = r r' e^{i(\theta+\theta')}$
- $z^n = (r e^{i\theta})^n = r^n (e^{i\theta})^n = r^n e^{in\theta}$
- $1/z = 1/(r e^{i\theta}) = \frac{1}{r} e^{-i\theta}$
- $z^* = r e^{-i\theta}$

Tout nombre complexe de module 1 s'écrit sous la forme $z = e^{i\theta}$, autrement dit $z = \cos \theta + i \sin \theta$.



4. Écriture trigonométrique des qubits

4.1. Écriture des qubits

À l'aide de la notation exponentielle, un qubit $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$ peut aussi s'écrire :

$$|\psi\rangle = r e^{i\theta} |0\rangle + r' e^{i\theta'} |1\rangle.$$

Un tel qubit est normalisé si $r^2 + r'^2 = 1$.

Certains utilisent un vocabulaire issu de la physique :

- θ est la **phase** associée à $|0\rangle$,
- θ' est la phase associée à $|1\rangle$.

L'écriture algébrique est adaptée à un calcul de somme tandis que la notation exponentielle rend le calcul d'une multiplication plus facile.

Exemple.

Si $|\phi\rangle = 2e^{i\frac{\pi}{3}} |0\rangle + 3e^{i\frac{\pi}{4}} |1\rangle$ et $|\psi\rangle = 4e^{i\frac{\pi}{5}} |0\rangle + 5e^{i\frac{\pi}{6}} |1\rangle$. Alors :

$$\begin{aligned} |\phi\rangle \cdot |\psi\rangle &= \left(2e^{i\frac{\pi}{3}} |0\rangle + 3e^{i\frac{\pi}{4}} |1\rangle\right) \times \left(4e^{i\frac{\pi}{5}} |0\rangle + 5e^{i\frac{\pi}{6}} |1\rangle\right) \\ &= 2e^{i\frac{\pi}{3}} \cdot 4e^{i\frac{\pi}{5}} |0.0\rangle + 2e^{i\frac{\pi}{3}} \cdot 5e^{i\frac{\pi}{6}} |0.1\rangle + 3e^{i\frac{\pi}{4}} \cdot 4e^{i\frac{\pi}{5}} |1.0\rangle + 3e^{i\frac{\pi}{4}} \cdot 5e^{i\frac{\pi}{6}} |1.1\rangle \\ &= 8e^{i(\frac{\pi}{3}+\frac{\pi}{5})} |0.0\rangle + 10e^{i(\frac{\pi}{3}+\frac{\pi}{6})} |0.1\rangle + 12e^{i(\frac{\pi}{4}+\frac{\pi}{5})} |1.0\rangle + 15e^{i(\frac{\pi}{4}+\frac{\pi}{6})} |1.1\rangle. \end{aligned}$$

4.2. Équivalence de qubits

La mesure physique d'un qubit $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$ ne permet pas d'accéder aux valeurs de α et β . Par exemple $|0\rangle + |1\rangle$ et $2|0\rangle + 2|1\rangle$ ne pourront pas être distingués par des mesures, ils donnent tous les deux une mesure 0 ou 1 avec probabilité 1/2.

On dit que deux états sont **équivalents** si on peut passer de l'un à l'autre par les opérations suivantes :

- multiplication par une constante réelle :

$$k |\psi\rangle \equiv |\psi\rangle \quad \text{où } k \in \mathbb{R}^*,$$

- multiplication par $e^{i\theta}$ (un nombre complexe de module 1) :

$$e^{i\theta} |\psi\rangle \equiv |\psi\rangle \quad \text{où } \theta \in \mathbb{R}.$$

Une reformulation est de dire que deux qubit $|\phi\rangle$ et $|\psi\rangle$ sont **équivalents** s'il existe $z \in \mathbb{C}^*$ tel que $|\phi\rangle = z |\psi\rangle$.

Deux états quantiques équivalents ne peuvent pas être distingués par des mesures.

Exemple.

- Par exemple

$$|0\rangle + |1\rangle \equiv \frac{1}{\sqrt{2}} |0\rangle + \frac{1}{\sqrt{2}} |1\rangle.$$

On passe de l'un à l'autre en multipliant par $k = 1/\sqrt{2}$.

- On a aussi

$$i |0\rangle + (1 - i) |1\rangle \equiv -|0\rangle + (1 + i) |1\rangle$$

On passe de l'un à l'autre en multipliant par $i = e^{i\frac{\pi}{2}}$.

- On peut combiner les deux opérations :

$$(1 + 2i) |0\rangle + i |1\rangle \equiv |0\rangle + \frac{2+i}{5} |1\rangle$$

On passe de l'un à l'autre en multipliant par $z = \frac{1-2i}{5}$. Les deux qubits équivalents $(1 + 2i)|0\rangle + i|1\rangle$ et $|0\rangle + \frac{2+i}{5}|1\rangle$ conduisent tous les deux lors d'une mesure à 0 avec une probabilité $\frac{5}{6}$ et à 1 avec une probabilité $\frac{1}{6}$.

Remarque.

- Attention, deux états équivalents ne sont pas égaux! Il ne faut pas les interchanger dans les calculs intermédiaires. Cependant, lors de la mesure finale, on peut remplacer un état par un état équivalent sans changer le résultat.
- En effet, les deux opérations élémentaires qui définissent l'équivalence ne changent pas le calcul de probabilité pour la mesure.
- L'équivalence des qubits évite en particulier de parler de normalisation.

Proposition 2.

Un qubit $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ non nul (c'est-à-dire $(\alpha, \beta) \neq (0, 0)$) est équivalent à un qubit de norme 1 de la forme :

$$|\psi\rangle \equiv \cos\left(\frac{\theta}{2}\right)|0\rangle + \sin\left(\frac{\theta}{2}\right)e^{i\phi}|1\rangle$$

De plus l'écriture est unique lorsque l'on a les conditions $0 < \theta < \pi$ et $-\pi < \phi \leq \pi$.

Exemple.

- $|\psi\rangle = i\sqrt{2}|0\rangle + \sqrt{3}(1+i)|1\rangle$. On commence par rendre le coefficient devant $|0\rangle$ réel positif. Pour cela on multiplie tout par $-i = e^{-i\frac{\pi}{2}}$:

$$|\psi\rangle \equiv (-i)(i\sqrt{2}|0\rangle + \sqrt{3}(1+i)|1\rangle) = \sqrt{2}|0\rangle + \sqrt{3}(1-i)|1\rangle.$$

La norme de ce dernier qubit est $2\sqrt{2}$, on divise donc par cette norme. Ainsi :

$$|\psi\rangle \equiv \frac{1}{2}|0\rangle + \frac{\sqrt{3}}{2}\frac{1-i}{\sqrt{2}}|1\rangle.$$

On pose d'une part $\theta = \frac{2\pi}{3}$ pour lequel $\cos\left(\frac{\theta}{2}\right) = \frac{1}{2}$ et $\sin\left(\frac{\theta}{2}\right) = \frac{\sqrt{3}}{2}$ et d'autre part $\phi = -\frac{\pi}{4}$ pour lequel $e^{i\phi} = \frac{1-i}{\sqrt{2}}$.

- Il n'est en général pas possible d'expliciter les angles θ et ϕ . Considérons par exemple $|\psi\rangle = 1|0\rangle - 2i|1\rangle$. Alors $\|\psi\| = \sqrt{5}$ et $|\psi\rangle \equiv \frac{1}{\sqrt{5}}|0\rangle - \frac{2}{\sqrt{5}}i|1\rangle$. On sait qu'il existe un réel θ tel que $\cos\left(\frac{\theta}{2}\right) = \frac{1}{\sqrt{5}}$ et $\sin\left(\frac{\theta}{2}\right) = \frac{2}{\sqrt{5}}$, ce θ est défini par $\frac{\theta}{2} = \arccos\left(\frac{1}{\sqrt{5}}\right)$ mais n'a pas d'expression plus explicite. On pose alors $\phi = -\frac{\pi}{2}$ qui vérifie $e^{i\phi} = -i$. Ces θ et ϕ conviennent.

Démonstration.

Existence. On commence par transformer le coefficient de $|0\rangle$ en un réel positif. Si $\alpha = re^{i\theta}$ alors

$$\begin{aligned} |\psi\rangle &\equiv e^{-i\theta}|\psi\rangle \\ &= e^{-i\theta}(re^{i\theta}|0\rangle + \beta|1\rangle) \\ &= r|0\rangle + \beta \cdot e^{-i\theta}|1\rangle. \end{aligned}$$

On normalise ensuite ce qubit en divisant par $\|\psi\|$:

$$\begin{aligned} |\psi\rangle &\equiv \frac{1}{\|\psi\|} |\psi\rangle \\ &\equiv \frac{1}{\|\psi\|} (r |0\rangle + \beta \cdot e^{-i\theta} |1\rangle) \\ &= \frac{r}{\|\psi\|} |0\rangle + \frac{\beta \cdot e^{-i\theta}}{\|\psi\|} |1\rangle. \end{aligned}$$

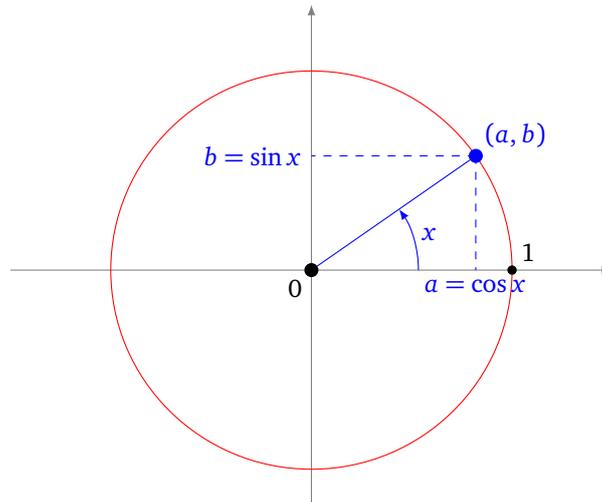
Ce dernier qubit s'écrit :

$$|\psi'\rangle = r' |0\rangle + \beta' |1\rangle$$

avec $r' \in \mathbb{R}$ et $\beta' \in \mathbb{C}$. Mais comme par définition $|\psi'\rangle$ est un qubit de module 1, on a de plus $r'^2 + |\beta'|^2 = 1$ et en particulier $0 \leq r' \leq 1$ et $|\beta'| \leq 1$.

Rappel. Pour deux nombres réels $a, b \geq 0$ vérifiant $a^2 + b^2 = 1$, il existe $x \in [0, \frac{\pi}{2}]$ tel que

$$\begin{cases} a = \cos x \\ b = \sin x \end{cases}$$



On applique le rappel à r' et $|\beta'|$ afin d'obtenir $x = \frac{\theta}{2}$ ($\theta \in [0, \pi]$), et d'écrire $r' = \cos\left(\frac{\theta}{2}\right)$ et $|\beta'| = \sin\left(\frac{\theta}{2}\right)$. Finalement, $\beta' = \sin\left(\frac{\theta}{2}\right) e^{i\phi}$ pour un certain argument $\phi \in \mathbb{R}$. Ainsi $|\psi\rangle$ est bien équivalent à un qubit de la forme souhaitée :

$$|\psi\rangle \equiv |\psi'\rangle = \cos\left(\frac{\theta}{2}\right) |0\rangle + \sin\left(\frac{\theta}{2}\right) e^{i\phi} |1\rangle.$$

On aurait pu effectuer toutes les opérations en une seule fois. En effet, si $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$ avec $\alpha = r e^{i\theta}$ alors $\frac{e^{-i\theta}}{\|\psi\|} |\psi\rangle$ est de la forme voulue.

Unicité. L'unicité découle de la construction. On peut aussi la prouver de la façon suivante. Si on suppose qu'il existe (θ, ϕ) et (θ', ϕ') tels que

$$\cos\left(\frac{\theta}{2}\right) |0\rangle + \sin\left(\frac{\theta}{2}\right) e^{i\phi} |1\rangle = \cos\left(\frac{\theta'}{2}\right) |0\rangle + \sin\left(\frac{\theta'}{2}\right) e^{i\phi'} |1\rangle$$

alors, par identification, les coefficients devant $|0\rangle$ sont égaux, de même pour les coefficients devant $|1\rangle$.

Ainsi

$$\begin{cases} \cos\left(\frac{\theta}{2}\right) = \cos\left(\frac{\theta'}{2}\right) \\ \sin\left(\frac{\theta}{2}\right) e^{i\phi} = \sin\left(\frac{\theta'}{2}\right) e^{i\phi'} \end{cases}$$

Mais comme $0 < \frac{\theta}{2} < \frac{\pi}{2}$ et $0 < \frac{\theta'}{2} < \frac{\pi}{2}$ alors $\cos(\frac{\theta}{2}) = \cos(\frac{\theta'}{2})$ implique $\theta = \theta'$. On en déduit donc que $\sin(\frac{\theta}{2}) = \sin(\frac{\theta'}{2})$, puis que $e^{i\phi} = e^{i\phi'}$. Deux arguments sont égaux modulo 2π , mais comme on a imposé $-\pi < \phi \leq \pi$ et $-\pi < \phi' \leq \pi$, on a $\phi = \phi'$. \square

5. Sphère de Bloch

5.1. Représentation

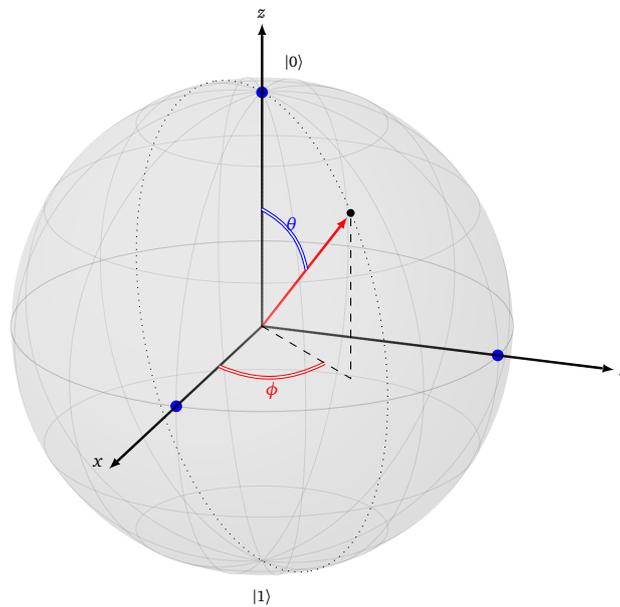
Un qubit $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ est déterminé par ses 2 coefficients complexes α, β , donc par 4 paramètres réels $\text{Re}(\alpha), \text{Im}(\alpha), \text{Re}(\beta), \text{Im}(\beta)$.

Mais un qubit est équivalent à un qubit de la forme :

$$|\psi'\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + \sin\left(\frac{\theta}{2}\right)e^{i\phi}|1\rangle$$

avec seulement deux paramètres réels θ, ϕ qui vérifient $0 \leq \theta \leq \pi$ et $-\pi < \phi \leq \pi$.

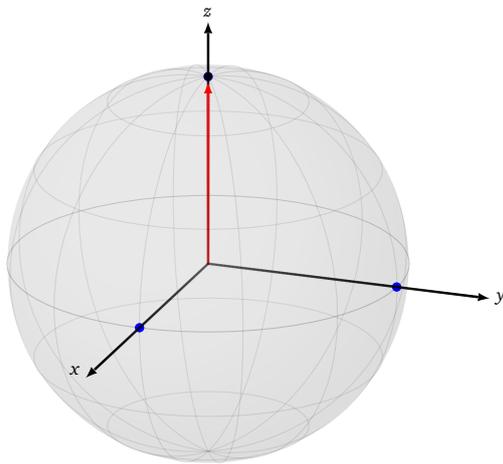
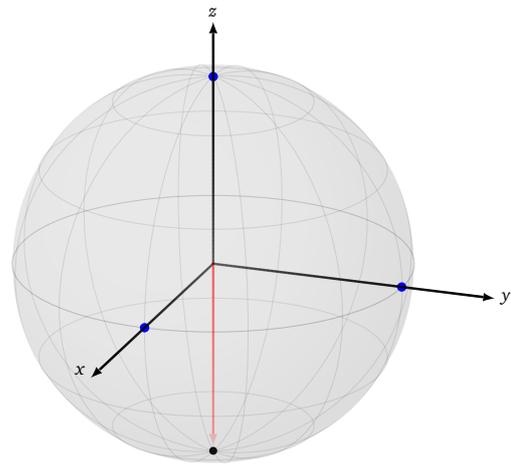
Cela permet de représenter un qubit sur la **sphère de Bloch**, par un point (ou un vecteur) de colatitude θ et longitude ϕ , et un rayon 1.



Les formules pour obtenir les coordonnées $(x, y, z) \in \mathbb{R}^3$ de ce point sont :

$$\begin{cases} x = \sin \theta \cdot \cos \phi \\ y = \sin \theta \cdot \sin \phi \\ z = \cos \theta \end{cases}$$

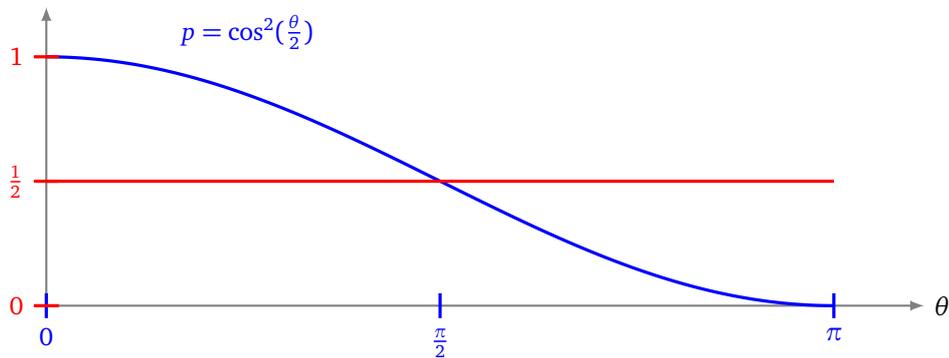
États quantiques de base. L'état de base $|0\rangle$ correspond au pôle Nord de coordonnées $(x, y, z) = (0, 0, 1)$ avec pour colatitude $\theta = 0$ (et n'importe quelle valeur comme longitude ϕ).

 $|0\rangle$  $|1\rangle$

L'état de base $|1\rangle$ correspond au pôle Sud de coordonnées $(x, y, z) = (0, 0, -1)$ avec pour colatitute $\theta = \pi$ (ou 180°) (et n'importe quelle valeur comme longitude ϕ).

Mesure. Pour un qubit $|\psi\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + \sin\left(\frac{\theta}{2}\right)e^{i\phi}|1\rangle$, la probabilité que sa mesure donne 0 est $\cos^2\left(\frac{\theta}{2}\right)$. Ainsi, si le qubit est plus proche du pôle Nord, c'est-à-dire $0 \leq \theta < \frac{\pi}{2}$, alors la probabilité de mesurer 0 est plus forte. Par contre, si le qubit est plus proche du pôle Sud, c'est-à-dire $\frac{\pi}{2} < \theta \leq \pi$, alors la probabilité de mesurer 1 est plus forte. Un qubit sur l'équateur se mesure en 0 ou 1 avec la même probabilité.

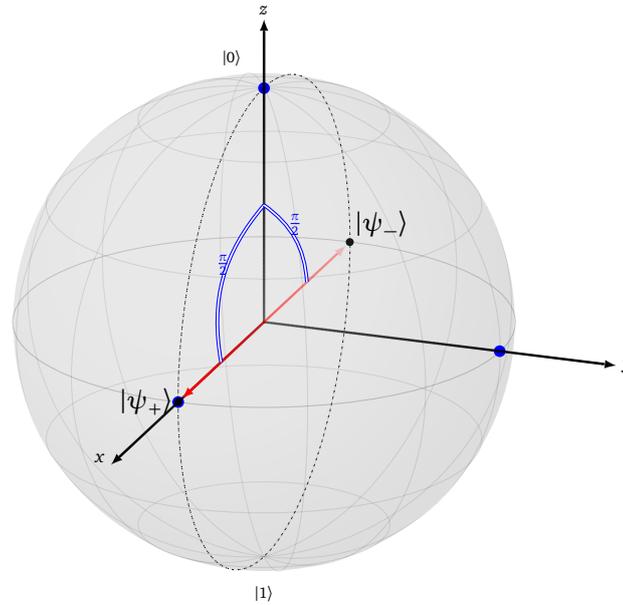
probabilité de mesure 0



Exemple.

L'état $|\psi_+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ a pour paramètres $\theta = \frac{\pi}{2}$ (ou 90°) et $\phi = 0$.

L'état $|\psi_-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$ a pour paramètres $\theta = \frac{\pi}{2}$ et $\phi = \pi$.

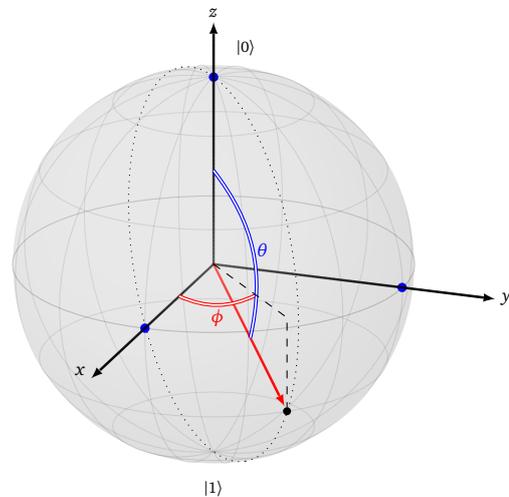
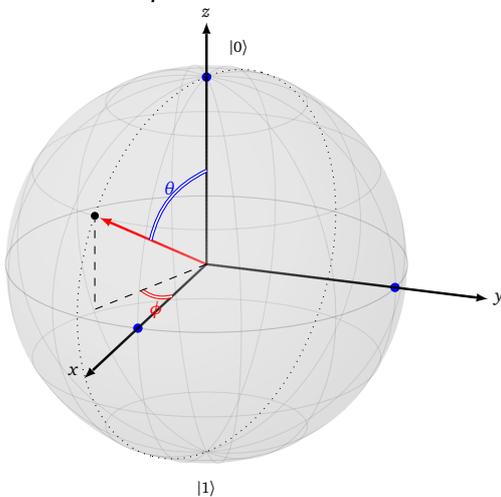


Ils sont tous les deux situés sur l'équateur, donc se mesurent en 0 ou 1 avec une probabilité $\frac{1}{2}$.

Remarque. On rappelle que l'écriture d'un qubit sous la forme $\cos\left(\frac{\theta}{2}\right)|0\rangle + \sin\left(\frac{\theta}{2}\right)e^{i\phi}|1\rangle$ ne s'obtient qu'à équivalence près. Aussi la représentation sur la sphère de Bloch n'est que partielle et ne permet pas une représentation complète d'un qubit.

Exercice.

1. Tracer les points suivants sur la sphère de Bloch :
 - (a) Les points de coordonnées sphériques $(\theta, \phi) = \left(\frac{\pi}{3}, -\frac{\pi}{4}\right)$ et $(\theta, \phi) = \left(\frac{\pi}{2}, \frac{2\pi}{3}\right)$. Exprimer les qubits correspondants.
 - (b) Les points de coordonnées cartésiennes $(x, y, z) = (0, -1, 0)$ et $\left(\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, 0\right), \left(\frac{1}{\sqrt{2}}, 0, -\frac{1}{\sqrt{2}}\right)$. Exprimer les qubits correspondants.
 - (c) Les points des états $|\psi_1\rangle = -\frac{1}{2}|0\rangle - \frac{\sqrt{3}}{2}e^{i\pi}|1\rangle$ (attention au facteur 2 dans la formule $\frac{\theta}{2}$!) et $|\psi_2\rangle = |0\rangle + i|1\rangle$ (penser à normaliser).
2. Trouver les valeurs des qubits suivants placés sur la sphère de Bloch. Une lecture graphique approximative des valeurs θ et ϕ suffit.



3. Trouver l'expression des états quantiques situés sur l'équateur.
4. À quelle transformation géométrique correspond la transformation d'un qubit $|\psi\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + \sin\left(\frac{\theta}{2}\right)e^{i\phi}|1\rangle$ en $|\psi'\rangle = \cos\left(\frac{\pi}{2} - \frac{\theta}{2}\right)|0\rangle + \sin\left(\frac{\pi}{2} - \frac{\theta}{2}\right)e^{i\phi}|1\rangle$?

5. Trouver l'expression de la symétrie centrale de centre l'origine. Exprimez d'abord la transformation $(\theta, \phi) \mapsto (\theta', \phi')$ puis l'action $|\psi\rangle \mapsto |\psi'\rangle$.

5.2. Portes X, Y et Z de Pauli

Les portes de Pauli X, Y et Z transforment un qubit en un autre qubit. Elles ont chacune une interprétation géométrique simple lorsque l'on regarde leur action sur la sphère de Bloch.

Porte X. La porte X échange les coefficients d'un qubit :

$$\text{porte X} : \begin{cases} |0\rangle \mapsto |1\rangle \\ |1\rangle \mapsto |0\rangle \end{cases}$$

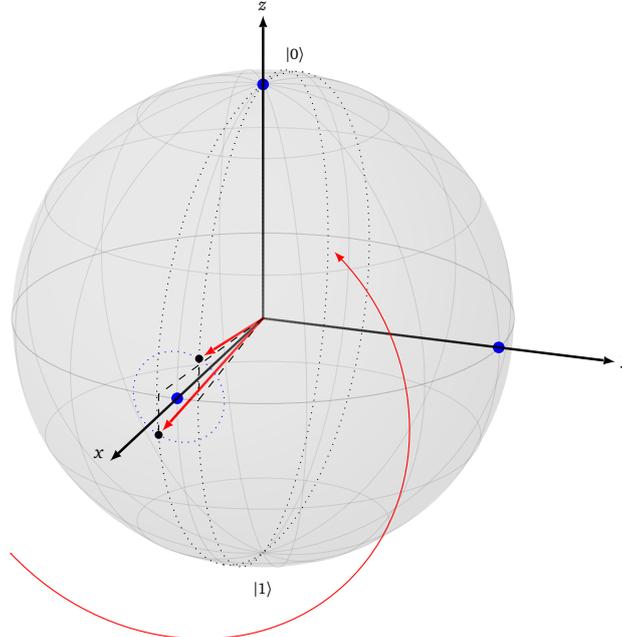
Autrement dit :

$$X(\alpha |0\rangle + \beta |1\rangle) = \beta |0\rangle + \alpha |1\rangle .$$

Regardons ce que cela donne sur la sphère de Bloch. Soit $|\psi\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + \sin\left(\frac{\theta}{2}\right)e^{i\phi}|1\rangle$, alors

$$\begin{aligned} X(|\psi\rangle) &= \sin\frac{\theta}{2}e^{i\phi}|0\rangle + \cos\frac{\theta}{2}|1\rangle \\ &\equiv \sin\frac{\theta}{2}|0\rangle + \cos\frac{\theta}{2}e^{-i\phi}|1\rangle \\ &= \cos\left(\frac{\pi}{2} - \frac{\theta}{2}\right)|0\rangle + \sin\left(\frac{\pi}{2} - \frac{\theta}{2}\right)e^{-i\phi}|1\rangle \\ &= \cos\frac{\theta'}{2}|0\rangle + \sin\frac{\theta'}{2}e^{i\phi'}|1\rangle \quad \text{avec } \theta' = \pi - \theta \text{ et } \phi' \equiv -\phi \pmod{2\pi} \end{aligned}$$

Ainsi les coordonnées sphériques (θ, ϕ) sont transformées par X en $(\pi - \theta, -\phi)$. Géométriquement $X(|\psi\rangle)$ est obtenu sur la sphère de Bloch par la rotation (de l'espace) d'axe (Ox) et d'angle π (un demi-tour donc).

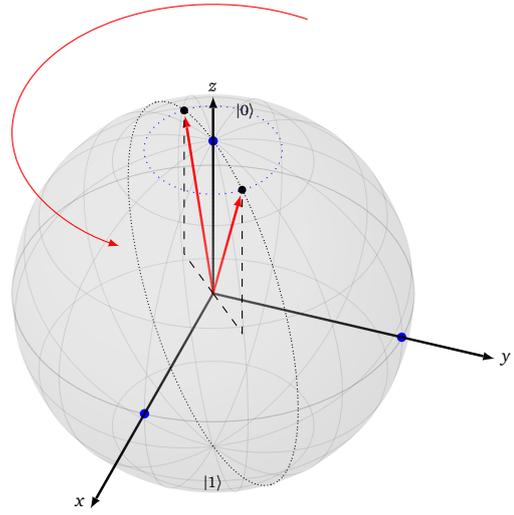
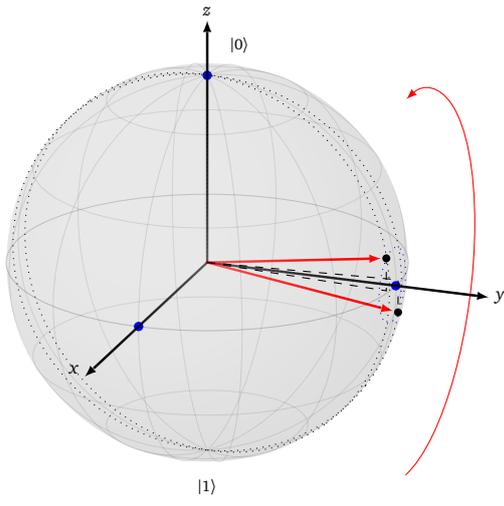


Portes Y et Z. Rappelons l'action des portes de Pauli Y et Z sur les états de base :

$$\text{porte Y} : \begin{cases} |0\rangle \mapsto i|1\rangle \\ |1\rangle \mapsto -i|0\rangle \end{cases} \quad \text{porte Z} : \begin{cases} |0\rangle \mapsto |0\rangle \\ |1\rangle \mapsto -|1\rangle \end{cases}$$

Géométriquement la porte Y correspond à la rotation d'axe (Oy) et d'angle π . Les coordonnées (θ, ϕ) sont transformées en $(\pi - \theta, \pi - \phi)$.

La porte Z correspond à la rotation d'axe (Oz) et d'angle π . Les coordonnées (θ, ϕ) sont transformées en $(\theta, \phi + \pi)$.



Vecteurs et matrices

Vidéo ■ partie 4.1. Vecteurs

Vidéo ■ partie 4.2. Produit scalaire hermitien

Vidéo ■ partie 4.3. Produit tensoriel de vecteurs

Vidéo ■ partie 4.4. Matrices

Vidéo ■ partie 4.5. Matrice adjointe

Vidéo ■ partie 4.6. Matrice unitaire

Un qubit est un vecteur et les opérations sur les qubits sont codées par des matrices. Nous étudions ici le calcul sur les vecteurs, les matrices et leur lien avec les qubits.

1. Vecteurs

1.1. Vecteurs du plan

On commence par la notion de vecteur du plan. Un **vecteur du plan** est la donnée de deux nombres réels, noté :

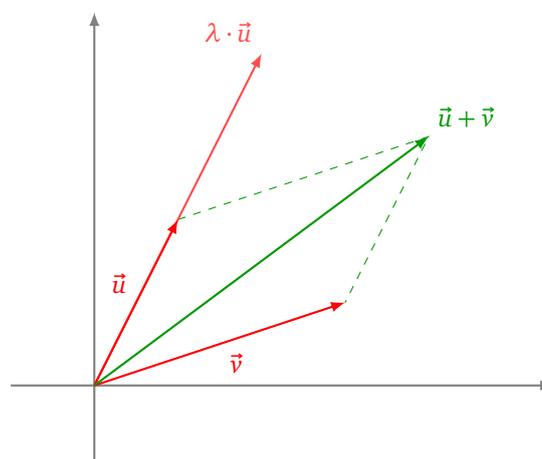
$$\vec{u} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \quad \text{avec } x_1, x_2 \in \mathbb{R}.$$

On peut additionner deux vecteurs :

$$\vec{u} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \quad \vec{v} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \quad \text{alors} \quad \vec{u} + \vec{v} = \begin{pmatrix} x_1 + y_1 \\ x_2 + y_2 \end{pmatrix}.$$

On peut multiplier un vecteur par un coefficient réel λ :

$$\vec{u} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \quad \lambda \cdot \vec{v} = \begin{pmatrix} \lambda x_1 \\ \lambda x_2 \end{pmatrix}.$$



Le **vecteur nul** a toutes ses coordonnées nulles :

$$\vec{0} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

La **norme** (ou **longueur**) d'un vecteur est :

$$\|\vec{u}\| = \sqrt{x_1^2 + x_2^2}$$

1.2. Vecteurs à coefficients complexes

Nous généralisons la notion précédente : le nombre n de coefficients n'est pas limité à 2 et ceux-ci sont maintenant des nombres complexes (et non plus des nombres réels).

Notons \mathbb{K} un corps, qui pour nous sera $\mathbb{K} = \mathbb{R}$ ou $\mathbb{K} = \mathbb{C}$. Fixons $n \geq 1$ un entier. Un **vecteur** de taille n à coefficients dans \mathbb{K} s'écrit :

$$u = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \quad \text{avec } x_1, x_2, \dots, x_n \in \mathbb{K}.$$

Noter qu'à partir de maintenant on n'utilise plus la notation avec une flèche au-dessus du nom du vecteur. L'addition de deux vecteurs :

$$u = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \quad v = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} \quad \text{alors } u + v = \begin{pmatrix} x_1 + y_1 \\ x_2 + y_2 \\ \vdots \\ x_n + y_n \end{pmatrix}.$$

Le **vecteur nul** est :

$$\begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

Nous allons voir plusieurs multiplications associées à des vecteurs. Pour l'instant on définit seulement la multiplication par un scalaire $\lambda \in \mathbb{K}$:

$$u = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \quad \lambda \cdot u = \begin{pmatrix} \lambda x_1 \\ \lambda x_2 \\ \vdots \\ \lambda x_n \end{pmatrix}.$$

Rappels. Pour un nombre complexe x , on note x^* son conjugué. Si x est un nombre réel alors $x^* = x$.

Le **vecteur dual** d'un vecteur u est un vecteur de même taille, dont les coefficients sont les conjugués de ceux de u , et qui est écrit sous la forme d'un vecteur ligne :

$$u = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \quad u^* = (x_1^* \quad x_2^* \quad \dots \quad x_n^*).$$

Exemple.

$$u = \begin{pmatrix} 1+i \\ i \\ 2 \\ 3-4i \end{pmatrix} \quad u^* = (1-i \quad -i \quad 2 \quad 3+4i).$$

1.3. Qubit sous forme de vecteur

Un qubit est un vecteur, ses coefficients sont des nombres complexes et sa taille est toujours une puissance de 2. Un 1-qubit est un vecteur de taille 2 :

$$|\psi\rangle = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \quad \text{avec } x_1, x_2 \in \mathbb{C}.$$

On note $|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ (qui n'est pas le vecteur nul !) et $|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ les deux 1-qubits de base. Plus généralement un n -qubit est un vecteur de taille 2^n :

$$|\psi\rangle = \begin{pmatrix} x_1 \\ \vdots \\ x_{2^n} \end{pmatrix} \quad \text{avec } x_1, \dots, x_{2^n} \in \mathbb{C}.$$

Le dual du vecteur $|\psi\rangle$ sera noté $\langle\psi|$:

$$\langle\psi| = |\psi\rangle^* = (x_1^* \quad x_2^* \quad \cdots \quad x_{2^n}^*)$$

On rappelle que la notation $|\psi\rangle$ se lit « ket psi ». La notation $\langle\psi|$ se lit « bra psi ».

2. Produit scalaire

2.1. Produit scalaire hermitien

Nous allons définir une opération qui, à partir de deux vecteurs, donne un scalaire (c'est-à-dire un nombre complexe si $\mathbb{K} = \mathbb{C}$ ou un nombre réel si $\mathbb{K} = \mathbb{R}$).

Soient

$$u = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \quad v = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}.$$

Le **produit scalaire hermitien** des deux vecteurs u et v est défini par :

$$\langle u|v\rangle = \sum_{i=1}^n x_i^* \cdot y_i$$

Autrement dit :

$$\langle u|v\rangle = x_1^* \cdot y_1 + x_2^* \cdot y_2 + \cdots + x_n^* \cdot y_n.$$

Proposition 1.

Le produit scalaire hermitien est linéaire par rapport au terme de droite et anti-linéaire par rapport au terme de gauche :

$$\langle u|v_1 + v_2\rangle = \langle u|v_1\rangle + \langle u|v_2\rangle \quad \langle u_1 + u_2|v\rangle = \langle u_1|v\rangle + \langle u_2|v\rangle$$

et pour $\lambda \in \mathbb{C}$:

$$\boxed{\langle u|\lambda v\rangle = \lambda \langle u|v\rangle \quad \text{et} \quad \langle \lambda u|v\rangle = \lambda^* \langle u|v\rangle}$$

Enfin :

$$\boxed{\langle v|u\rangle = \langle u|v\rangle^*}$$

Notez bien le coefficient λ^* obtenu par anti-linéarité par rapport au terme de gauche.

Exemple.

$$\begin{aligned} & \langle (1+i)u_1 + (4+2i)u_2 | i v_1 + (1-2i)v_2 \rangle \\ &= i \langle (1+i)u_1 + (4+2i)u_2 | v_1 \rangle + (1-2i) \langle (1+i)u_1 + (4+2i)u_2 | v_2 \rangle \quad \text{linéarité à droite} \\ &= i(1-i) \langle u_1 | v_1 \rangle + i(4-2i) \langle u_2 | v_1 \rangle + (1-2i)(1-i) \langle u_1 | v_2 \rangle + (1-2i)(4-2i) \langle u_2 | v_2 \rangle \quad \text{anti-linéarité à gauche} \end{aligned}$$

2.2. Norme

La *norme* du vecteur $u = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$, notée $\|u\|$, est définie par :

$$\boxed{\|u\| = \sqrt{\sum_{i=1}^n |x_i|^2}}$$

Autrement dit :

$$\|u\|^2 = |x_1|^2 + |x_2|^2 + \dots + |x_n|^2.$$

C'est un nombre réel positif.

On rappelle que $|z|$, le module du nombre complexe $z = a + ib$, est un nombre réel positif, et que :

$$|z|^2 = a^2 + b^2 = z^* \cdot z.$$

On peut donc récrire la norme à l'aide du produit scalaire hermitien :

$$\|u\| = \sqrt{\langle u|u\rangle}.$$

On retient aussi :

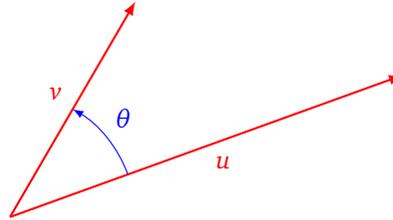
$$\boxed{\|u\|^2 = \langle u|u\rangle}$$

2.3. Vecteurs orthogonaux

Rappel sur le produit scalaire réel. Pour deux vecteurs du plan, le produit scalaire correspond à une mesure de l'angle entre les deux vecteurs. En effet, on a la formule :

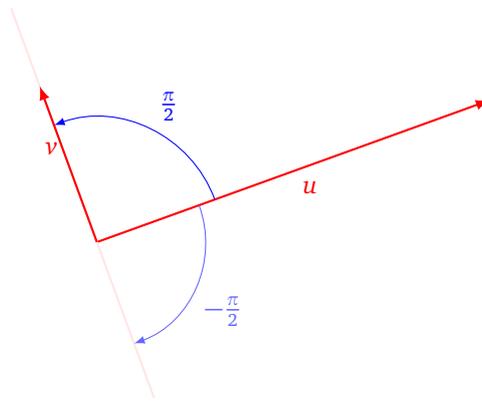
$$\langle u|v \rangle = \|u\| \cdot \|v\| \cdot \cos(\theta)$$

où θ est l'angle entre les vecteurs u et v .



On dit que deux vecteurs du plan sont **orthogonaux** si $\theta = \pm \frac{\pi}{2} \pmod{2\pi}$. Ainsi deux vecteurs du plan sont orthogonaux si et seulement si leur produit scalaire est nul :

$$\langle u|v \rangle = 0.$$



Cas général. On utilise le produit scalaire hermitien pour définir la notion d'orthogonalité pour des vecteurs quelconques. Deux vecteurs u et v de \mathbb{K}^n sont **orthogonaux** si leur produit scalaire hermitien est nul :

$$\langle u|v \rangle = 0.$$

Exemple : le qubit $|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ et le qubit $|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ sont des qubits orthogonaux.

Qubits orthogonaux et sphère de Bloch. Considérons deux qubits $|\psi\rangle$ et $|\psi'\rangle$ écrits sous forme normalisée

$$|\psi\rangle \equiv \cos\left(\frac{\theta}{2}\right)|0\rangle + \sin\left(\frac{\theta}{2}\right)e^{i\phi}|1\rangle = \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) \\ \sin\left(\frac{\theta}{2}\right)e^{i\phi} \end{pmatrix}$$

$$|\psi'\rangle \equiv \cos\left(\frac{\theta'}{2}\right)|0\rangle + \sin\left(\frac{\theta'}{2}\right)e^{i\phi'}|1\rangle = \begin{pmatrix} \cos\left(\frac{\theta'}{2}\right) \\ \sin\left(\frac{\theta'}{2}\right)e^{i\phi'} \end{pmatrix}$$

Sous quelles conditions ces qubits sont-ils orthogonaux? On exclut le cas $\theta = 0$, qui correspond au qubit $|0\rangle$, car le seul qubit orthogonal à $|0\rangle$ est $|1\rangle$ (à équivalence près). Pour la même raison on exclut le cas $\theta = \pi$, qui correspond au qubit $|1\rangle$. Ainsi on a

$$0 < \theta, \theta' < \pi \quad \text{et} \quad -\pi < \phi, \phi' \leq \pi.$$

On calcule leur produit scalaire hermitien :

$$\begin{aligned} \langle \psi|\psi' \rangle &= \cos\left(\frac{\theta}{2}\right) \cdot \cos\left(\frac{\theta'}{2}\right) + \sin\left(\frac{\theta}{2}\right)e^{-i\phi} \cdot \sin\left(\frac{\theta'}{2}\right)e^{i\phi'} \\ &= \cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta'}{2}\right) + \sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\theta'}{2}\right)e^{i(\phi'-\phi)} \end{aligned}$$

Avant d'être nul, ce produit scalaire doit être un nombre réel. Bien sûr, les sinus et les cosinus sont des nombres réels, mais il faut aussi que $e^{i(\phi' - \phi)}$ soit un nombre réel. Or

$$e^{i(\phi' - \phi)} \in \mathbb{R} \iff \phi' - \phi \equiv 0 \pmod{2\pi} \text{ ou } \phi' - \phi \equiv \pi \pmod{2\pi}.$$

En effet, on a $e^{i\alpha} = 1$ si et seulement si $\alpha \equiv 0 \pmod{2\pi}$, et $e^{i\alpha} = -1$ si et seulement si $\alpha \equiv \pi \pmod{2\pi}$.

Premier cas : $\phi' - \phi \equiv 0 \pmod{2\pi}$. Alors $\phi = \phi'$, et

$$\begin{aligned} \langle \psi | \psi' \rangle = 0 &\iff \cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta'}{2}\right) + \sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\theta'}{2}\right) = 0 \\ &\iff \cos\left(\frac{\theta}{2} - \frac{\theta'}{2}\right) = 0 \\ &\iff \frac{\theta}{2} - \frac{\theta'}{2} = \frac{\pi}{2} \pmod{\pi} \\ &\iff \theta - \theta' = \pi \pmod{2\pi} \end{aligned}$$

Mais cette dernière égalité est impossible car $0 < \theta < \pi$ et $0 < \theta' < \pi$, donc $-\pi < \theta - \theta' < \pi$. Le premier cas ne conduit donc à aucune solution.

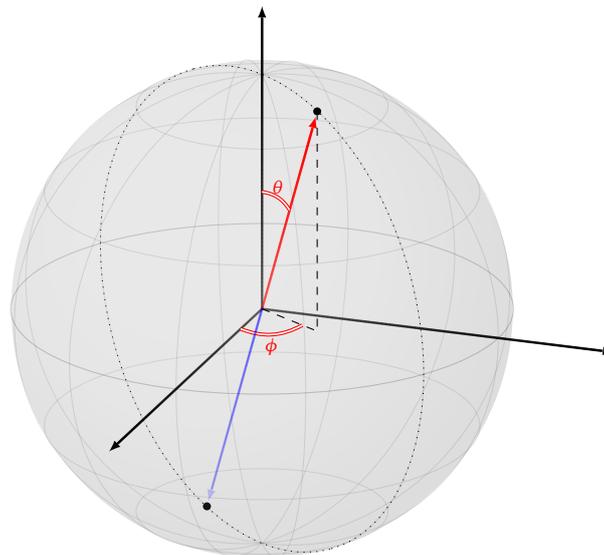
Second cas : $\phi' - \phi \equiv \pi \pmod{2\pi}$. Alors $\phi' = \phi + \pi \pmod{2\pi}$, et alors

$$\begin{aligned} \langle \psi | \psi' \rangle = 0 &\iff \cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta'}{2}\right) - \sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\theta'}{2}\right) = 0 \\ &\iff \cos\left(\frac{\theta}{2} + \frac{\theta'}{2}\right) = 0 \\ &\iff \frac{\theta}{2} + \frac{\theta'}{2} = \frac{\pi}{2} \pmod{\pi} \\ &\iff \theta' = \pi - \theta \pmod{2\pi} \end{aligned}$$

Nous avons obtenu une solution : le qubit de représentation $(\theta', \phi') = (\pi - \theta, \phi + \pi)$ est orthogonal au qubit de représentation (θ, ϕ) . Géométriquement le qubit $|\psi'\rangle$ est antipodal au qubit $|\psi\rangle$ sur la sphère de Bloch. Autrement dit, l'un s'obtient de l'autre par la symétrie centrale centrée à l'origine. Noter que c'est aussi valide pour $|0\rangle$ et $|1\rangle$.

On retient :

Deux 1-qubits sont orthogonaux si, et seulement si, ils sont antipodaux sur la sphère de Bloch.



2.4. Inégalité de Cauchy-Schwarz

Terminons par l'énoncé d'une inégalité importante.

Théorème 1 (Inégalité de Cauchy-Schwarz).

$$|\langle u|v \rangle| \leq \|u\| \cdot \|v\|$$

3. Produit tensoriel de vecteurs

3.1. Définition

Soient

$$u = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \in \mathbb{K}^n \quad \text{et} \quad v = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{pmatrix} \in \mathbb{K}^m.$$

Le **produit tensoriel** de u par v , noté $u \otimes v$, est le vecteur de \mathbb{K}^{nm} défini par :

$$u \otimes v = \begin{pmatrix} x_1 \begin{pmatrix} y_1 \\ \vdots \\ y_m \end{pmatrix} \\ x_2 \begin{pmatrix} y_1 \\ \vdots \\ y_m \end{pmatrix} \\ \vdots \\ \vdots \\ x_n \begin{pmatrix} y_1 \\ \vdots \\ y_m \end{pmatrix} \end{pmatrix} = \begin{pmatrix} x_1 y_1 \\ \vdots \\ x_1 y_m \\ x_2 y_1 \\ \vdots \\ x_2 y_m \\ \vdots \\ \vdots \\ x_n y_1 \\ \vdots \\ x_n y_m \end{pmatrix}$$

Autrement dit, on prend des copies du vecteur v , et chaque copie est multipliée par une coordonnée du vecteur u .

Par exemple :

$$\begin{pmatrix} a \\ b \end{pmatrix} \otimes \begin{pmatrix} c \\ d \end{pmatrix} = \begin{pmatrix} a \begin{pmatrix} c \\ d \end{pmatrix} \\ b \begin{pmatrix} c \\ d \end{pmatrix} \end{pmatrix} = \begin{pmatrix} ac \\ ad \\ bc \\ bd \end{pmatrix}$$

En général $u \otimes v \neq v \otimes u$:

$$\begin{pmatrix} 1 \\ 2 \end{pmatrix} \otimes \begin{pmatrix} 3 \\ 4 \\ 5 \end{pmatrix} = \begin{pmatrix} 3 \\ 4 \\ 5 \\ 6 \\ 8 \\ 10 \end{pmatrix} \quad \begin{pmatrix} 3 \\ 4 \\ 5 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 2 \end{pmatrix} = \begin{pmatrix} 3 \\ 6 \\ 4 \\ 8 \\ 5 \\ 10 \end{pmatrix}$$

3.2. Propriétés

Proposition 2.

Le produit tensoriel est linéaire à gauche et à droite :

$$(\lambda u) \otimes v = \lambda(u \otimes v) = u \otimes (\lambda v) \quad \lambda \in \mathbb{C}$$

$$(u_1 + u_2) \otimes v = u_1 \otimes v + u_2 \otimes v$$

$$u \otimes (v_1 + v_2) = u \otimes v_1 + u \otimes v_2$$

Exemple.

On développe l'expression $(u_1 + u_2) \otimes (v_1 + v_2)$ en deux temps :

$$\begin{aligned} (u_1 + u_2) \otimes (v_1 + v_2) &= u_1 \otimes (v_1 + v_2) + u_2 \otimes (v_1 + v_2) && \text{linéarité à gauche} \\ &= u_1 \otimes v_1 + u_1 \otimes v_2 + u_2 \otimes v_1 + u_2 \otimes v_2 && \text{linéarité à droite} \end{aligned}$$

3.3. Produit de qubits

Si $|\phi\rangle$ est un n -qubit et $|\psi\rangle$ est un m -qubit, alors le **produit** de $|\phi\rangle \cdot |\psi\rangle$ est défini par le produit tensoriel :

$$|\phi\rangle \cdot |\psi\rangle = |\phi\rangle \otimes |\psi\rangle$$

Le produit $|\phi\rangle \cdot |\psi\rangle$ est un $(n + m)$ -qubit (un vecteur de taille $2^n \cdot 2^m = 2^{n+m}$).

On rappelle que

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Ainsi :

$$\begin{aligned} |0.0\rangle = |0\rangle \cdot |0\rangle &= \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} & |0.1\rangle = |0\rangle \cdot |1\rangle &= \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \\ |1.0\rangle = |1\rangle \cdot |0\rangle &= \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} & |1.1\rangle = |1\rangle \cdot |1\rangle &= \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \end{aligned}$$

Exemple.

Voyons comment calculer le produit $|\phi\rangle \cdot |\psi\rangle$ dans le cas où :

$$|\phi\rangle = (1 + 2i)|0\rangle + i|1\rangle \quad |\psi\rangle = 2|0\rangle + (3 - 4i)|1\rangle$$

1. *Calcul tensoriel.* Nous revenons à la définition vectorielle des qubits.

$$\begin{aligned}
 |\phi\rangle \otimes |\psi\rangle &= \left((1+2i) \begin{pmatrix} 1 \\ 0 \end{pmatrix} + i \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right) \otimes \left(2 \begin{pmatrix} 1 \\ 0 \end{pmatrix} + (3-4i) \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right) \\
 &= (1+2i) \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes 2 \begin{pmatrix} 1 \\ 0 \end{pmatrix} + (1+2i) \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes (3-4i) \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\
 &\quad + i \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes 2 \begin{pmatrix} 1 \\ 0 \end{pmatrix} + i \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes (3-4i) \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\
 &= 2(1+2i) |0.0\rangle + (1+2i)(3-4i) |0.1\rangle + 2i |1.0\rangle + (4+3i) |1.1\rangle
 \end{aligned}$$

2. *Calcul formel.* C'est la technique vue dans le chapitre « Découverte de l'informatique quantique ». C'est en fait le même calcul que précédemment, mais sans revenir aux vecteurs :

$$\begin{aligned}
 |\phi\rangle \otimes |\psi\rangle &= ((1+2i)|0\rangle + i|1\rangle) \otimes (2|0\rangle + (3-4i)|1\rangle) \\
 &= (1+2i)|0\rangle \otimes 2|0\rangle + (1+2i)|0\rangle \otimes (3-4i)|1\rangle + i|1\rangle \otimes 2|0\rangle + i|1\rangle \otimes (3-4i)|1\rangle \\
 &= 2(1+2i) |0.0\rangle + (1+2i)(3-4i) |0.1\rangle + 2i |1.0\rangle + (4+3i) |1.1\rangle
 \end{aligned}$$

On note l'intérêt de la notation $|\cdot\rangle$ qui permet d'écrire les calculs de façon condensée, mais il faut bien comprendre que la justification théorique qui nous permet cette écriture est le calcul tensoriel sur les vecteurs.

3.4. Intrication quantique

Définition.

- Un 2-qubit $|\phi\rangle$ est **non intriqué** s'il existe deux 1-qubits $|\psi_1\rangle$ et $|\psi_2\rangle$ tels que :

$$|\phi\rangle = |\psi_1\rangle \cdot |\psi_2\rangle.$$

- S'il n'existe aucun $|\psi_1\rangle$ et $|\psi_2\rangle$ tels que $|\phi\rangle = |\psi_1\rangle \cdot |\psi_2\rangle$, alors le qubit $|\phi\rangle$ est dit **intriqué**.

Exemple.

Le 2-qubit $|\phi\rangle$ suivant n'est pas intriqué :

$$|\phi\rangle = |0.0\rangle - |0.1\rangle + |1.0\rangle - |1.1\rangle.$$

En effet si on pose :

$$|\psi_1\rangle = |0\rangle + |1\rangle \quad |\psi_2\rangle = |0\rangle - |1\rangle$$

alors

$$|\psi_1\rangle \cdot |\psi_2\rangle = (|0\rangle + |1\rangle) \cdot (|0\rangle - |1\rangle) = |0.0\rangle - |0.1\rangle + |1.0\rangle - |1.1\rangle = |\phi\rangle.$$

Exemple.

L'état de Bell $|\Phi^+\rangle$ est intriqué :

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}} |0.0\rangle + \frac{1}{\sqrt{2}} |1.1\rangle$$

Preuve. Supposons par l'absurde que $|\Phi^+\rangle$ ne soit pas intriqué, alors il existerait $|\psi_1\rangle = \alpha|0\rangle + \beta|1\rangle$ et $|\psi_2\rangle = \alpha'|0\rangle + \beta'|1\rangle$ tels que $|\Phi^+\rangle = |\psi_1\rangle \cdot |\psi_2\rangle$, où $\alpha, \beta, \alpha', \beta'$ sont des nombres complexes.

D'une part, on aurait :

$$|\Phi^+\rangle = |\psi_1\rangle \cdot |\psi_2\rangle = \alpha\alpha' |0.0\rangle + \alpha\beta' |0.1\rangle + \beta\alpha' |1.0\rangle + \beta\beta' |1.1\rangle.$$

Mais d'autre part $|\Phi^+\rangle = \frac{1}{\sqrt{2}}|0.0\rangle + \frac{1}{\sqrt{2}}|1.1\rangle$. Par identification des coefficients on obtient :

$$\begin{cases} \alpha\alpha' = \frac{1}{\sqrt{2}} \\ \beta\beta' = \frac{1}{\sqrt{2}} \end{cases} \quad \text{et} \quad \begin{cases} \alpha\beta' = 0 \\ \beta\alpha' = 0 \end{cases} .$$

Les équations de gauche impliquent que α, β, α' et β' sont tous non nuls, ce qui contredit les équations de droite. Ainsi notre hypothèse de départ est nécessairement fautive, ce qui implique qu'il ne peut exister de tels $|\psi_1\rangle$ et $|\psi_2\rangle$, c'est-à-dire que le qubit $|\Phi^+\rangle$ est intriqué.

4. Matrices

Nous allons voir les notions de base concernant les matrices. Nous nous concentrons en particulier sur les matrices de taille 2×2 .

4.1. Définition

Une **matrice** est un tableau de nombres représenté de la manière suivante :

$$A = \begin{pmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,j} & \dots & a_{1,p} \\ a_{2,1} & a_{2,2} & \dots & a_{2,j} & \dots & a_{2,p} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ a_{i,1} & a_{i,2} & \dots & a_{i,j} & \dots & a_{i,p} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ a_{n,1} & a_{n,2} & \dots & a_{n,j} & \dots & a_{n,p} \end{pmatrix} \quad \text{ou} \quad A = (a_{i,j})_{\substack{1 \leq i \leq n \\ 1 \leq j \leq p}} \quad \text{ou} \quad (a_{i,j}).$$

Les $a_{i,j}$ seront pour nous des nombres réels ou des nombres complexes. On note $M_{n,p}(\mathbb{K})$ les matrices de taille $n \times p$ à coefficients dans \mathbb{K} .

Par exemple :

$$A = \begin{pmatrix} 1+i & -2i & 5 \\ i & 0 & 1+7i \end{pmatrix} \in M_{2,3}(\mathbb{C}),$$

A est une matrice 2×3 à coefficients complexes.

Si $n = p$ (même nombre de lignes que de colonnes), la matrice est dite **matrice carrée**. On note $M_n(\mathbb{K})$ au lieu de $M_{n,n}(\mathbb{K})$. Dans ce cas les éléments $a_{1,1}, a_{2,2}, \dots, a_{n,n}$ forment la **diagonale principale** de la matrice :

$$\begin{pmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1} & a_{n,2} & \dots & a_{n,n} \end{pmatrix} .$$

On retrouve des cas particuliers déjà rencontrés.

- Une matrice qui n'a qu'une seule ligne ($n = 1$) est appelée **matrice ligne** ou **vecteur ligne**. On la note

$$A = (a_{1,1} \quad a_{1,2} \quad \dots \quad a_{1,p}).$$

- De même, une matrice qui n'a qu'une seule colonne ($p = 1$) est appelée **matrice colonne** ou **vecteur colonne**. On la note

$$A = \begin{pmatrix} a_{1,1} \\ a_{2,1} \\ \vdots \\ a_{n,1} \end{pmatrix} .$$

Définition (Somme de deux matrices).

Soient A et B deux matrices ayant la même taille $n \times p$. Leur **somme** $C = A + B$ est la matrice de taille $n \times p$ définie par

$$c_{ij} = a_{ij} + b_{ij}$$

pour $1 \leq i \leq n$ et $1 \leq j \leq p$.

En d'autres termes, on somme coefficients à coefficients.

Remarque : on note indifféremment a_{ij} ou $a_{i,j}$ pour les coefficients de la matrice A .

Si $A = \begin{pmatrix} 3+i & -2 \\ 1 & 7i \end{pmatrix}$ et $B = \begin{pmatrix} -2 & 5+2i \\ 3i & -i \end{pmatrix}$ alors $A+B = \begin{pmatrix} 1+i & 3+2i \\ 1+3i & 6i \end{pmatrix}$.

La matrice de taille $n \times p$ dont tous les coefficients sont des zéros est appelée la **matrice nulle** et est notée $0_{n,p}$ ou plus simplement 0 . Dans le calcul matriciel, la matrice nulle joue le rôle du nombre 0 pour les réels, c'est l'élément neutre pour l'addition.

4.2. Produit de matrices

Définition (Produit de deux matrices).

Soient $A = (a_{ij})$ une matrice $n \times p$ et $B = (b_{ij})$ une matrice $p \times q$. Alors le produit $C = AB$ est une matrice $n \times q$ dont les coefficients c_{ij} sont définis par :

$$c_{ij} = \sum_{k=1}^p a_{ik} b_{kj}$$

où $1 \leq i \leq n$ et $1 \leq j \leq q$.

On peut écrire le coefficient général de façon plus développée, à savoir :

$$c_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + \dots + a_{ik}b_{kj} + \dots + a_{ip}b_{pj}.$$

Il est commode de disposer les calculs de la façon suivante :

$$A \rightarrow \begin{pmatrix} \times & \times & \times & \times \end{pmatrix} \begin{pmatrix} \times \\ \times \\ \times \\ \times \\ \vdots \\ - & - & - & c_{ij} \end{pmatrix} \leftarrow AB \quad \begin{pmatrix} \times \\ \times \\ \times \\ \times \\ \vdots \\ - & - & - & c_{ij} \end{pmatrix} \leftarrow B$$

Avec cette disposition, on considère d'abord la ligne de la matrice A située à gauche du coefficient que l'on veut calculer (ligne numéro i représentée par des \times dans A) et aussi la colonne de la matrice B située au-dessus du coefficient que l'on veut calculer (colonne numéro j représentée par des \times dans B). On calcule le produit du premier coefficient de la ligne par le premier coefficient de la colonne ($a_{i1} \times b_{1j}$), que l'on ajoute au produit du deuxième coefficient de la ligne par le deuxième coefficient de la colonne ($a_{i2} \times b_{2j}$), que l'on ajoute au produit du troisième...

4.3. Exemples

Exemple.

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \end{pmatrix} \quad B = \begin{pmatrix} 1 & 2 \\ -1 & 1 \\ 1 & 1 \end{pmatrix}$$

On dispose d'abord le produit correctement (à gauche) : la matrice obtenue sera de taille 2×2 . Puis on calcule chacun des coefficients, en commençant par le premier coefficient $c_{11} = 1 \times 1 + 2 \times (-1) + 3 \times 1 = 2$ (au milieu), puis les autres (à droite).

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \end{pmatrix} \begin{pmatrix} 1 & 2 \\ -1 & 1 \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \end{pmatrix} \begin{pmatrix} 1 & 2 \\ -1 & 1 \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \end{pmatrix} \begin{pmatrix} 1 & 2 \\ -1 & 1 \\ 1 & 1 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \end{pmatrix} \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix} = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \end{pmatrix} \begin{pmatrix} 2 & c_{12} \\ c_{21} & c_{22} \end{pmatrix} = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \end{pmatrix} \begin{pmatrix} 2 & 7 \\ 3 & 11 \end{pmatrix}$$

La matrice carrée suivante s'appelle la **matrice identité** :

$$I_n = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{pmatrix}.$$

Ses éléments diagonaux sont égaux à 1 et tous ses autres éléments sont égaux à 0. Elle se note I_n ou simplement I . Dans le calcul matriciel, la matrice identité joue un rôle analogue à celui du nombre 1 pour les réels. C'est l'élément neutre pour la multiplication. En d'autres termes :

Proposition 3.

Si A est une matrice $n \times p$, alors

$$I_n \cdot A = A \quad \text{et} \quad A \cdot I_p = A.$$

4.4. Matrice inverse, déterminant

Définition (Matrice inverse).

Soit A une matrice carrée de taille $n \times n$. S'il existe une matrice carrée B de taille $n \times n$ telle que

$$AB = I \quad \text{et} \quad BA = I,$$

on dit que A est **inversible**. On appelle B l'**inverse de A** et on la note A^{-1} .

Considérons le cas d'une matrice 2×2 : $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$.

Proposition 4.

Soit $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$. Si $ad - bc \neq 0$, alors A est inversible et

$$A^{-1} = \frac{1}{ad - bc} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$$

Le nombre $ad - bc \in \mathbb{K}$ s'appelle le **déterminant** de la matrice $A \in M_2(\mathbb{K})$.

Plus généralement pour une matrice carrée $A \in M_n(\mathbb{K})$, il existe un scalaire $\det(A) \in \mathbb{K}$, appelé **déterminant** de A tel que :

- si $\det(A) \neq 0$ alors la matrice A est inversible ;
- $\det(AB) = \det(A) \cdot \det(B)$;
- $\det(I) = 1$;
- $\det(A^{-1}) = 1/\det(A)$, si A est inversible.

Nous admettons ces propriétés et nous n'expliquons pas ici comment calculer le déterminant en général.

5. Matrice adjointe

Une matrice adjointe est la version complexe d'une matrice transposée.

5.1. La transposition

On commence par rappeler que la transposition est une opération qui transforme une matrice : les lignes de A deviennent les colonnes de A^T .

Voici une matrice A de taille $n \times p$ et sa **matrice transposée** notée A^T qui est de taille $p \times n$:

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1p} \\ a_{21} & a_{22} & \dots & a_{2p} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \dots & a_{np} \end{pmatrix} \quad A^T = \begin{pmatrix} a_{11} & a_{21} & \dots & a_{n1} \\ a_{12} & a_{22} & \dots & a_{n2} \\ \vdots & \vdots & & \vdots \\ a_{1p} & a_{2p} & \dots & a_{np} \end{pmatrix}.$$

Autrement dit : le coefficient à la place (i, j) de A^T est a_{ji} .

5.2. Matrice adjointe

Nos matrices ont des coefficients complexes, la matrice adjointe s'obtient par transposition et conjugaison complexe. On rappelle que si $a \in \mathbb{C}$, alors a^* est le conjugué.

Définition.

On appelle **matrice adjointe** de A , de taille $n \times p$, la matrice A^* de taille $p \times n$ définie par :

$$A^* = \begin{pmatrix} a_{11}^* & a_{21}^* & \dots & a_{n1}^* \\ a_{12}^* & a_{22}^* & \dots & a_{n2}^* \\ \vdots & \vdots & & \vdots \\ a_{1p}^* & a_{2p}^* & \dots & a_{np}^* \end{pmatrix}.$$

Exemple.

$$A = \begin{pmatrix} 1+i & 2+i \\ 3+i & 4+i \\ 5+i & 6+i \end{pmatrix} \quad A^* = \begin{pmatrix} 1-i & 3-i & 5-i \\ 2-i & 4-i & 6-i \end{pmatrix}$$

Nous avons déjà vu le cas des vecteurs : l'adjoint d'un vecteur colonne est un vecteur ligne, et réciproquement.

$$u = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \quad u^* = (x_1^* \quad \dots \quad x_n^*)$$

$$v = (x_1 \quad \dots \quad x_n) \quad v^* = \begin{pmatrix} x_1^* \\ \vdots \\ x_n^* \end{pmatrix}$$

Proposition 5.

Pour deux matrices A et B de tailles respectives $n \times p$ et $p \times m$:

$$\boxed{(A^*)^* = A \quad (AB)^* = B^*A^*}$$

La relation $(A^*)^* = A$ signifie que l'adjointe de l'adjointe est la matrice elle-même. C'est déjà le cas pour la transposition et aussi la conjugaison complexe.

On va prouver la seconde assertion $(AB)^* = B^*A^*$. Notez bien l'inversion de l'ordre, que l'on rencontre déjà pour les inverses $(AB)^{-1} = B^{-1}A^{-1}$. On rappelle aussi que l'ordre d'un produit de matrices est important, car en général $AB \neq BA$.

Démonstration. On va faire la preuve pour les matrices 2×2 uniquement. Soient :

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \quad B = \begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix}.$$

Alors

$$AB = \begin{pmatrix} a\alpha + b\gamma & a\beta + b\delta \\ c\alpha + d\gamma & c\beta + d\delta \end{pmatrix} \quad (AB)^* = \begin{pmatrix} a^*\alpha^* + b^*\gamma^* & c^*\alpha^* + d^*\gamma^* \\ a^*\beta^* + b^*\delta^* & c^*\beta^* + d^*\delta^* \end{pmatrix}.$$

Et d'autre part

$$A^* = \begin{pmatrix} a^* & c^* \\ b^* & d^* \end{pmatrix} \quad B^* = \begin{pmatrix} \alpha^* & \gamma^* \\ \beta^* & \delta^* \end{pmatrix} \quad B^*A^* = \begin{pmatrix} a^*\alpha^* + b^*\gamma^* & c^*\alpha^* + d^*\gamma^* \\ a^*\beta^* + b^*\delta^* & c^*\beta^* + d^*\delta^* \end{pmatrix}$$

On a bien $(AB)^* = B^*A^*$. □

5.3. Notation bra-ket

On rappelle la notation « ket » $|\psi\rangle$ et la notation « bra » $\langle\phi|$. En posant :

$$|\psi\rangle = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \quad \text{et} \quad \langle\phi| = \begin{pmatrix} x_1 & & \\ & \ddots & \\ & & x_n \end{pmatrix}$$

alors

$$\langle\phi| = |\phi\rangle^* = (x_1^* \quad \cdots \quad x_n^*).$$

Calculons le produit de matrices $\langle\phi| \times |\psi\rangle$:

$$\langle\phi| \times |\psi\rangle = (x_1^* \quad \cdots \quad x_n^*) \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} = x_1^*y_1 + \cdots + x_n^*y_n = \langle\phi|\psi\rangle.$$

C'est le produit d'un vecteur ligne par un vecteur colonne qui donne une matrice de taille 1×1 , qu'on identifie à un nombre complexe.

Ce calcul justifie la notation « bra-ket » : le produit $\langle\phi| \times |\psi\rangle$ correspond au produit scalaire hermitien $\langle\phi|\psi\rangle$. Ainsi la notation « bra-ket » est un jeu de mots associé au « bracket » du produit scalaire hermitien (*bracket* signifie crochet).

5.4. Produit scalaire hermitien

Proposition 6.

$$\boxed{\langle Au|v\rangle = \langle u|A^*v\rangle}$$

Démonstration. Nous faisons la preuve uniquement pour les matrices de taille 2×2 .

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \quad A^* = \begin{pmatrix} a^* & c^* \\ b^* & d^* \end{pmatrix} \quad u = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \quad v = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$$

$$Au = \begin{pmatrix} ax_1 + bx_2 \\ cx_1 + dx_2 \end{pmatrix} \quad \langle Au|v \rangle = (ax_1 + bx_2)^*y_1 + (cx_1 + dx_2)^*y_2$$

$$A^*v = \begin{pmatrix} a^*y_1 + c^*y_2 \\ b^*y_1 + d^*y_2 \end{pmatrix} \quad \langle u|A^*v \rangle = x_1^*(a^*y_1 + c^*y_2) + x_2^*(b^*y_1 + d^*y_2)$$

Ainsi

$$\langle Au|v \rangle = a^*x_1^*y_1 + b^*x_2^*y_1 + c^*x_1^*y_2 + d^*x_2^*y_2 = \langle u|A^*v \rangle.$$

□

6. Matrice unitaire

On travaille souvent avec des qubits de norme 1. Les portes logiques transforment les qubits, mais doivent tout de même transformer un qubit $|\phi\rangle$ de norme 1 en un qubit $|\psi\rangle$ de norme 1.

Lorsque cette transformation est linéaire et s'écrit $A|\phi\rangle = |\psi\rangle$, la matrice A est d'un type particulier : c'est une matrice unitaire. Dans ce chapitre les exemples seront des matrices 2×2 . On retrouvera le cas général dans le chapitre « Portes quantiques ».

6.1. Définition

Définition.

Une matrice $A \in M_n$ est **unitaire** si :

$$A^*A = I$$

On note U_n l'ensemble des matrices unitaires de taille $n \times n$.

Si A est une matrice unitaire alors on a

$$A^{-1} = A^* \quad \text{et} \quad AA^* = I.$$

Exemple.

Les matrices de Pauli sont les matrices unitaires suivantes :

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Vérifier que l'on a bien $A^*A = I$. De plus pour ces exemples on a $A^* = A$.

La propriété fondamentale des matrices unitaires est qu'elles préservent le produit scalaire hermitien.

Proposition 7.

Si A est une matrice unitaire alors

$$\langle Au|Av \rangle = \langle u|v \rangle$$

En termes de vecteurs du plan, cela signifie que l'angle entre deux vecteurs est préservé par l'action d'une matrice unitaire.



Démonstration.

$$\langle Au|Av \rangle = \langle u|A^*Av \rangle = \langle u|v \rangle$$

□

6.2. Matrice unitaire de dimension 2

Soit $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ de taille 2×2 . Notons cette matrice à l'aide de ses vecteurs colonnes :

$$A = (u \ v) \quad \text{avec} \quad u = \begin{pmatrix} a \\ c \end{pmatrix}, \quad v = \begin{pmatrix} b \\ d \end{pmatrix}.$$

Proposition 8.

La matrice A est unitaire si et seulement si les vecteurs (u, v) forment une base orthonormale, c'est-à-dire satisfont les conditions :

$$\|u\| = 1, \quad \|v\| = 1 \quad \text{et} \quad \langle u|v \rangle = 0.$$

Démonstration.

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \quad A^* = \begin{pmatrix} a^* & c^* \\ b^* & d^* \end{pmatrix}$$

$$A^*A = \begin{pmatrix} aa^* + cc^* & ba^* + dc^* \\ ab^* + cd^* & bb^* + dd^* \end{pmatrix}$$

Si A est une matrice unitaire alors

$$A^*A = I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

On identifie les coefficients :

$$\begin{cases} aa^* + cc^* = 1 \\ bb^* + dd^* = 1 \\ a^*b + c^*d = 0 \end{cases} \quad \text{donc} \quad \begin{cases} \|u\|^2 = |a|^2 + |c|^2 = 1 \\ \|v\|^2 = |b|^2 + |d|^2 = 1 \\ \langle u|v \rangle = a^*b + c^*d = 0 \end{cases}$$

On n'utilise pas l'égalité $ab^* + cd^* = 0$ qui est en fait $(a^*b + c^*d)^* = 0$.

Réciproquement, si on a les égalités $\|u\| = 1, \|v\| = 1$ et $\langle u|v \rangle = 0$, alors les coefficients de A^*A sont les coefficients de l'identité. □

Exemple.

La matrice suivante est unitaire :

$$U(\theta, \phi, \lambda) = \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & -\sin\left(\frac{\theta}{2}\right)e^{i\lambda} \\ \sin\left(\frac{\theta}{2}\right)e^{i\phi} & \cos\left(\frac{\theta}{2}\right)e^{i(\phi+\lambda)} \end{pmatrix}.$$

On vérifie que les deux vecteurs verticaux formant cette matrice sont de norme 1 et orthogonaux.

Cette transformation est disponible sous la forme d'une porte quantique.

$$\text{---} \boxed{U_3(\theta, \phi, \lambda)} \text{---}$$

Proposition 9.

L'ensemble des matrices unitaires forme un groupe pour la multiplication. En particulier si $A, B \in U_n$ alors $AB \in U_n$ et $A^{-1} \in U_n$.

Démonstration. Soient $A, B \in U_n$.

$$(AB)^*(AB) = (B^*A^*)(AB) = B^*(A^*A)B = B^*IB = B^*B = I.$$

De même, comme $A^{-1} = A^*$:

$$(A^{-1})^*A^{-1} = (A^*)^*A^* = AA^* = I.$$

□

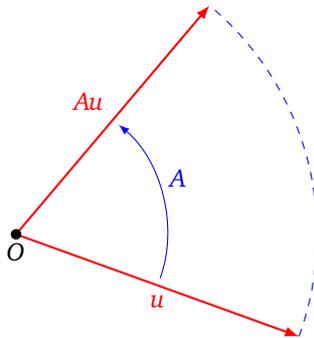
6.3. Longueur préservée

Une matrice unitaire préserve les longueurs, autrement dit si A est une matrice unitaire et u un vecteur alors $\|Au\| = \|u\|$. En fait cette particularité caractérise les matrices unitaires.

Proposition 10.

Soit $A \in M_2$. La matrice A est unitaire si et seulement pour tout vecteur u , on a

$$\|Au\| = \|u\|.$$



Démonstration.

- Sens \Rightarrow .

$$\|Au\|^2 = \langle Au|Au \rangle = \langle u|A^*Au \rangle = \langle u|u \rangle = \|u\|^2.$$

- Sens \Leftarrow .

Notons la matrice A sous la forme de ses vecteurs colonnes $A = \begin{pmatrix} u & v \end{pmatrix}$ et supposons qu'elle préserve les longueurs. Nous allons utiliser la caractérisation de la proposition 8.

— Comme $A \begin{pmatrix} 1 \\ 0 \end{pmatrix} = u$ et que A préserve les longueurs alors

$$\|A \begin{pmatrix} 1 \\ 0 \end{pmatrix}\| = \left\| \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\| \quad \text{donc} \quad \|u\| = 1.$$

— De même $A \begin{pmatrix} 0 \\ 1 \end{pmatrix} = v$, donc $\|v\| = 1$.

— D'une part $A \begin{pmatrix} 1 \\ 1 \end{pmatrix} = u + v$, donc $\|u + v\| = \sqrt{2}$. Ainsi :

$$\begin{aligned} \|u + v\|^2 = 2 &\implies \langle u + v | u + v \rangle = 2 \\ &\implies \langle u | u + v \rangle + \langle v | u + v \rangle = 2 \\ &\implies \langle u | u \rangle + \langle u | v \rangle + \langle v | u \rangle + \langle v | v \rangle = 2 \\ &\implies \|u\|^2 + \langle u | v \rangle + (\langle u | v \rangle)^* + \|v\|^2 = 2 \quad \text{mais } \|u\|^2 = 1 \text{ et } \|v\|^2 = 1 \\ &\implies 2 \operatorname{Re}(\langle u | v \rangle) = 0 \quad \text{sachant que } z + z^* = 2 \operatorname{Re}(z) \end{aligned}$$

— D'autre part $A \begin{pmatrix} 1 \\ i \end{pmatrix} = u + iv$, donc $\|u + iv\| = \sqrt{2}$. Ainsi :

$$\begin{aligned} \|u + iv\|^2 = 2 &\implies \langle u + iv | u + iv \rangle = 2 \\ &\implies \langle u | u + iv \rangle + \langle iv | u + iv \rangle = 2 \\ &\implies \langle u | u \rangle + i \langle u | v \rangle - i \langle v | u \rangle + \langle v | v \rangle = 2 \\ &\implies \|u\|^2 + i \langle u | v \rangle - i (\langle u | v \rangle)^* + \|v\|^2 = 2 \\ &\implies 2 \operatorname{Im}(\langle u | v \rangle) = 0 \quad \text{sachant que } z - z^* = 2i \operatorname{Im}(z) \end{aligned}$$

— On a prouvé que la partie réelle et la partie imaginaire de $\langle u | v \rangle$ sont nulles. Ainsi $\langle u | v \rangle = 0$.

— On a donc $\|u\| = 1$, $\|v\| = 1$ et $\langle u | v \rangle = 0$, alors par la proposition 8, la matrice $A = \begin{pmatrix} u & v \end{pmatrix}$ est unitaire. \square

6.4. Matrice spéciale unitaire

Parmi les matrices unitaires, celles dont le déterminant vaut 1 sont particulièrement intéressantes.

Définition.

Une matrice $A \in M_n$ est *spéciale unitaire* si elle est unitaire (c'est-à-dire $A^*A = I$) et de déterminant 1 :

$$\det(A) = 1.$$

On note SU_n l'ensemble des matrices spéciales unitaires de taille $n \times n$.

Exemple. Les matrices de Pauli (voir l'exemple plus haut) ne sont pas spéciales unitaires car de déterminant -1 , par contre en multipliant tous les coefficients par i , on obtient un déterminant $+1$, donc $iX, iY, iZ \in SU_2$.

Proposition 11.

L'ensemble des matrices spéciales unitaires forme un groupe pour la multiplication. En particulier si $A, B \in SU_n$ alors $AB \in SU_n$ et $A^{-1} \in SU_n$.

Démonstration. On sait déjà que AB et A^{-1} sont des matrices unitaires et que de plus $\det(AB) = \det(A) \det(B) = 1$ et $\det(A^{-1}) = \frac{1}{\det(A)} = 1$. \square

Dans le cas de matrices de taille 2×2 , nous décrivons l'ensemble des matrices de SU_2 .

Proposition 12.

Une matrice spéciale unitaire de taille 2×2 , s'écrit sous la forme

$$A = \begin{pmatrix} \alpha & -\beta^* \\ \beta & \alpha^* \end{pmatrix} \quad \text{avec } \alpha, \beta \in \mathbb{C} \text{ tels que } |\alpha|^2 + |\beta|^2 = 1.$$

Démonstration. Tout d'abord comme $A \in SU_2$ alors en particulier $A \in U_2$. D'après la proposition 8, A s'écrit sous la forme de ses vecteurs colonnes :

$$A = \begin{pmatrix} u & v \end{pmatrix} \quad \|u\| = 1 \quad \|v\| = 1 \quad \langle u | v \rangle = 0.$$

Notons $u = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$. Comme $\|u\| = 1$ alors $|\alpha|^2 + |\beta|^2 = 1$. Notons $v = \begin{pmatrix} \gamma \\ \delta \end{pmatrix}$. Comme u et v sont orthogonaux, car $\langle u|v \rangle = 0$, alors $\alpha^* \gamma + \beta^* \delta = 0$. Cela implique $\alpha^* \gamma = -\beta^* \delta$. Si $\alpha \neq 0$, on pose $\lambda = \frac{\delta}{\alpha^*}$. On a alors $\gamma = -\frac{\delta \beta^*}{\alpha^*} = -\lambda \beta^*$ et $\delta = \lambda \alpha^*$. (Si $\alpha = 0$ alors on a nécessairement $\beta \neq 0$ donc $\delta = 0$ et on a encore une relation $\gamma = -\lambda \beta^*$ et $\delta = \lambda \alpha^*$ avec $\lambda = -\frac{\gamma}{\beta^*}$).

Donc la matrice A s'écrit :

$$A = \begin{pmatrix} \alpha & -\lambda \beta^* \\ \beta & \lambda \alpha^* \end{pmatrix}.$$

Or

$$\det(A) = \lambda \alpha \alpha^* + \lambda \beta \beta^* = \lambda(|\alpha|^2 + |\beta|^2) = \lambda.$$

Comme $\det(A) = 1$, alors $\lambda = 1$. Ainsi :

$$A = \begin{pmatrix} \alpha & -\beta^* \\ \beta & \alpha^* \end{pmatrix} \quad \text{avec } |\alpha|^2 + |\beta|^2 = 1.$$

□

Terminons par une propriété, dite de **transitivité**. On peut transformer un vecteur en n'importe quel autre vecteur par une matrice spéciale unitaire, à condition que ces deux vecteurs aient la même longueur.

Proposition 13.

Soient $u \in \mathbb{C}^2$ et $v \in \mathbb{C}^2$ deux vecteurs avec $\|u\| = \|v\|$. Il existe une matrice $A \in SU_2$ telle que $Au = v$.

Une telle matrice A n'est pas unique.

Application. Si $|\phi\rangle$ et $|\psi\rangle$ sont de norme 1, alors il existe $A \in SU_2$ telle que $|\psi\rangle = A|\phi\rangle$.

Démonstration. Sans perte de généralité on suppose $\|u\| = \|v\| = 1$.

Étape 1. Il existe $B \in SU_2$ telle que $B \begin{pmatrix} 1 \\ 0 \end{pmatrix} = u$. En effet, si on note $u = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$ avec $\alpha, \beta \in \mathbb{C}$. Alors posons :

$$B = \begin{pmatrix} \alpha & -\beta^* \\ \beta & \alpha^* \end{pmatrix}$$

Comme $\|u\|^2 = |\alpha|^2 + |\beta|^2 = 1$, c'est bien une matrice spéciale unitaire : $B \in SU_2$.

Comme $B \in SU_2$, alors $B^{-1} \in SU_2$ et vérifie $B^{-1}u = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$.

Étape 2. On reprend la construction de la première étape pour construire cette fois $C \in SU_2$ telle que $C \begin{pmatrix} 1 \\ 0 \end{pmatrix} = v$.

Étape 3. La matrice $A = CB^{-1}$ convient. En effet, comme $B, C \in SU_2$ alors $CB^{-1} \in SU_2$ et

$$Au = (CB^{-1})u = C(B^{-1}u) = C \begin{pmatrix} 1 \\ 0 \end{pmatrix} = v.$$

□

Note. Certains passages de ce chapitre sont extraits du chapitre « Matrice » du livre « Algèbre » d'Exo7.

Informatique classique

Vidéo ■ partie 5.1. Bits classiques

Vidéo ■ partie 5.2. Portes logiques

Vidéo ■ partie 5.3. Algorithmes et complexité

Nous rappelons quelques principes de base du fonctionnement d'un ordinateur classique avec les notions de bits, de portes logiques et de complexité d'un algorithme.

1. Bits classiques

1.1. 0 ou 1

Un *bit* est une valeur 0 ou 1 et correspond à l'information minimale pour l'informatique classique. Cette valeur peut être codée par une information physique (allumé/éteint, 0 volts/5 volts, l'orientation magnétique d'un élément...).

1.2. Écriture binaire

Avec plusieurs bits on peut transmettre plus d'information. Voyons le cas d'un entier qui peut être représenté en écriture binaire.

Par exemple, 1.0.1.1.0.0.1 (prononcer les chiffres un par un) est l'écriture binaire de l'entier 89. Comment faire ce calcul ? C'est comme pour la base 10, mais en utilisant les puissances de 2.

1	0	1	1	0	0	1
64	32	16	8	4	2	1
2^6	2^5	2^4	2^3	2^2	2^1	2^0

Donc l'écriture **1.0.1.1.0.0.1** représente l'entier :

$$\begin{aligned}
 & 1 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \\
 &= 1 \times 64 + 0 \times 32 + 1 \times 16 + 1 \times 8 + 0 \times 4 + 0 \times 2 + 1 \times 1 \\
 &= 64 + 16 + 8 + 1 \\
 &= 89
 \end{aligned}$$

Plus généralement un *n-bit*, $b_{n-1} \dots b_2 \cdot b_1 \cdot b_0$, est l'écriture en base 2 de l'entier :

$$N = \sum_{i=0}^{n-1} b_i 2^i = b_{n-1} \cdot 2^{n-1} + \dots + b_2 \cdot 2^2 + b_1 \cdot 2 + b_0$$

avec $b_i = 0$ ou $b_i = 1$.

1.3. Codage

Il n'y a pas que les entiers qui peuvent être représentés par une succession de bits.

Une succession de bits peut aussi représenter :

- un caractère, par exemple le code ASCII de « A » est l'entier 65 et se code sur 8 bits par 0.1.0.0.0.0.0.1,
- une instruction, par exemple « Ajouter 1 », « Copier cette variable »,...

Plus généralement, on formalise un ordinateur par une *machine de Turing* qui est un modèle abstrait d'ordinateur qui lit et écrit des 0 et des 1 sur un ruban.

1.4. Logarithme

Avec n bits, il y a 2^n combinaisons possibles. Si on a N objets à énumérer, alors combien de bits faut-il pour les énumérer ?

On va utiliser le *logarithme en base 2* qui est défini par la relation :

$$N = 2^n \iff n = \log_2(N).$$

On peut également le définir à l'aide du logarithme népérien $\ln(x)$:

$$\log_2(x) = \frac{\ln(x)}{\ln(2)}.$$

Ainsi pour énumérer N objets, il faut au moins $n = \log_2(N)$ bits d'information. Dans la pratique on arrondit à l'entier supérieur : $n = \lceil \log_2(N) \rceil$ où $\lceil x \rceil$ correspond à l'arrondi supérieur, comme le ferait `ceil(x)` avec *Python*.

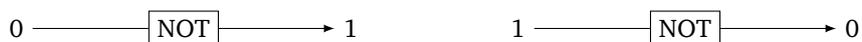
2. Portes logiques

2.1. Quelques portes

Une *porte logique* est une fonction qui prend en entrée des bits et renvoie un bit en sortie. D'un point de vue mathématique, c'est donc une fonction $f : \{0, 1\}^n \rightarrow \{0, 1\}$, où $n \geq 1$. D'un point de vue électronique, cette porte peut être réalisée par des composants élémentaires (diode, transistor,...).

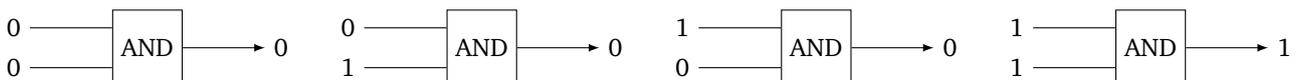
On présente ici quelques portes classiques. Il faut penser à la valeur 0 comme « Faux » et à la valeur 1 comme « Vrai ».

Porte NOT. Elle a une seule entrée et change 0 en 1, et 1 en 0.



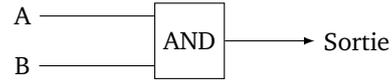
Entrée	Sortie
0	1
1	0

Porte AND.

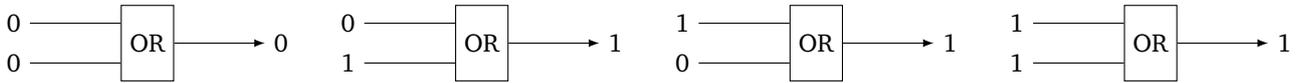


On résume l'action de la porte AND par le tableau suivant :

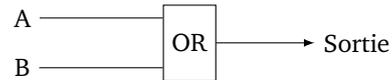
A	B	Sortie
0	0	0
0	1	0
1	0	0
1	1	1



Porte OR.

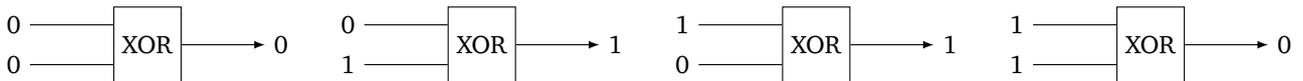


A	B	Sortie
0	0	0
0	1	1
1	0	1
1	1	1



Porte XOR.

Le « OU exclusif » c'est le « ou » dans « fromage ou dessert », c'est l'un ou l'autre mais pas les deux.



On résume l'action de la porte XOR par le tableau suivant :

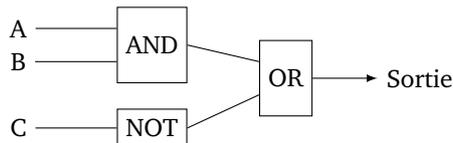
A	B	Sortie
0	0	0
0	1	1
1	0	1
1	1	0



2.2. Circuit

Un *circuit* est construit à partir de plusieurs portes logiques et définit une *fonction logique* : $f : \{0, 1\}^n \rightarrow \{0, 1\}$, où $n \geq 1$.

Exemple. Voici un exemple de circuit :



À vous de compléter le tableau donnant la valeur de sortie en fonction de celles de départ !

A	B	C	Sortie
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

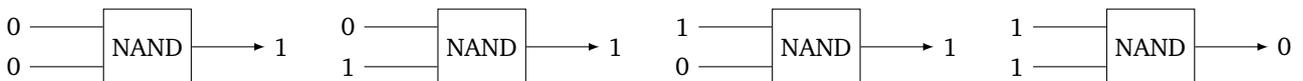
Combinatoire. Soit $n \geq 1$ le nombre d'entrées d'un circuit. Une entrée est du type $(b_1, b_2, \dots, b_n) \in \{0, 1\}^n$. Il y a donc 2^n entrées différentes possibles.

On rappelle que pour E de cardinal m et F de cardinal p , le nombre de fonctions $f : E \rightarrow F$ possibles est p^m . Dans notre cas, on considère toutes les fonctions $f : \{0, 1\}^n \rightarrow \{0, 1\}$, donc $m = 2^n$ et $p = 2$, il y donc a $2^{(2^n)}$ fonctions logiques possibles.

Ce nombre $N = 2^{(2^n)}$ de fonctions possibles est énorme. Par exemple, pour $n = 5$ entrées il y a $N = 2^{(2^5)} = 2^{32} = 4\,294\,967\,296$ fonctions logiques possibles. Faut-il autant de portes logiques différentes pour réaliser ces fonctions? Ce serait dramatique pour nos ordinateurs qui devraient être gigantesques. On va voir qu'un miracle se produit : une seule porte logique suffit à réaliser toutes les fonctions logiques possibles.

2.3. La porte NAND est universelle

Porte NAND. La porte NAND est définie par les quatre états suivants.



A	B	Sortie
0	0	1
0	1	1
1	0	1
1	1	0



Elle peut se réaliser comme la porte NOT(AND), c'est-à-dire la porte AND suivie de la porte NOT.



Mais nous préférons la voir comme une porte à part entière.

Elle a la propriété fondamentale suivante :

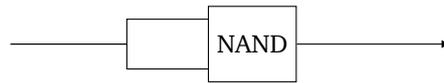
Théorème 1.

La porte NAND est universelle : n'importe quelle fonction logique $f : \{0, 1\}^n \rightarrow \{0, 1\}$ peut-être réalisée par un circuit ne comportant que des portes NAND.

Nous n'allons pas prouver ce théorème, mais nous allons voir comment retrouver les portes de base que nous connaissons à partir de portes NAND.

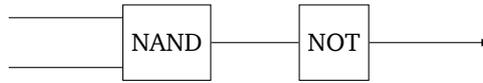
Réalisation des portes élémentaires.

On commence par réaliser une porte NOT en dupliquant l'unique entrée aux deux bornes de la borne NAND.



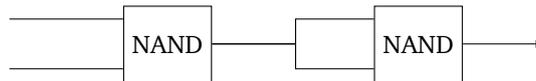
Équivalent porte NOT

Maintenant que l'on est capable de construire une porte NOT, on peut réaliser et retrouver la porte AND.



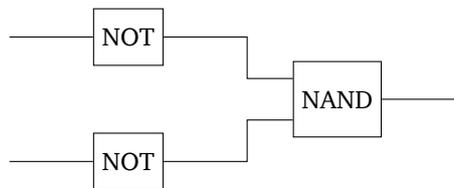
Équivalent porte AND

Si vous ne voulez vraiment que des portes NAND, il faut remplacer la porte NOT par une seconde porte NAND.



Équivalent porte AND

On peut aussi faire un circuit qui correspond à la porte OR.

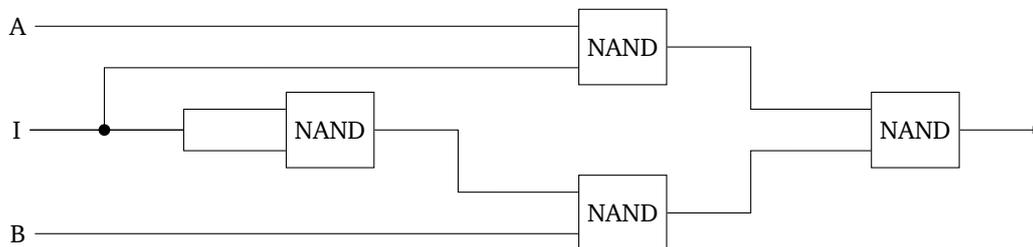


Équivalent porte OR

À vous de trouver comment réaliser une porte XOR.

Exercice.

Un circuit multiplexeur, noté MUX, est une sorte d'aiguillage entre deux choix A et B avec un interrupteur I qui permet la sélection. Si I vaut 1 alors la sortie du circuit est la valeur de A . Si I vaut 0 alors la sortie est la valeur de B . Cela peut aussi être vu comme un test rudimentaire : « Si $I = 1$ faire ceci, sinon faire cela. » Prouver que le circuit suivant est bien un circuit multiplexeur.



Circuit MUX

3. Algorithme et complexité

3.1. Vitesse d'un algorithme

Un algorithme c'est un peu comme une recette de cuisine : c'est une suite d'instructions qui permettent de résoudre un problème. Mais pour un même problème il peut exister plusieurs solutions. Il peut y avoir des algorithmes rapides, d'autres qui utilisent peu de mémoire, d'autres qui sont efficaces mais qui ne donnent la bonne réponse qu'avec 90% de certitude.

Voici un exemple de problème : décider si un entier n donné est un nombre premier.

- Une première méthode consiste à tester s'il admet un diviseur k , pour k variant de 2 jusqu'à $n - 1$. Si c'est le cas n n'est pas un nombre premier, sinon c'est bien un nombre premier.
- On peut aller beaucoup plus vite en testant seulement les diviseurs k compris entre 2 et \sqrt{n} .
- On pourrait d'abord établir le début de la liste de tous les nombres premiers par le crible d'Ératosthène, puis tester si n est dedans.
- Il existe aussi des algorithmes probabilistes très efficaces, qui donnent une réponse du genre « oui n est probablement un nombre premier à 90% » ou « non je suis sûr que n n'est pas un nombre premier ». En répétant l'algorithme on peut obtenir une quasi-certitude.

Nous comparerons les algorithmes en nous limitant ici à la « vitesse » des algorithmes. La vitesse que l'on chronométrerait en secondes n'est pas une mesure objective car elle dépend trop du matériel et de l'implémentation. Nous allons étudier la vitesse théorique qui s'appelle la « complexité ». Cela demande d'introduire des notions mathématiques.

3.2. Notation « grand O »

On souhaite comparer deux suites, ou plus exactement leur ordre de grandeur. Par exemple les suites $(n^2)_{n \in \mathbb{N}}$ et $(3n^2)_{n \in \mathbb{N}}$ ont le même ordre de grandeur, mais sont beaucoup plus petites que la suite $(\frac{1}{2}e^n)_{n \in \mathbb{N}}$.

Notation « grand O ».

- On considère $(u_n)_{n \in \mathbb{N}}$ et $(v_n)_{n \in \mathbb{N}}$ deux suites de termes strictement positifs.
- On dit que (u_n) est un **grand O** de (v_n) si la suite $(\frac{u_n}{v_n})$ est bornée.
- Autrement dit il existe une constante réelle $k > 0$ telle que pour tout $n \in \mathbb{N}$:

$$u_n \leq kv_n.$$

- *Notation.* On note alors $u_n = O(v_n)$. Il s'agit de la lettre « O » (pour Ordre de grandeur) et pas du chiffre zéro.

Exemples

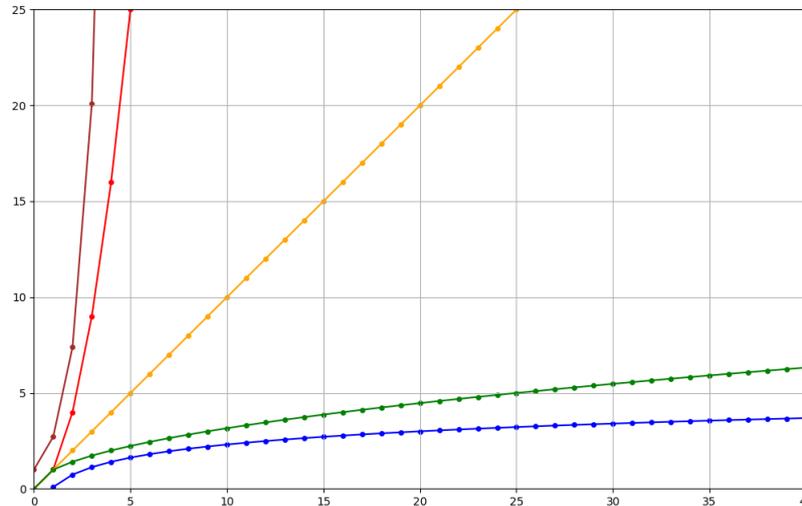
- Soient $u_n = 3n + 1$ et $v_n = 2n - 1$. Comme $\frac{u_n}{v_n} \rightarrow \frac{3}{2}$ lorsque $n \rightarrow +\infty$ alors la suite $(\frac{u_n}{v_n})$ est bornée donc $u_n = O(v_n)$.
- $u_n = 2n^2$ et $v_n = e^n$. Comme $\frac{u_n}{v_n} \rightarrow 0$ alors la suite $(\frac{u_n}{v_n})$ est bornée donc $u_n = O(v_n)$.
- $u_n = \sqrt{n}$ et $v_n = \ln(n)$. Comme $\frac{u_n}{v_n} \rightarrow +\infty$ lorsque $n \rightarrow +\infty$ alors la suite $(\frac{u_n}{v_n})$ n'est pas bornée. (u_n) n'est pas un grand O de (v_n) . Par contre dans l'autre sens, on a bien $v_n = O(u_n)$.
- $u_n = O(n)$ signifie qu'il existe $k > 0$ tel que $u_n \leq kn$ (pour tout $n \in \mathbb{N}$).
- $u_n = O(1)$ signifie que la suite (u_n) est bornée.

3.3. Suites de référence

On souhaite comparer une suite (u_n) avec des suites de référence. Voici les suites de référence choisies :

$$\underbrace{\ln(n)}_{\text{croissance logarithmique}} \quad \underbrace{n \quad n^2 \quad n^3 \quad \dots}_{\text{croissance polynomiale}} \quad \underbrace{e^n}_{\text{croissance exponentielle}}$$

- Les suites sont écrites en respectant l'ordre des O : on a $\ln(n) = O(n)$, $n = O(n^2)$, $n^2 = O(n^3)$, $n^3 = O(e^n)$.
- On pourrait intercaler d'autres suites, par exemple $\ln(n) = O(\sqrt{n})$ et $\sqrt{n} = O(n)$. Ou encore $n \ln(n) = O(n^2)$.



Les suites $\ln(n)$, \sqrt{n} , n , n^2 , e^n .

3.4. Complexité d'un algorithme

On mesure l'efficacité d'un algorithme à l'aide de la complexité.

Nous définissons de manière informelle la complexité : la **complexité** d'un algorithme est le nombre d'opérations élémentaires exécutées.

- Ce que l'on appelle « opération élémentaire » peut varier selon le contexte : pour un calcul cela peut être le nombre de multiplications, pour un tri le nombre de comparaisons. . .
- La complexité C_n dépend de la taille n des données en entrée (par exemple le nombre de chiffres d'un entier ou bien la longueur de la liste). On obtient ainsi une suite (C_n) .
- Les bons algorithmes ont des complexités polynomiales qui sont en $O(n)$ (linéaire), en $O(n^2)$ (quadratique) ou bien en $O(n^k)$, $k \in \mathbb{N}^*$ (polynomiale). Les mauvais algorithmes ont des complexités exponentielles, en $O(e^n)$ par exemple.

3.5. Exemples de complexité

Multiplication de deux entiers.

On souhaite multiplier deux entiers a et b de n chiffres. Il y a plusieurs méthodes, on les compare en comptant le nombre d'opérations élémentaires : ici les multiplications de petits nombres (entiers à 1 ou 2 chiffres).

Algorithme	Ordre de la complexité
Multiplication d'école	$O(n^2)$
Multiplication de Karatsuba	$O(n^{\log_2(3)}) \simeq O(n^{1.58})$
Transformée de Fourier rapide	$O(n \cdot \ln(n) \cdot \ln(\ln(n)))$

Voici des exemples d'ordre de grandeur de la complexité pour différentes valeurs de n .

Algorithme	$n = 10$	$n = 100$	$n = 1000$
Multiplication d'école	100	10 000	1 000 000
Multiplication de Karatsuba	38	1478	56 870
Transformée de Fourier rapide	19	703	13 350

Plus l'entier n est grand, plus un bon algorithme prend l'avantage.

On termine par des exemples de problèmes et d'algorithmes correspondant à différentes classes de complexité.

Complexité	Problème et algorithme
$O(1)$	accès à un élément d'une liste de taille n test si un nombre est pair ou impair
$O(\log_2(n))$	recherche par dichotomie dans une liste ordonnée de taille n
$O(n)$	recherche d'un maximum dans une liste non ordonnée de taille n recherche si un élément est présent dans une liste de taille n
$O(n \log_2(n))$	tri d'une liste de taille n par <i>mergesort</i>
$O(n^2)$	tri d'une liste de taille n par <i>bubble sort</i> , <i>selection sort</i> , <i>insertion sort</i> recherche si un élément est présent en double dans une liste de taille n
$O(2^n)$	problème du voyageur de commerce avec n villes trouver tous les sous-ensembles de $\{1, 2, \dots, n\}$

La dernière catégorie d'algorithmes dont la complexité est en $O(2^n)$ est de complexité exponentielle (car $2^n = e^{n \ln 2}$). Ce qui signifie que pour des valeurs de n moyennes ou grandes, ces algorithmes sont inutilisables car ils n'aboutiront pas dans un temps raisonnable.

Note. Certains passages de cette section sont extraits du chapitre « Tri – Complexité » du livre « Python au lycée (tome 2) ».

Physique quantique

Vidéo ■ partie 6.1. Particule

Vidéo ■ partie 6.2. Dualité onde/corpuscule

Vidéo ■ partie 6.3. Fonction d'onde et équation de Schrödinger

Vidéo ■ partie 6.4. Qubits

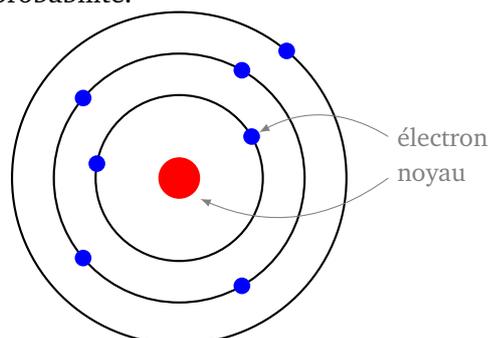
L'objectif est de comprendre les notions de base de la physique quantique.

1. Particule

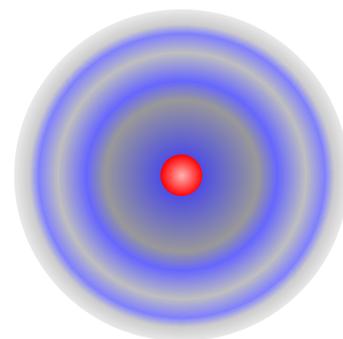
1.1. Bestiaire

Commençons par un tour d'horizon des particules :

- Les **protons** (charge électrique positive) et les **neutrons** (pas de charge) constituent le noyau des atomes.
- Autour du noyau gravitent des **électrons** (charge négative).
- Un modèle simple de structure de l'atome est décrit par les électrons qui se répartissent sur des couches sphériques ayant pour centre le noyau, ce modèle est maintenant désuet. Avec la mécanique quantique, on considère qu'un électron peut être à n'importe quelle position autour du noyau, mais pas partout avec la même probabilité.



Modèle (désuet) de couches d'électrons

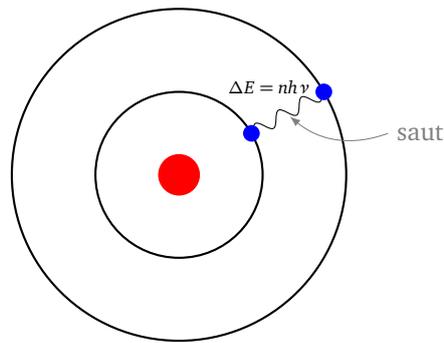


Modèle quantique

- Le **photon** est la particule fondamentale de la lumière (et des ondes électromagnétiques). Il n'a ni masse ni charge.

1.2. Quantification

Des expériences montrent que le changement d'état des électrons dans un atome correspond à des niveaux d'énergie bien déterminés. On le conçoit bien avec le modèle dans lequel les électrons gravitent sur des couches : pour qu'un électron passe à une couche supérieure, il faut lui fournir un certain niveau d'énergie.



Changement de couche

La formule qui calcule l'énergie à fournir est donnée par :

$$\Delta E = E_{\text{après}} - E_{\text{avant}} = nh\nu$$

où n est un entier, ν est la fréquence du rayonnement reçu (ou émis) (exprimée en s^{-1}) et h est la **constante de Planck**, $h \simeq 6.62 \cdot 10^{-34} J \cdot s$.

Il y a donc un phénomène de **quantification**, car l'énergie reçue (ou émise) ne peut prendre que des valeurs discrètes, c'est-à-dire que l'on peut indexer par l'entier n qui vaut 1, 2, 3... et non par des valeurs continues (comme on l'aurait avec un paramètre réel $t \geq 0$).

La **formule de Bohr-Einstein** lie l'énergie à la fréquence :

$$E = h\nu$$

On rencontre aussi souvent la **constante de Planck réduite** :

$$\hbar = \frac{h}{2\pi}$$

1.3. Principe d'incertitude d'Heisenberg

Le **principe d'incertitude d'Heisenberg** s'énonce mathématiquement ainsi :

$$\sigma_x \cdot \sigma_v \geq \frac{h}{4\pi m}$$

où :

- σ_x est l'écart-type d'une série de mesures de la position $\mathbf{x} = (x, y, z)$ d'une particule. Cet écart-type est obtenu en répétant plusieurs fois l'expérience suivante : on prépare la particule, toujours avec le même état initial, puis on mesure sa position.
- σ_v est l'écart-type de la mesure de la vitesse de la particule $\mathbf{v} = (v_x, v_y, v_z)$.

L'interprétation de la formule est la suivante : si on connaît avec une grande précision la position d'une particule alors on ne peut pas connaître très précisément sa vitesse. Et réciproquement si on connaît très précisément la vitesse, on ne peut pas connaître très précisément la position.

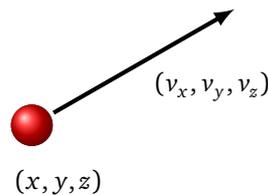
En effet, imaginons que l'on connaisse très précisément la position d'une particule, alors lorsque l'on va répéter la mesure de la position, on obtiendra presque toujours la même mesure. Autrement dit, les écarts entre les différentes mesures sont tout petits : $\sigma_x \simeq 0$. Par le principe d'incertitude, $\sigma_v \geq \frac{h}{4\pi m \sigma_x}$ doit être très grand. C'est-à-dire que les différentes mesures de la vitesse conduisent à des grands écarts.

L'incertitude est d'autant plus grande que la masse m est petite. Par contre la constante de Planck h étant toute petite ($h \simeq 6.62 \cdot 10^{-34} J \cdot s$), cette incertitude ne concerne vraiment que les particules et pas les objets plus gros de la physique classique.

2. Dualité onde/corpuscule

En physique classique, on sépare l'étude des corpuscules (un objet matériel comme une bille) de celle des ondes (par exemple une onde sonore ou bien la lumière). En physique quantique, une particule se comporte à la fois comme une onde et un corpuscule. C'est la très déroutante « dualité onde/corpuscule ».

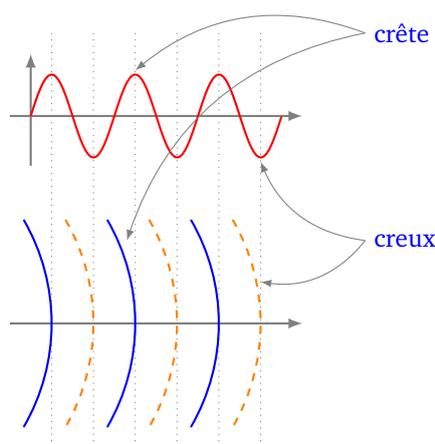
Corpuscule. Un « corpuscule » est un petit objet aux propriétés physiques bien déterminées : comme sa position (x, y, z) , sa vitesse (v_x, v_y, v_z) , sa masse... Il peut avoir une taille mais est souvent modélisé par un point. La mécanique d'un point matériel est bien connue, même si on a vu, avec le principe d'incertitude d'Heisenberg, que la mesure de la position et celle de la vitesse ne sont pas si évidentes que cela !



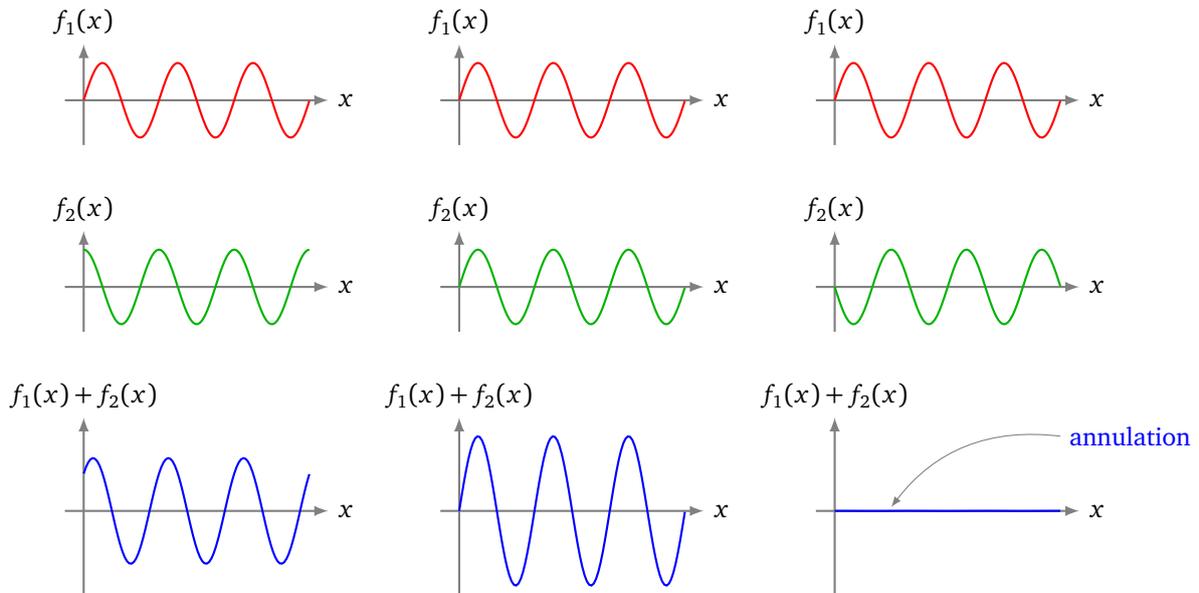
Onde. Une « onde » est la variation d'une propriété physique par propagation. Par exemple, une onde sonore est la variation de la pression : un mouvement initial fait que des molécules d'air viennent frapper les molécules voisines qui à leur tour déplacent les molécules suivantes. Pourtant chaque molécule ne se déplace presque pas (elle revient à sa place après avoir rebondi sur sa voisine) par contre l'onde se déplace. Une vague est un autre exemple d'onde, les molécules d'eau font varier la hauteur. Pourtant les molécules d'eau ne se déplacent pas en suivant les vagues. D'ailleurs un bouchon flottant sur les vagues ne se déplace pas horizontalement.

Les ondes électromagnétiques se déplacent dans le vide à la vitesse de la lumière ($300\,000\text{ km} \cdot \text{s}^{-1}$). Le courant électrique se déplace dans un fil de cuivre un peu moins vite ($200\,000\text{ km} \cdot \text{s}^{-1}$). Par contre les électrons qui propagent ce courant se déplacent très lentement (quelques centimètres par heure).

Voici deux vues d'une onde sinusoïdale : la première vue en une vue en coupe (on voit les vagues avec les crêtes et les creux), la seconde vue est une vue de dessus, comme si on avait jeté un caillou dans l'eau et qu'on observe les lignes de crête circulaires.



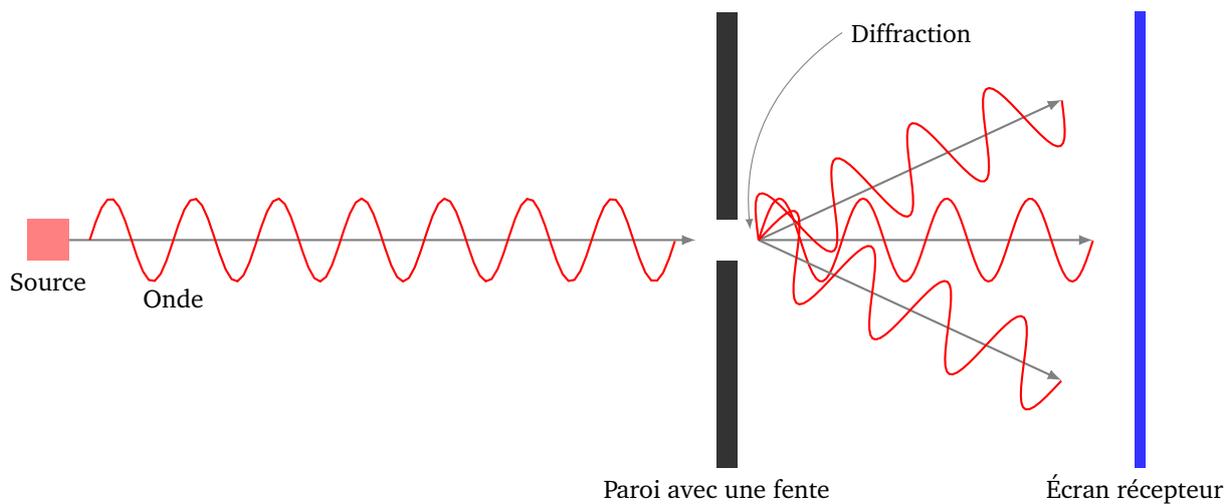
Une des propriétés fondamentales des ondes est le principe de superposition. Deux ondes se superposent (figure de gauche), ce qui peut conduire à une onde renforcée (figure centrale), ou bien à annulation (figure de droite).



Nous allons étudier les phénomènes d'interférence à l'aide de l'expérience de Young à deux fentes, mais commençons par un cas plus simple.

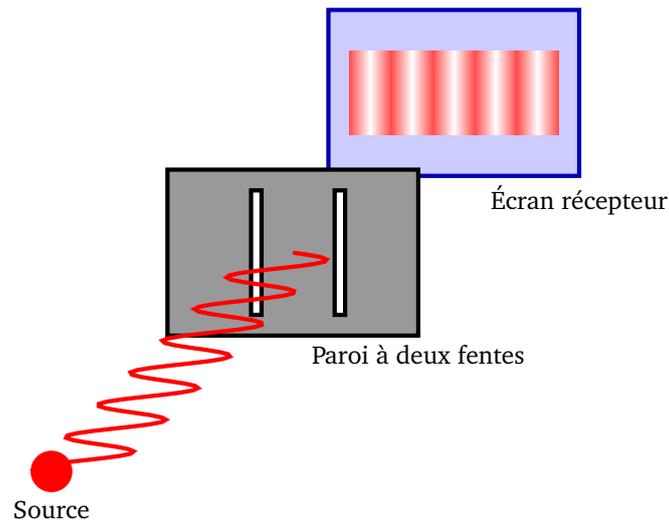
Expérience à une fente. Une onde est émise d'un point. Elle se propage vers une paroi opaque dans laquelle on a effectué une fente mince (ou bien un petit trou). En passant cette fente l'onde se diffracte (elle perd sa direction d'origine) et vient éclairer un écran récepteur. Que voit-on sur cet écran ? L'onde frappe l'écran de manière essentiellement uniforme. En réalité la partie centrale est plus touchée que les bords et fait apparaître une large bande de diffraction, mais ce n'est pas important pour nous.

Ci-dessous le résultat d'une expérience à une fente à l'aide d'une lumière laser.

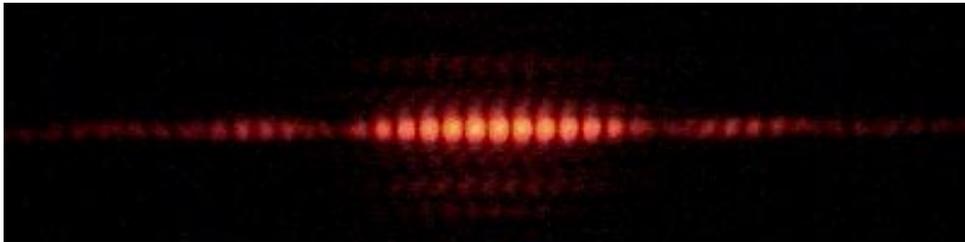


Expérience à deux fentes.

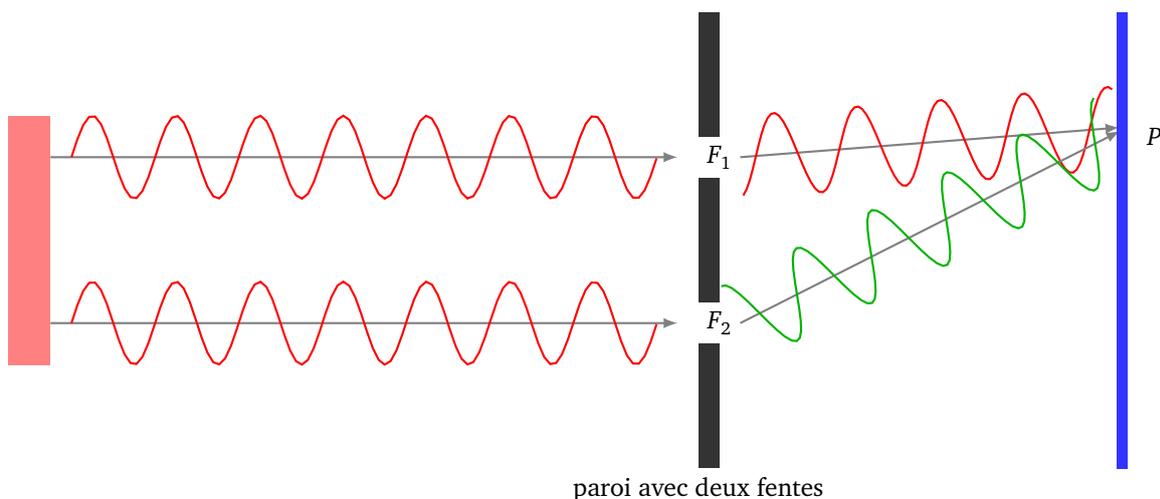
On reprend l'expérience mais cette fois avec deux fentes.



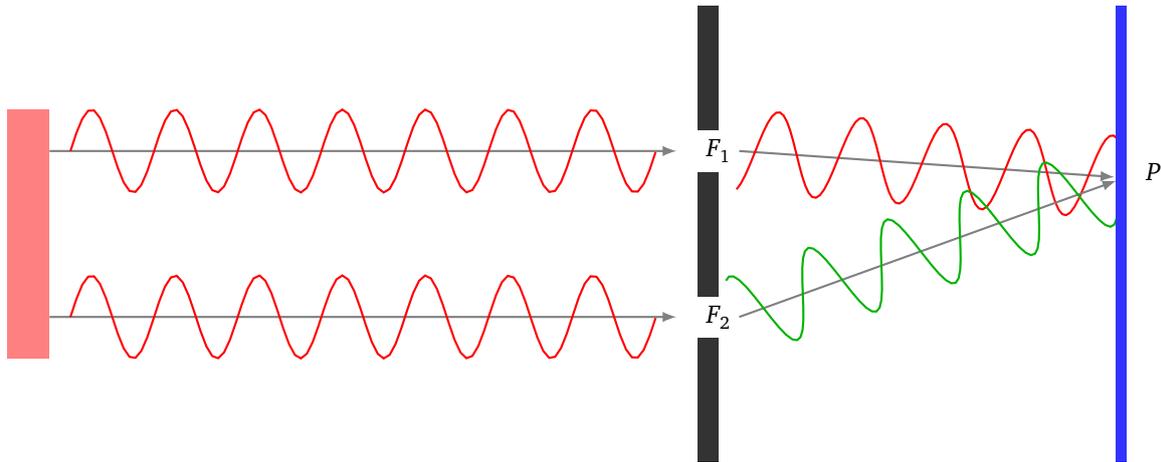
L'onde passe simultanément par les deux trous et se diffracte. Un point de l'écran est donc atteint depuis deux sources différentes. Que voit-on sur l'écran ? On note des franges minces et nettes qui sont des franges d'interférence.



Comment expliquer cela ? Plaçons-nous en un point P de l'écran. Ce point capte l'onde depuis deux sources différentes F_1 et F_2 , mais ces deux sources ne sont pas situées à la même distance de P : $PF_1 \neq PF_2$. Il y a donc un décalage (appelé déphasage) entre l'onde reçue depuis F_1 et l'onde reçue depuis F_2 . Selon la position du point P , les ondes peuvent s'amplifier ce qui conduit aux franges claires les plus atteintes. C'est ce qu'il se passe ci-dessous au point P où les deux sinusoides arrivent en phase (les deux crêtes arrivent en même temps au point P , et au fil du temps les ondes restent en phase).



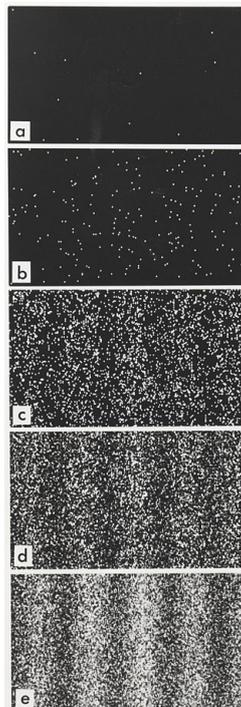
Mais les ondes peuvent s'annuler ce qui correspond aux franges sombres (rien n'est capté). C'est le cas au point de la configuration ci-dessous. En ce nouveau point P une sinusoïde présente une crête alors que l'autre sinusoïde est à un creux (de même amplitude), la résultante des deux est nulle (et reste nulle au fil du temps).



Mathématiquement si le signal issu de F_1 est $f_1(t) = \sin(\omega t)$ alors le signal issu de F_2 est $f_2(t) = \sin(\omega t + \phi_P)$ où ϕ_P est le **déphasage**, sa valeur dépend de la position du point P . Si ϕ_P est de la forme $2k\pi$ alors $f_2(t) = \sin(\omega t)$ et donc les ondes s'ajoutent pour donner $f(t) = f_1(t) + f_2(t) = 2 \sin(\omega t)$. Par contre si ϕ_P est de la forme $\pi + 2k\pi$ alors $f_2(t) = -\sin(\omega t)$ et donc les ondes s'annulent pour donner $f(t) = f_1(t) + f_2(t) = 0$.

Que se passe-t-il si on réalise l'expérience à deux fentes avec un corpuscule (par exemple en envoyant une multitude de petites billes) au lieu d'une onde? Chaque corpuscule passe par la fente 1 ou bien par la fente 2 (mais pas les deux) et il y a un phénomène de diffraction à chaque fente. Que voit-on à l'écran? L'écran est uniformément atteint. Il n'y a pas de phénomène d'interférence puisque une bille ne passe que par un seul trou.

Expérience quantique à deux fentes. Voici l'expérience incroyable : on envoie une à une des particules à travers le dispositif à deux fentes. Ces particules sont par exemple des photons ou des électrons. Voici ce que l'on obtient à l'écran au fur et à mesure des lancers (ici des électrons). Nombre d'électrons captés (a) 11; (b) 200, (c) 6000, (d) 40 000, (e) 140 000.

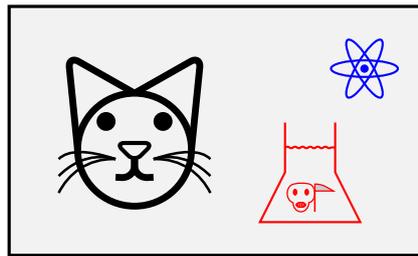


On observe des franges d'interférence alors qu'on a envoyé les particules une par une ! Ces particules sont bien des corpuscules puisque qu'on les envoie une par une et on peut même les compter. Mais les franges d'interférence prouvent que ces particules sont aussi des ondes ! Ainsi on ne peut parler de trajectoire pour

une particule quantique. Une particule quantique possède à la fois des propriétés corpusculaires et des propriétés ondulatoires : c'est la « dualité onde/corpuscule ».

Mesure. Si on reprend l'expérience quantique à deux fentes mais que cette fois on mesure par quel trou la particule passe, alors on obtient un écran uniformément atteint ! Il faut comprendre que la mesure perturbe irrémédiablement l'état quantique et la particule perd ses caractéristiques quantiques. Le fait de rajouter une mesure entraîne que ce n'est pas du tout la même expérience qu'auparavant.

Le chat de Schrödinger. Dans le monde dans lequel nous vivons les objets sont « gros » et leur appliquer des propriétés quantiques n'a pas trop de sens. C'est pourtant ce que l'on fait avec l'expérience de pensée du chat de Schrödinger. Dans une boîte recouverte d'un voile noir, on met un chat, un flacon de poison et une particule. Si la particule se désintègre, elle casse la fiole qui libère le poison et le chat meurt.



La particule a une demi-vie d'une heure, c'est-à-dire qu'au bout d'une heure, il y a une chance sur deux qu'elle se soit désintégrée. Question : au bout d'une heure le chat est-il mort ou vivant ? Ou bien moitié-mort et moitié-vivant ? En fait, tant que le voile noir n'est pas levé le chat n'est pas vraiment mort. Par contre lever le voile (ce qui correspond à une mesure) enlève le caractère mi-mort/mi-vivant au chat et rend l'état du chat certain. C'est assez perturbant !

On retient que la mécanique quantique est une théorie qui s'applique aux particules de petite taille, et souvent sur de courtes périodes.

3. Fonction d'onde et équation de Schrödinger

3.1. Fonction d'onde

Fonction d'onde. Le mouvement d'une particule en physique classique est décrit par sa position (x, y, z) (qui peut varier en fonction du temps) et sa vitesse (v_x, v_y, v_z) . Ce modèle n'est plus valide en physique quantique.

En physique quantique on associe à une particule une **fonction d'onde** Ψ :

$$\Psi(x, y, z, t) : \mathbb{R}^4 \longrightarrow \mathbb{C}.$$

Cette fonction dépend du point de l'espace (x, y, z) et du temps t ; elle est à valeurs complexes et n'a pas d'interprétation physique.

Probabilité. Par contre la fonction d'onde permet de déterminer la probabilité de la particule d'être à une position donnée : la probabilité que la particule soit mesurée à la position (x, y, z) au temps t est donnée par la densité de probabilité :

$$|\Psi(x, y, z, t)|^2.$$

Autrement dit :

$$dP(x, y, z, t) = |\Psi(x, y, z, t)|^2 dx dy dz$$

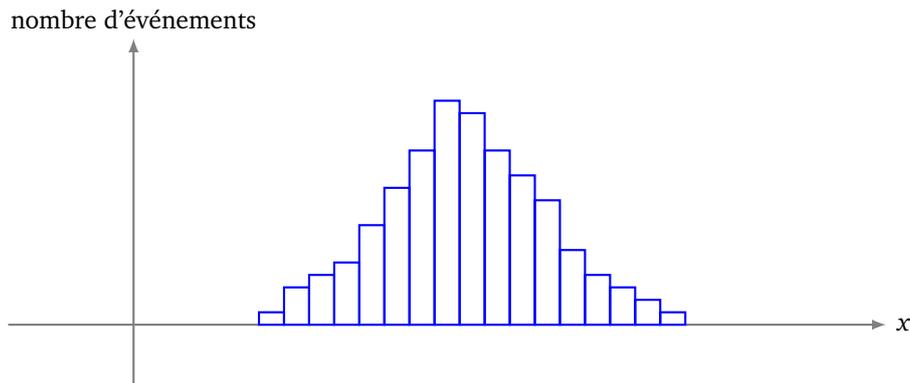
où $dx dy dz$ représente un petit cube élémentaire autour de la position (x, y, z) . Nous avons la relation :

$$\int |\Psi(x, y, z, t)|^2 dx dy dz = 1$$

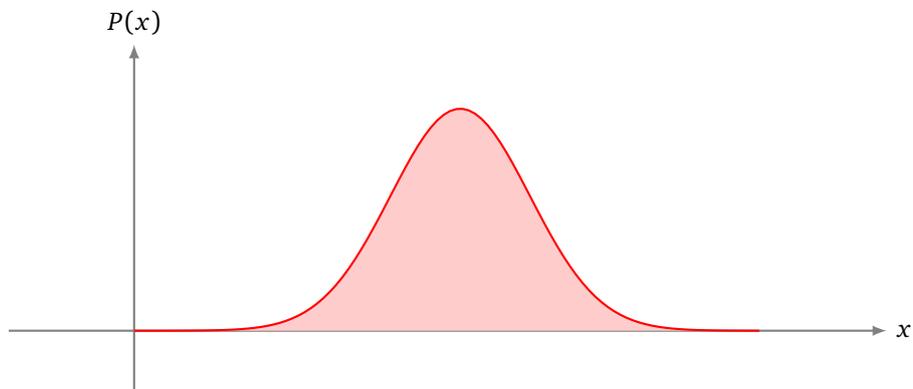
qui signifie que la somme des probabilités (pour toutes les positions possibles) est 1.

Mesure de la position. Il faut bien comprendre que l'on ne connaît pas la position de la particule. Ce n'est pas un défaut ou un manque de précision mais la base de la mécanique quantique. La notion de position n'a pas de sens avant la mesure. On pourrait dire que la particule est « partout », et que c'est la mesure qui détermine une position (x, y, z) . Enfin, même si la particule peut être mesurée à n'importe quelle position, certaines positions sont cependant plus probables que d'autres.

Voici un exemple de répartition des mesures d'une position x (on se limite à la mesure de la seule coordonnée x) obtenue en répétant l'expérience : je prépare la particule, puis je mesure la position x . On note que certaines positions sont plus fréquentes que d'autres.



Mathématiquement les résultats s'interprètent avec la densité de probabilité $P(x) = |\Psi(x)|^2$ qui décrit la répartition probable. L'aire sous la courbe vaut 1, car $\int |\Psi(x)|^2 dx = 1$ (ce qui est la traduction que la somme des probabilités vaut 1).



3.2. Équation de Schrödinger

La loi qui régit le mouvement d'une particule classique est le principe fondamental de la mécanique :

$$\sum \vec{F} = m\vec{a}.$$

La somme des forces est égale à la masse multipliée par l'accélération. L'accélération est la dérivée de la vitesse par rapport au temps, qui elle-même est la dérivée de la position par rapport au temps. Ainsi on n'obtient pas exactement la position $\mathbf{x} = (x(t), y(t), z(t))$ mais une équation différentielle faisant intervenir la dérivée seconde $\frac{d^2\mathbf{x}}{dt^2}(t)$. Si on sait résoudre cette équation différentielle (ce qui n'est pas toujours possible), on trouve la position \mathbf{x} .

Qu'en est-il pour la mécanique quantique ? On a vu que le comportement d'une particule est régi par sa fonction d'onde $\Psi(x, y, z, t)$. La loi quantique fondamentale qui régit le comportement d'une particule libre est donnée par *l'équation de Schrödinger* :

$$\boxed{i\frac{\partial\Psi}{\partial t} = -\frac{\hbar}{2m}\Delta\Psi}$$

où :

- i est le nombre complexe avec $i^2 = -1$,
- $\hbar = \frac{h}{2\pi}$ est la constante de Planck réduite,
- m est la masse de la particule,
- $\Psi(x, y, z, t)$ est la fonction d'onde,
- $\frac{\partial\Psi}{\partial t}$ est la dérivée de la fonction d'onde par rapport au temps (on parle de dérivée partielle par rapport à t),
- $\Delta\Psi$ est le Laplacien de Ψ , c'est la somme des dérivées partielles secondes en x , y et z :

$$\Delta\Psi = \frac{\partial^2\Psi}{\partial x^2} + \frac{\partial^2\Psi}{\partial y^2} + \frac{\partial^2\Psi}{\partial z^2}.$$

L'équation de Schrödinger est une équation différentielle (ici une équation aux dérivées partielles) qui régit Ψ et donc le comportement de la particule. Par contre, en trouver des solutions est difficile. Si de plus la particule est soumise à des forces externes, données par un potentiel $V(x, y, z, t)$, alors l'équation de Schrödinger devient :

$$i\hbar\frac{\partial\Psi}{\partial t}(x, y, z, t) = -\frac{\hbar^2}{2m}\Delta\Psi(x, y, z, t) + V(x, y, z, t)\Psi(x, y, z, t).$$

3.3. Principe de superposition

Il est facile de vérifier que si Ψ_1 et Ψ_2 sont deux fonctions d'onde, solutions de l'équation de Schrödinger alors la combinaison linéaire

$$\Psi = \alpha\Psi_1 + \beta\Psi_2 \quad \text{avec } \alpha, \beta \in \mathbb{C}$$

est aussi une solution (il faut tout de même vérifier la condition de normalisation $\int |\Psi|^2 = 1$).

Par contre en termes de probabilités, c'est plus compliqué. Si on note $P_1 = |\Psi_1|^2$ et $P_2 = |\Psi_2|^2$ les densités de probabilité associées aux deux fonctions d'onde et avec par exemple $\Psi = \frac{1}{\sqrt{2}}(\Psi_1 + \Psi_2)$, alors on pourrait penser que la densité de probabilité $P = |\Psi|^2$ est $\frac{1}{2}(P_1 + P_2)$ mais ce n'est pas le cas. En effet :

$$\begin{aligned} P = |\Psi|^2 &= |\Psi_1 + \Psi_2|^2 = \frac{1}{2}(\Psi_1 + \Psi_2)(\Psi_1 + \Psi_2)^* \\ &= \frac{1}{2}(|\Psi_1|^2 + |\Psi_2|^2 + \Psi_1\Psi_2^* + \Psi_1^*\Psi_2) = \frac{1}{2}(P_1 + P_2 + \Psi_1\Psi_2^* + \Psi_1^*\Psi_2). \end{aligned}$$

Le terme $\Psi_1\Psi_2^* + \Psi_1^*\Psi_2$ est un terme d'interférence entre les deux fonctions d'ondes.

4. Qubits

4.1. Réalisation de qubits

Il existe de nombreuses façons de réaliser physiquement un qubit, chacune présentant ses avantages et ses défis technologiques. Il est encore difficile de déterminer laquelle de ces technologies permettra de construire l'ordinateur quantique du futur.

- Polarisation des photons. La lumière polarisée vibre simultanément dans deux directions orthogonales. La mesure peut conduire à la polarisation horizontale \rightarrow (qui est $|0\rangle$) ou verticale \uparrow (qui est $|1\rangle$).
- Spin d'électrons piégés. Les électrons possèdent un mouvement cinétique interne. On dit qu'un électron « tourne sur lui-même ». Le moment cinétique peut prendre une infinité de valeurs. Mais lorsque l'on mesure ce moment cinétique on n'obtient que deux valeurs possibles $+\frac{\hbar}{2}$ ou $-\frac{\hbar}{2}$. On note $|0\rangle$ et $|1\rangle$ ces deux mesures possibles. Par contre avant la mesure, le moment cinétique avait la forme d'un qubit $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ avec $|\alpha|^2 + |\beta|^2 = 1$.

- Ions piégés. Les ions d'atomes sont piégés dans une cavité magnétique et dans le vide, ils sont manipulés par des rayons laser.
- Atomes froids. Des atomes sont refroidis par des lasers, leur état quantique correspond à leur niveau d'énergie, ils sont également manipulés par des lasers.
- Les supra-conducteurs sur silicium. L'objet quantique est dans ce cas un courant obtenu dans une jonction Josephson constituée par des oxydes d'aluminium sur des puces de silicium. L'état de ces qubits est manipulé par des impulsions à micro-onde qui constituent les portes quantiques.

4.2. Mesure

La mesure joue un rôle important en physique quantique : la mesure est un acte irréversible qui change la fonction d'onde. Avant la mesure, la fonction d'onde d'une particule peut conduire à plusieurs mesures possibles (si on répète l'expérience de la préparation de la particule, puis de la mesure, on obtient des résultats qui peuvent être différents à chaque mesure). Par contre après une mesure, la fonction d'onde est changée définitivement, on parle de la « réduction du paquet d'onde », et ne conduit plus qu'à un seul résultat : si on effectue une seconde mesure immédiatement après, on obtient le même résultat.

La théorie de la décohérence quantique tente d'expliquer le passage du monde quantique à la physique classique. Un état quantique ne reste « cohérent » que s'il n'est pas perturbé. Une particule vit dans un environnement complexe et perd sa cohérence quantique plus ou moins rapidement. En particulier une mesure physique vient déranger la cohérence, et la particule change d'état quantique. Par exemple une molécule (de taille environ 10^{-7} m) dans un vide de laboratoire conserve sa cohérence quantique seulement environ 10^{-17} seconde.

4.3. Intrication quantique

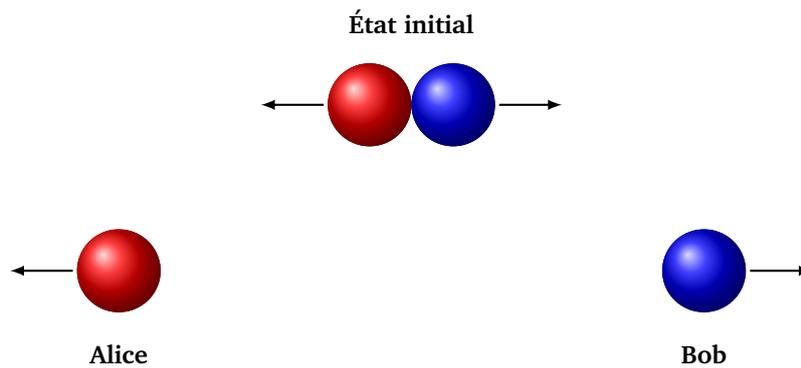
Analogie élémentaire. Commençons par des analogies simplistes. On dispose de deux cartes : une rouge et une bleue. On place ces deux cartes dans deux enveloppes. On donne, au hasard, une enveloppe à Alice et une autre à Bob. Ensuite Alice et Bob se séparent. Pour Alice la probabilité d'avoir la carte rouge est de $1/2$, la carte bleue également $1/2$. De même pour Bob. Si Alice ouvre son enveloppe, et par exemple découvre la carte rouge. Alors bien sûr, elle a la carte rouge avec probabilité 1, mais surtout Bob a maintenant la carte bleue avec probabilité 1, même s'il n'a pas ouvert son enveloppe. Les deux cartes sont liées : on parle d'intrication. La mesure de l'un fait disparaître les probabilités pour les deux cartes.



Un des principes de la théorie de la relativité est qu'aucune particule, aucun signal, aucune information ne peut voyager plus vite que la lumière. Ce principe est-il violé ici ? Dès qu'Alice découvre que son enveloppe contient la carte rouge, elle a automatiquement la certitude que Bob a en sa possession la carte bleue. Ce n'est pas un paradoxe car en fait Bob reste ignorant de la couleur de sa carte. Il peut toutefois connaître la couleur de sa carte sans ouvrir son enveloppe. Il suffit qu'Alice lui téléphone et lui dise qu'elle a eu la carte rouge, ainsi Bob saura qu'il a la carte bleue. Mais aucun principe n'est violé car la vitesse de transmission téléphonique d'une information ne dépasse pas la vitesse de la lumière.

Autre analogie. On continue avec une expérience un peu plus physique. On prend deux particules identiques que l'on fait cogner l'une contre l'autre au point O . L'une part vers Alice et l'autre vers Bob. Elles ont exactement des mouvements opposés : position et vitesse sont égales au signe près. Si Alice mesure la

position ou la vitesse de sa particule, alors cela détermine la position ou la vitesse de la particule Bob. Ainsi Bob peut connaître la position ou la vitesse de sa particule sans faire lui-même une mesure.



Expérience réelle. Dans la pratique on sait préparer deux photons qui sont intriqués par polarisation. Ensuite on sait envoyer les photons à des centaines de kilomètres l'un de l'autre en les maintenant intriqués, ce qui permet de réaliser le codage super-dense et la téléportation quantique.

Crédits photos. [Wikipedia, Double-slit experiment](#) :

- pour les photos de l'expérience en lumière laser à une et deux fentes, Jordgette, CC BY-SA 3.0,
- pour les photos de l'interférence quantique, Dr. Tonomura, CC BY-SA 3.0.

Téléportation quantique

Vidéo ■ partie 7.1. Protocole de la téléportation

Vidéo ■ partie 7.2. Calculs

Vidéo ■ partie 7.3. Mesure partielle

Vidéo ■ partie 7.4. Les états de Bell

Vidéo ■ partie 7.5. Mesure partielle dans la base de Bell

La téléportation quantique permet de transmettre un qubit d'un point A à un point B.

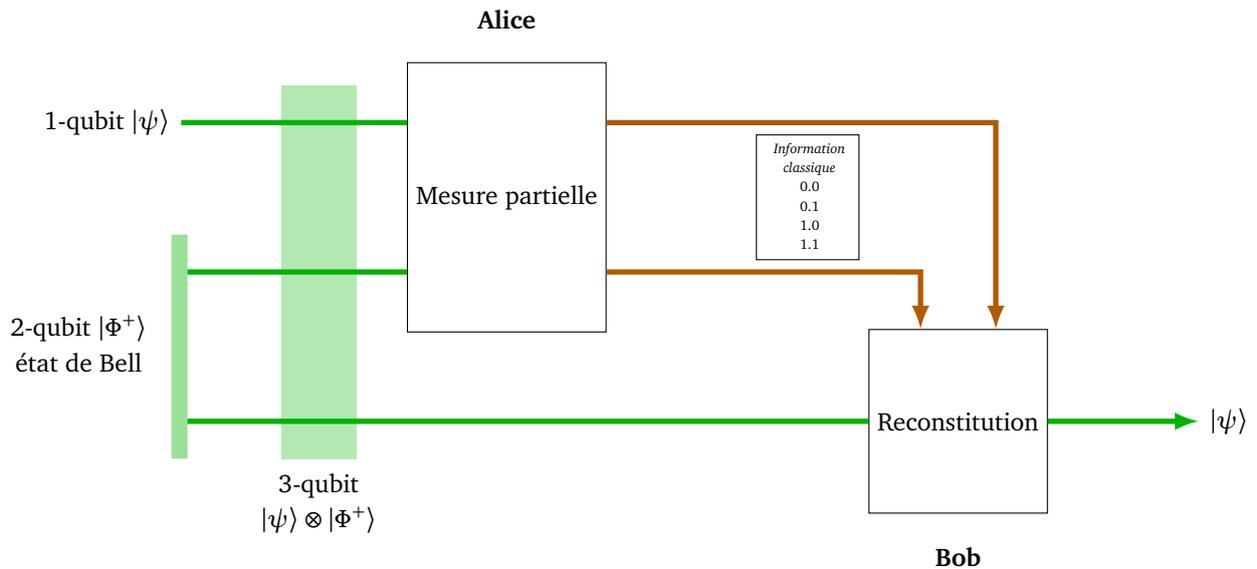
1. Téléportation

1.1. Principe

Alice possède un qubit $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ et souhaite le transmettre à Bob. C'est possible grâce à la téléportation quantique ! C'est un protocole simple ayant des similarités avec le codage super-dense (voir le chapitre « Découverte de l'informatique quantique »).

Plus en détails :

- Alice possède un 1-qubit $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$.
- On a préparé un 2-qubit à l'état de Bell $|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|0.0\rangle + |1.1\rangle)$.
- On réalise le 3-qubit $|\psi\rangle \otimes |\Phi^+\rangle$.
- Alice réalise une mesure partielle (sur les deux premiers qubits) et obtient une information classique (une paire de bits parmi 0.0, 0.1, 1.0, 1.1).
- Bob reconstitue le qubit $|\psi\rangle$ à partir de cette information que lui transmet Alice et du troisième qubit du 3-qubit.

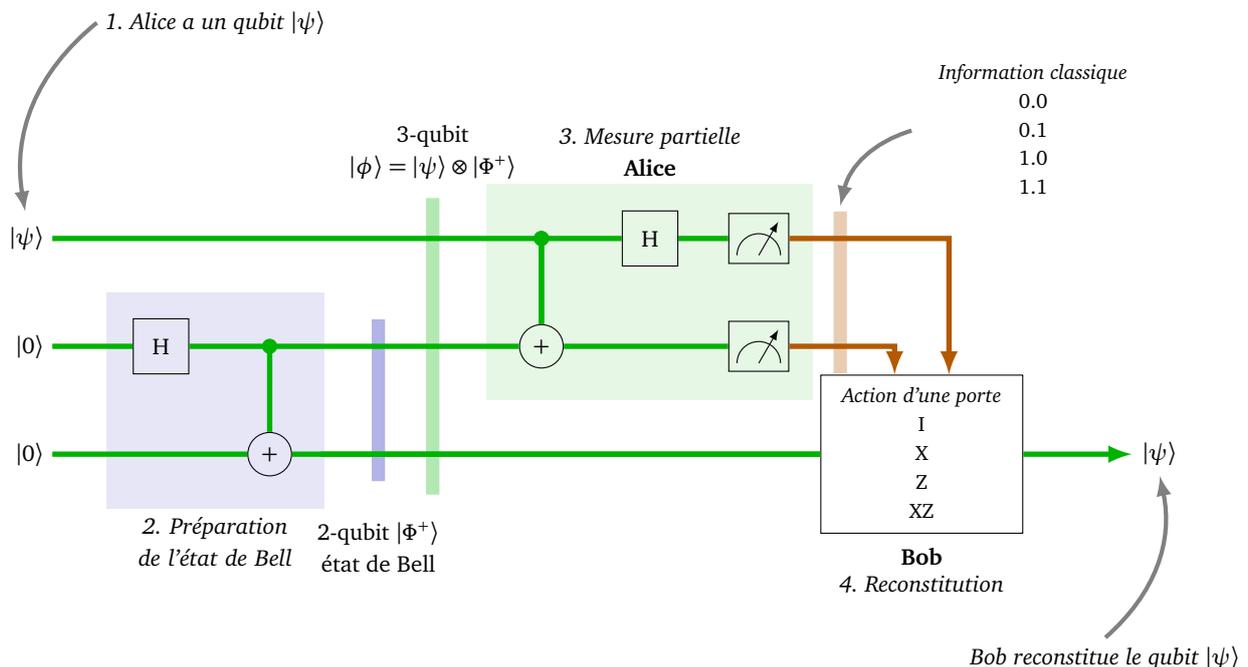


Remarques :

- Le qubit est bien « téléporté », c'est-à-dire qu'il est reconstitué par Bob. Ce n'est physiquement pas la particule originale mais une copie ; seuls deux bits classiques sont envoyés d'Alice à Bob.
- Contrairement au codage super-dense dans lequel Bob obtient à la fin un message d'Alice limité à quatre possibilités, avec la téléportation quantique Bob obtient d'Alice un qubit $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$ parmi une infinité de possibilités car α et β sont des nombres complexes (seulement limités par la contrainte $|\alpha|^2 + |\beta|^2 = 1$).
- La téléportation quantique n'est pas un « copier-coller ». En effet, Alice effectue une mesure sur $|\psi\rangle$ et n'a donc plus ce qubit en sa possession en fin de protocole.
- La téléportation quantique n'est pas immédiate (et ne dépasse pas la vitesse de la lumière) car il faut qu'Alice transmette à Bob une information classique (deux bits classiques).
- Le point-clé est l'intrication des deux qubits de l'état de Bell qui restent liés malgré la distance.

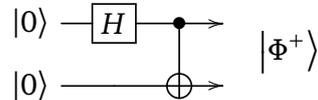
1.2. Protocole de la téléportation

Détaillons le protocole de téléportation.

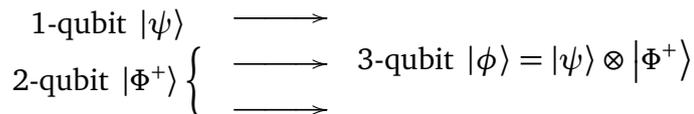


Voici les étapes :

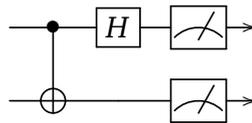
1. Alice possède un qubit $|\psi\rangle$. Alice souhaite transmettre à Bob sans se déplacer le 1-qubit $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$.
2. Préparation de l'état de Bell. L'état de Bell $|\Phi^+\rangle$ peut être préparé par une tierce personne. Nous avons déjà vu comment préparer cet état (voir le chapitre « Découverte de l'informatique quantique ») et nous y reviendrons. À partir de l'état $|0.0\rangle$ on applique le circuit suivant, composé d'une porte de Hadamard, suivie d'une porte *CNOT* :



Le 1-qubit $|\psi\rangle$ réuni avec le 2-qubit $|\Phi^+\rangle$ réalise le 3-qubit $|\phi\rangle = |\psi\rangle \otimes |\Phi^+\rangle$.



3. *Mesure partielle d'Alice*. Alice réalise une mesure partielle (dite mesure partielle dans la base de Bell, voir plus loin). Pour cela elle réalise un circuit qui agit uniquement sur les deux premiers qubits de $|\phi\rangle$: une porte *CNOT*, suivie d'une porte de Hadamard, suivie de deux mesures (sur les qubits 1 et 2 mais pas sur le numéro 3).



Elle obtient deux bits classiques, 0,0, 0,1, 1,0 ou 1,1 selon les mesures. Elle transmet ces deux bits à Bob.

4. *Reconstitution du qubit $|\psi\rangle$ par Bob*. Bob a en main, d'une part le troisième qubit de $|\phi\rangle$, que l'on note $|\phi_3\rangle$, provenant du circuit et la connaissance des deux bits transmis par Alice.
 - Cas 0,0, il ne fait rien (autrement dit, il applique l'identité I), il conserve $|\phi_3\rangle$ qui est en fait $|\psi\rangle$.
 - Cas 0,1, il applique une porte X à $|\phi_3\rangle$ et obtient $|\psi\rangle$.
 - Cas 1,0, il applique une porte Z à $|\phi_3\rangle$ et obtient $|\psi\rangle$.
 - Cas 1,1, il applique une porte X suivie d'une porte Z à $|\phi_3\rangle$ et obtient $|\psi\rangle$.

Dans tous les cas Bob reconstitue le qubit $|\psi\rangle$!

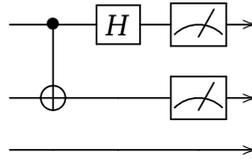
2. Calculs

Nous allons effectuer les calculs qui prouvent que la téléportation quantique fonctionne. Cependant pour comprendre cela en profondeur, il faudra attendre les sections suivantes de ce chapitre.

- Le qubit en possession d'Alice à téléporter est un 1-qubit $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ avec $\alpha, \beta \in \mathbb{C}$ et $|\alpha|^2 + |\beta|^2 = 1$.
- L'état de Bell est $|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|0.0\rangle + |1.1\rangle)$. C'est un 2-qubit (voir plus loin pour plus de détails).
- L'état total composé de $|\psi\rangle$ et $|\Phi^+\rangle$ est le 3-qubit $|\phi\rangle = |\psi\rangle \otimes |\Phi^+\rangle$ que l'on va calculer :

$$\begin{aligned}
 |\phi\rangle &= |\psi\rangle \otimes |\Phi^+\rangle \\
 &= (\alpha|0\rangle + \beta|1\rangle) \left(\frac{1}{\sqrt{2}}|0.0\rangle + \frac{1}{\sqrt{2}}|1.1\rangle \right) \\
 &= \frac{1}{\sqrt{2}} (\alpha|0.0.0\rangle + \alpha|0.1.1\rangle + \beta|1.0.0\rangle + \beta|1.1.1\rangle)
 \end{aligned}$$

- La suite du circuit est :



Elle ne concerne en fait que les deux premiers qubits de $|\phi\rangle$ et est effectuée par Alice.

- La porte *CNOT* change l'état du second qubit en fonction de l'état du premier (le troisième reste inchangé, par exemple $|1.0.0\rangle$ devient $|1.1.0\rangle$). Ainsi

$$|\phi'\rangle = \text{CNOT} |\phi\rangle = \frac{1}{\sqrt{2}} (\alpha |0.0.0\rangle + \alpha |0.1.1\rangle + \beta |1.1.0\rangle + \beta |1.0.1\rangle)$$

- Ensuite la porte de Hadamard ne change que l'état du premier qubit. On rappelle que $H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ et $H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. Ainsi

$$\begin{aligned} |\phi''\rangle &= H |\phi'\rangle \\ &= \frac{1}{\sqrt{2}} \left(\frac{\alpha}{\sqrt{2}} |(0+1).0.0\rangle + \frac{\alpha}{\sqrt{2}} |(0+1).1.1\rangle + \frac{\beta}{\sqrt{2}} |(0-1).1.0\rangle + \frac{\beta}{\sqrt{2}} |(0-1).0.1\rangle \right) \\ &= \frac{\alpha}{2} |0.0.0\rangle + \frac{\alpha}{2} |1.0.0\rangle + \frac{\alpha}{2} |0.1.1\rangle + \frac{\alpha}{2} |1.1.1\rangle \\ &\quad + \frac{\beta}{2} |0.1.0\rangle - \frac{\beta}{2} |1.1.0\rangle + \frac{\beta}{2} |0.0.1\rangle - \frac{\beta}{2} |1.0.1\rangle \end{aligned}$$

- On regroupe les termes qui ont les deux premiers qubits identiques :

$$\begin{aligned} |\phi''\rangle &= \frac{\alpha}{2} |0.0.0\rangle + \frac{\beta}{2} |0.0.1\rangle \\ &\quad + \frac{\beta}{2} |0.1.0\rangle + \frac{\alpha}{2} |0.1.1\rangle \\ &\quad + \frac{\alpha}{2} |1.0.0\rangle - \frac{\beta}{2} |1.0.1\rangle \\ &\quad - \frac{\beta}{2} |1.1.0\rangle + \frac{\alpha}{2} |1.1.1\rangle \end{aligned}$$

- On factorise selon les deux premiers qubits :

$$\begin{aligned} |\phi''\rangle &= \frac{1}{2} |0.0\rangle (\alpha |0\rangle + \beta |1\rangle) \\ &\quad + \frac{1}{2} |0.1\rangle (\beta |0\rangle + \alpha |1\rangle) \\ &\quad + \frac{1}{2} |1.0\rangle (\alpha |0\rangle - \beta |1\rangle) \\ &\quad + \frac{1}{2} |1.1\rangle (-\beta |0\rangle + \alpha |1\rangle) \end{aligned}$$

- La mesure partielle sur les deux premiers qubits conduit (de façon équiprobable) à 0.0 ou 0.1 ou 1.0 ou 1.1. Mais notons que dans tous les cas le troisième qubit est presque $|\psi\rangle$:

- Si Alice mesure 0.0 alors le troisième qubit reçu par Bob est déjà $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$. Il le conserve tel quel (transformation identité I).
- Si Alice mesure 0.1 alors le troisième qubit reçu par Bob est $\beta |0\rangle + \alpha |1\rangle$ et comme il applique ensuite la transformation X (qui échange $|0\rangle$ et $|1\rangle$) cela lui fournit $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$.
- Si Alice mesure 1.0 alors le troisième qubit reçu par Bob est $\alpha |0\rangle - \beta |1\rangle$ et il applique alors la transformation Z (qui change $|1\rangle$ en $-|1\rangle$ et laisse invariant $|0\rangle$) cela lui fournit $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$.
- Si Alice mesure 1.1 alors le troisième qubit reçu par Bob est $-\beta |0\rangle + \alpha |1\rangle$ et il applique la transformation X suivie de Z cela lui fournit encore une fois $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$.

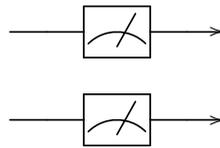
3. Mesure partielle

3.1. Mesure classique d'un 2-qubit

Soit le 2-qubit

$$|\psi\rangle = \alpha |0.0\rangle + \beta |0.1\rangle + \gamma |1.0\rangle + \delta |1.1\rangle$$

où $\alpha, \beta, \gamma, \delta$ sont des nombres complexes tels que $|\alpha|^2 + |\beta|^2 + |\gamma|^2 + |\delta|^2 = 1$. Le circuit suivant effectue la mesure de $|\psi\rangle$ sur chacun des deux 1-qubits.



On sait que la mesure donne :

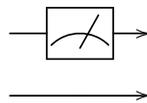
- 0.0 avec la probabilité $|\alpha|^2$,
- 0.1 avec la probabilité $|\beta|^2$,
- 1.0 avec la probabilité $|\gamma|^2$,
- 1.1 avec la probabilité $|\delta|^2$.

3.2. Mesure partielle d'un 2-qubit

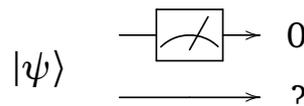
Mesure partielle. On reprend le même 2-qubit

$$|\psi\rangle = \alpha |0.0\rangle + \beta |0.1\rangle + \gamma |1.0\rangle + \delta |1.1\rangle.$$

Le circuit suivant effectue la *mesure partielle* de $|\psi\rangle$ sur son premier 1-qubit seulement (on conserve le second 1-qubit tel quel).



Question. Si on mesure 0 sur le premier qubit, que peut-il arriver sur le second qubit ?



Exemple. Commençons par étudier un exemple :

$$|\psi\rangle = \frac{\sqrt{2}}{2} |0.0\rangle + \frac{1}{2} |0.1\rangle + \frac{1}{2} |1.1\rangle$$

Si pour le premier qubit on mesure $b_1 = 1$, alors le second qubit est nécessairement $q_2 = |1\rangle$ (qui se mesurerait en $b_2 = 1$). En effet dans $|\psi\rangle$ on a le terme $|1.1\rangle$ (donc $q_1 = |1\rangle$ et $q_2 = |1\rangle$) mais pas de terme $|1.0\rangle$ (qui aurait pu donner $q_1 = |1\rangle$ et $q_2 = |0\rangle$).

Si par contre pour le premier qubit on mesure $b_1 = 0$, alors comment faire ? On factorise les termes de $|\psi\rangle$ selon le premier qubit :

$$|\psi\rangle = \frac{1}{2} |0\rangle \cdot (\sqrt{2}|0\rangle + |1\rangle) + \frac{1}{2} |1\rangle \cdot |1\rangle$$

On obtient alors que :

- si $q_1 = |0\rangle$, alors q_2 est donné par $\sqrt{2}|0\rangle + |1\rangle$ qu'il faut normaliser en tant que 1-qubit, c'est-à-dire $q_2 = \frac{\sqrt{2}}{\sqrt{3}} |0\rangle + \frac{1}{\sqrt{3}} |1\rangle$. Une mesure de ce second qubit q_2 , sachant que le premier qubit est mesuré à 0, donnerait 0 avec probabilité $2/3$ et 1 avec probabilité $1/3$.
- si $q_1 = |1\rangle$, alors $q_2 = |1\rangle$, on retrouve bien que si on mesure 1 pour le premier qubit, on mesure nécessairement 1 pour le second.

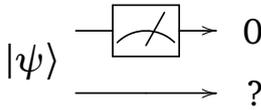
Cas général. Revenons au cas d'un 2-qubit général :

$$|\psi\rangle = \alpha |0.0\rangle + \beta |0.1\rangle + \gamma |1.0\rangle + \delta |1.1\rangle.$$

On factorise par le premier qubit :

$$|\psi\rangle = |0\rangle \cdot (\alpha |0\rangle + \beta |1\rangle) + |1\rangle \cdot (\gamma |0\rangle + \delta |1\rangle).$$

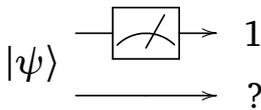
Si on mesure 0 sur le premier qubit, alors le second qubit est le normalisé de $\alpha |0\rangle + \beta |1\rangle$, c'est donc



$$q_2 = \frac{\alpha}{\sqrt{|\alpha|^2 + |\beta|^2}} |0\rangle + \frac{\beta}{\sqrt{|\alpha|^2 + |\beta|^2}} |1\rangle$$

Si on mesurait ce second qubit, on obtiendrait donc 0 avec probabilité $\frac{|\alpha|^2}{|\alpha|^2 + |\beta|^2}$ et 1 avec la probabilité $\frac{|\beta|^2}{|\alpha|^2 + |\beta|^2}$.

Si on mesure 1 sur le premier qubit alors le second qubit est le normalisé de $\gamma |0\rangle + \delta |1\rangle$, c'est donc

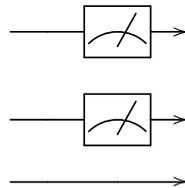


$$q_2 = \frac{\gamma}{\sqrt{|\gamma|^2 + |\delta|^2}} |0\rangle + \frac{\delta}{\sqrt{|\gamma|^2 + |\delta|^2}} |1\rangle$$

Si on mesurait ce second qubit, on obtiendrait donc 0 avec probabilité $\frac{|\gamma|^2}{|\gamma|^2 + |\delta|^2}$ et 1 avec la probabilité $\frac{|\delta|^2}{|\gamma|^2 + |\delta|^2}$.

3.3. Mesure partielle d'un 3-qubit

Le principe est le même si on effectue la mesure partielle des deux premiers qubits d'un 3-qubit.



Considérons par exemple

$$|\psi\rangle = \frac{1}{5} (2 |0.0.0\rangle - |0.0.1\rangle + 3 |0.1.0\rangle + |0.1.1\rangle - 2 |1.0.0\rangle + 2 |1.0.1\rangle + \sqrt{2} |1.1.1\rangle)$$

Factorisons ce 3-qubit selon ses deux premiers qubits :

$$5|\psi\rangle = |0.0\rangle (2|0\rangle - |1\rangle) + |0.1\rangle (3|0\rangle + |1\rangle) + 2|1.0\rangle (-|0\rangle + |1\rangle) + \sqrt{2}|1.1\rangle \cdot |1\rangle$$

- Si la mesure partielle est 0.0 alors le troisième qubit est le normalisé de $2|0\rangle - |1\rangle$, donc $q_3 = \frac{1}{\sqrt{5}}(2|0\rangle - |1\rangle)$. Ainsi la mesure du troisième qubit, sachant que les deux premiers ont été mesurés en 0.0, donnerait 0 avec probabilité 4/5 et 1 avec probabilité 1/5.
- Si la mesure partielle est 0.1 alors le troisième qubit normalisé est $q_3 = \frac{1}{\sqrt{10}}(3|0\rangle + |1\rangle)$.
- Si la mesure partielle est 1.0 alors le troisième qubit normalisé est $q_3 = \frac{1}{\sqrt{2}}(-|0\rangle + |1\rangle)$.
- Si la mesure partielle est 1.1 alors le troisième qubit est $q_3 = |1\rangle$. Dans ce cas, la mesure du troisième qubit donnerait toujours 1.

4. Les états de Bell

4.1. Les quatre états de Bell

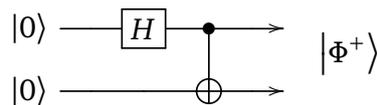
Les quatre états de Bell sont les 2-qubits suivants :

$$\begin{aligned} |\Phi^+\rangle &= \frac{1}{\sqrt{2}} (|0.0\rangle + |1.1\rangle) & |\Psi^+\rangle &= \frac{1}{\sqrt{2}} (|0.1\rangle + |1.0\rangle) \\ |\Phi^-\rangle &= \frac{1}{\sqrt{2}} (|0.0\rangle - |1.1\rangle) & |\Psi^-\rangle &= \frac{1}{\sqrt{2}} (|0.1\rangle - |1.0\rangle) \end{aligned}$$

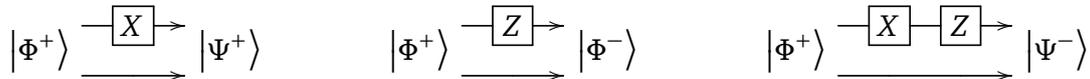
On rappelle que $|\Phi^+\rangle$ est un état intriqué (ce n'est pas le produit de deux 1-qubits). Il en est de même pour les trois autres états de Bell.

4.2. Construction

À partir du qubit $|0.0\rangle$ on construit l'état de Bell $|\Phi^+\rangle$ à l'aide d'une porte de Hadamard H suivie d'une porte $CNOT$.



On construit alors à partir de $|\Phi^+\rangle$, les autres états de Bell.



4.3. Base de Bell

On connaît déjà la **base canonique** des 2-qubits, formée des quatre états :

$$|0.0\rangle \quad |0.1\rangle \quad |1.0\rangle \quad |1.1\rangle$$

Être une **base** signifie que n'importe quel 2-qubit $|\psi\rangle$ de norme 1 s'écrit de façon unique :

$$|\psi\rangle = \alpha |0.0\rangle + \beta |0.1\rangle + \gamma |1.0\rangle + \delta |1.1\rangle$$

où $\alpha, \beta, \gamma, \delta$ sont des nombres complexes avec $|\alpha|^2 + |\beta|^2 + |\gamma|^2 + |\delta|^2 = 1$.

Proposition 1.

Les quatre états de Bell $|\Phi^+\rangle, |\Psi^+\rangle, |\Phi^-\rangle, |\Psi^-\rangle$ forment aussi une base des 2-qubits, appelée **base de Bell**. Cela signifie que n'importe quel 2-qubit $|\psi\rangle$ de norme 1 s'écrit de façon unique :

$$|\psi\rangle = \alpha' |\Phi^+\rangle + \beta' |\Psi^+\rangle + \gamma' |\Phi^-\rangle + \delta' |\Psi^-\rangle$$

avec $|\alpha'|^2 + |\beta'|^2 + |\gamma'|^2 + |\delta'|^2 = 1$.

Comment passer d'une base à une autre? Si $|\psi\rangle = \alpha' |\Phi^+\rangle + \beta' |\Psi^+\rangle + \gamma' |\Phi^-\rangle + \delta' |\Psi^-\rangle$, alors on substitue

$|\Phi^+\rangle, |\Psi^+\rangle, \dots$ à l'aide de l'égalité $|\Phi^+\rangle = \frac{1}{\sqrt{2}} (|0.0\rangle + |1.1\rangle), \dots$ Ce qui donne

$$\begin{aligned} |\psi\rangle &= \alpha' |\Phi^+\rangle + \beta' |\Psi^+\rangle + \gamma' |\Phi^-\rangle + \delta' |\Psi^-\rangle \\ &= \frac{1}{\sqrt{2}} \alpha' (|0.0\rangle + |1.1\rangle) + \frac{1}{\sqrt{2}} \beta' (|0.1\rangle + |1.0\rangle) + \frac{1}{\sqrt{2}} \gamma' (|0.0\rangle - |1.1\rangle) + \frac{1}{\sqrt{2}} \delta' (|0.1\rangle - |1.0\rangle) \\ &= \frac{1}{\sqrt{2}} (\alpha' + \gamma') |0.0\rangle + \frac{1}{\sqrt{2}} (\beta' + \delta') |0.1\rangle + \frac{1}{\sqrt{2}} (\beta' - \delta') |1.0\rangle + \frac{1}{\sqrt{2}} (\alpha' - \gamma') |1.1\rangle \end{aligned}$$

Pour le passage dans l'autre sens, on utilise les identités suivantes :

$$\begin{aligned} |0.0\rangle &= \frac{1}{\sqrt{2}} (|\Phi^+\rangle + |\Phi^-\rangle) \\ |0.1\rangle &= \frac{1}{\sqrt{2}} (|\Psi^+\rangle + |\Psi^-\rangle) \\ |1.0\rangle &= \frac{1}{\sqrt{2}} (|\Psi^+\rangle - |\Psi^-\rangle) \\ |1.1\rangle &= \frac{1}{\sqrt{2}} (|\Phi^+\rangle - |\Phi^-\rangle) \end{aligned}$$

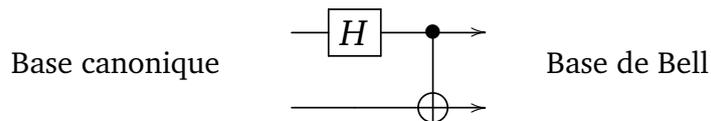
Par exemple si

$$|\psi\rangle = \frac{1}{\sqrt{15}} (|0.0\rangle + 2|0.1\rangle - 3|1.0\rangle - |1.1\rangle)$$

alors :

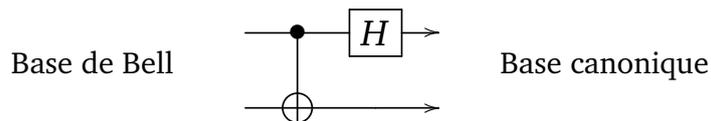
$$\begin{aligned} |\psi\rangle &= \frac{1}{\sqrt{30}} ((|\Phi^+\rangle + |\Phi^-\rangle) + 2(|\Psi^+\rangle + |\Psi^-\rangle) - 3(|\Psi^+\rangle - |\Psi^-\rangle) - (|\Phi^+\rangle - |\Phi^-\rangle)) \\ &= \frac{1}{\sqrt{30}} (0|\Phi^+\rangle - 1|\Psi^+\rangle + 2|\Phi^-\rangle + 5|\Psi^-\rangle) \\ &= \frac{1}{\sqrt{30}} (-|\Psi^+\rangle + 2|\Phi^-\rangle + 5|\Psi^-\rangle) \end{aligned}$$

Une autre vision du passage d'une base à une autre est celle des circuits :



C'est-à-dire, le circuit ci-dessus envoie $|0.0\rangle$ sur $|\Phi^+\rangle$, $|0.1\rangle$ s'envoie sur $|\Psi^+\rangle$, $|1.0\rangle$ s'envoie sur $|\Phi^-\rangle$, $|1.1\rangle$ s'envoie sur $|\Psi^-\rangle$.

Pour le circuit ci-dessous, c'est exactement l'inverse : $|\Phi^+\rangle$ s'envoie sur $|0.0\rangle$, etc.



5. Mesure partielle dans la base de Bell

5.1. Mesure partielle dans une autre base

La téléportation quantique est basée sur une mesure partielle par Alice, qui est en fait une mesure partielle dans la base Bell.

Une mesure partielle d'un 3-qubit $|\phi\rangle$ dans la base canonique, correspond à la factorisation :

$$|\phi\rangle = |0.0\rangle \cdot |\psi_1\rangle + |0.1\rangle \cdot |\psi_2\rangle + |1.0\rangle \cdot |\psi_3\rangle + |1.1\rangle \cdot |\psi_4\rangle.$$

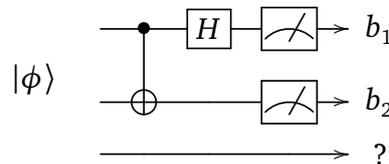
Ainsi, si la mesure partielle sur les deux premiers qubits donne 0.0, alors le troisième qubit est $|\psi_1\rangle$, si la mesure donne 0.1, alors c'est $|\psi_2\rangle$,...

On peut faire un travail similaire dans la base de Bell, avec la factorisation :

$$|\phi\rangle = |\Phi^+\rangle \cdot |\psi'_1\rangle + |\Psi^+\rangle \cdot |\psi'_2\rangle + |\Phi^-\rangle \cdot |\psi'_3\rangle + |\Psi^-\rangle \cdot |\psi'_4\rangle.$$

Si la mesure partielle dans la base de Bell correspond à $|\Phi^+\rangle$ alors le troisième qubit est $|\psi'_1\rangle$, etc.

Comment se fait la mesure partielle dans la base de Bell ? À l'aide du circuit utilisé par Alice !



- Si la mesure donne 0.0 alors les deux premiers qubits de $|\phi\rangle$ forment le 2-qubit $|\Phi^+\rangle$ et donc le troisième qubit de $|\phi\rangle$ est $|\psi'_1\rangle$.
- Si la mesure donne 0.1 alors les deux premiers qubits de $|\phi\rangle$ forment $|\Psi^+\rangle$ et donc le troisième qubit est $|\psi'_2\rangle$.
- etc.

5.2. Calculs de la téléportation quantique

On recommence le calcul de la section 2 qui explique la téléportation quantique, mais cette fois en mettant en évidence qu'elle est basée sur la mesure partielle de l'état $|\psi\rangle \otimes |\Phi^+\rangle$ dans la base de Bell.

On rappelle :

- le qubit à téléporter est $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$,
- l'état de Bell est $|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|0.0\rangle + |1.1\rangle)$,
- le 3-qubit composé de $|\psi\rangle$ et $|\Phi^+\rangle$ est $|\phi\rangle = |\psi\rangle \otimes |\Phi^+\rangle$.

$$\begin{aligned} |\phi\rangle &= |\psi\rangle \otimes |\Phi^+\rangle \\ &= \frac{1}{\sqrt{2}} (\alpha|0.0.0\rangle + \alpha|0.1.1\rangle + \beta|1.0.0\rangle + \beta|1.1.1\rangle) \\ &= \frac{1}{\sqrt{2}} (\alpha|0.0\rangle \cdot |0\rangle + \alpha|0.1\rangle \cdot |1\rangle + \beta|1.0\rangle \cdot |0\rangle + \beta|1.1\rangle \cdot |1\rangle) \\ &= \frac{1}{2}\alpha (|\Phi^+\rangle + |\Phi^-\rangle) \cdot |0\rangle + \frac{1}{2}\alpha (|\Psi^+\rangle + |\Psi^-\rangle) \cdot |1\rangle + \frac{1}{2}\beta (|\Psi^+\rangle - |\Psi^-\rangle) \cdot |0\rangle + \frac{1}{2}\beta (|\Phi^+\rangle - |\Phi^-\rangle) \cdot |1\rangle \\ &= \frac{1}{2} |\Phi^+\rangle (\alpha|0\rangle + \beta|1\rangle) \\ &\quad + \frac{1}{2} |\Psi^+\rangle (\beta|0\rangle + \alpha|1\rangle) \\ &\quad + \frac{1}{2} |\Phi^-\rangle (\alpha|0\rangle - \beta|1\rangle) \\ &\quad + \frac{1}{2} |\Psi^-\rangle (-\beta|0\rangle + \alpha|1\rangle) \end{aligned}$$

Ainsi, si la mesure partielle dans la base de Bell sur les deux premiers qubits identifie $|\Phi^+\rangle$, alors le troisième qubit est déjà $|\psi\rangle$, sinon Bob n'a qu'à appliquer les transformations X et Z pour reconstituer $|\psi\rangle$.

DEUXIÈME PARTIE

| 1 >

| J >

ALGORITHMES QUANTIQUES

Un premier algorithme quantique

Vidéo ■ partie 8. Un premier algorithme quantique

La force de l'informatique quantique est de pouvoir faire des calculs avec des 0 et des 1 en même temps. Au lieu de deux calculs classiques sur le bit 0, puis sur le bit 1, l'ordinateur quantique effectue un seul calcul sur un 1-qubit. Encore plus fort : avec un n -qubit, un seul calcul quantique remplace 2^n calculs classiques.

La réalité est cependant plus compliquée, car tous les algorithmes de l'informatique classique ne vont pas miraculeusement être plus rapides grâce à l'informatique quantique. Nous allons voir dans cette partie des problèmes que l'informatique quantique résout beaucoup mieux que les algorithmes classiques. Le but final est de comprendre l'algorithme quantique de Shor qui permet la factorisation rapide des entiers.

Nous commençons par étudier une version simple de l'algorithme de Deutsch–Jozsa afin de nous familiariser avec les objets, les techniques et les types d'algorithmes que nous découvrirons dans cette seconde partie du livre.

1. Objectifs

1.1. Motivation

L'algorithme de Deutsch–Jozsa n'est pas très utile ! Il permet de décider si une fonction est constante ou équilibrée. Cependant cet algorithme est très intéressant car il prouve que l'informatique quantique permet de faire des calculs plus rapidement qu'avec un ordinateur classique.

L'algorithme complet (avec n variables) sera étudié plus loin dans le chapitre « Algorithme de Deutsch–Jozsa ». Dans ce chapitre d'introduction, on se contente de présenter l'algorithme pour les fonctions les plus simples : celles ayant une, puis deux variables.

1.2. Fonction à étudier

On commence par le cas des fonctions d'une seule variable. L'ensemble de départ et d'arrivée est $\{0, 1\}$. Considérons une telle fonction :

$$f : \{0, 1\} \longrightarrow \{0, 1\}$$

Il y a en fait 4 fonctions possibles que l'on sépare en deux catégories :

Fonctions constantes

$$f_0 \begin{cases} 0 \mapsto 0 \\ 1 \mapsto 0 \end{cases}$$

$$f_1 \begin{cases} 0 \mapsto 1 \\ 1 \mapsto 1 \end{cases}$$

Fonctions équilibrées

$$f_2 \begin{cases} 0 \mapsto 0 \\ 1 \mapsto 1 \end{cases}$$

$$f_3 \begin{cases} 0 \mapsto 1 \\ 1 \mapsto 0 \end{cases}$$

Problème. On nous donne une fonction $f : \{0, 1\} \rightarrow \{0, 1\}$, comment déterminer si elle est constante ou équilibrée ?

1.3. Solution classique

La solution classique à ce problème est simple :

- calculer $f(0)$;
- calculer $f(1)$;
- conclure : si $f(0) = f(1)$ la fonction est constante, sinon elle est équilibrée.

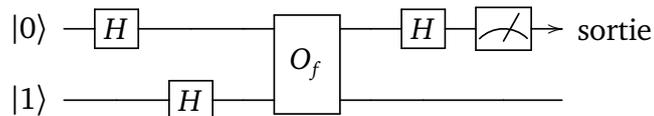
Cet algorithme est très simple, mais il demande deux évaluations de la fonction f (le calcul de $f(0)$ puis celui de $f(1)$) et on ne peut pas faire mieux. Si on définit la complexité de cet algorithme par le nombre d'évaluations de f , alors sa complexité vaut 2.

Nous allons voir un algorithme quantique dont la complexité est 1. Cela ne vous paraît peut-être pas formidablement mieux, mais dans le cas d'une fonction de n variables alors la complexité classique est d'ordre 2^n alors que l'algorithme quantique reste de complexité 1. L'amélioration est donc exponentielle !

2. Circuit quantique

2.1. Circuit

L'algorithme quantique est fourni par le circuit quantique ci-dessous qui répond au problème à l'aide d'une seule évaluation de f .



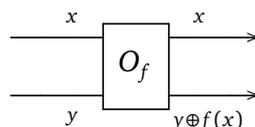
Le circuit est initialisé, puis utilise des portes de Hadamard H mais aussi un sous-circuit O_f , appelé « oracle » que nous détaillerons après.

Algorithme.

- *Entrée.* Une fonction $f : \{0, 1\} \rightarrow \{0, 1\}$.
- *Sortie.* La sortie est donnée par la mesure sur le premier qubit du circuit. Si la mesure vaut 0, la fonction est constante ; si la mesure vaut 1, la fonction est équilibrée.

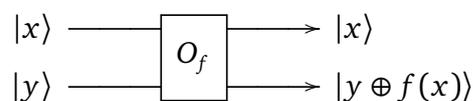
2.2. Oracles

Un *oracle* est un circuit quantique associé à une fonction f . Voici ce que réalise un oracle pour une fonction $f : \{0, 1\} \rightarrow \{0, 1\}$.



où x et y sont des bits classiques 0 ou 1.

L'oracle nous donne l'action de la porte O_f sur les qubits de base $|0\rangle$ et $|1\rangle$.



Détaillons ce qui se passe sur chaque ligne de l'oracle.

Première ligne. En entrée l'oracle reçoit le bit x et en sortie il renvoie cette même valeur x .

Seconde ligne. En entrée l'oracle reçoit le bit y mais la sortie dépend des valeurs de x , y et de la fonction f . Cette sortie est le bit 0 ou 1, donné par la formule :

$$y \oplus f(x)$$

Addition binaire. L'addition « \oplus » est l'addition binaire sur les bits 0 et 1. Elle est équivalente au « ou exclusif » :

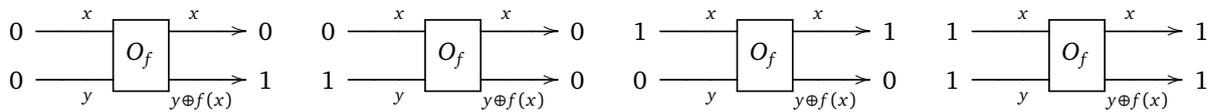
$$0 \oplus 0 = 0 \quad 0 \oplus 1 = 1 \quad 1 \oplus 0 = 1 \quad \boxed{1 \oplus 1 = 0}$$

On rappelle que les valeurs x , y et $f(x)$ valent 0 ou 1. Ainsi la sortie $y \oplus f(x)$ vaut aussi 0 ou 1.

Exemple.

Prenons la fonction f définie par $f(0) = 1$ et $f(1) = 0$.

- Pour $x = 0$, $y = 0$ alors $f(0) = 1$ donc $y \oplus f(x) = 0 \oplus 1 = 1$.
- Pour $x = 0$, $y = 1$ alors $f(0) = 1$ donc $y \oplus f(x) = 1 \oplus 1 = 0$.
- Pour $x = 1$, $y = 0$ alors $f(1) = 0$ donc $y \oplus f(x) = 0 \oplus 0 = 0$.
- Pour $x = 1$, $y = 1$ alors $f(1) = 0$ donc $y \oplus f(x) = 1 \oplus 0 = 1$.



Fonction de deux variables.

Dans notre situation, l'oracle fournit une fonction de deux variables $F : \{0, 1\}^2 \rightarrow \{0, 1\}^2$ définie par :

$$F(x, y) = (x, y \oplus f(x)).$$

Exemple.

Reprenons la fonction f définie par $f(0) = 1$ et $f(1) = 0$. Alors

$$(0, 0) \xrightarrow{F} (0, 1) \quad (0, 1) \xrightarrow{F} (0, 0) \quad (1, 0) \xrightarrow{F} (1, 0) \quad (1, 1) \xrightarrow{F} (1, 1)$$

Action sur les qubits.

L'oracle associé à f définit alors une fonction sur les 2-qubits. Notons $\tilde{F} : \mathbb{C}^4 \rightarrow \mathbb{C}^4$ définie sur la base canonique $(|0.0\rangle, |0.1\rangle, |1.0\rangle, |1.1\rangle)$ par la fonction F , c'est à dire $\tilde{F}(|x.y\rangle) = |F(x, y)\rangle$, puis étendue par linéarité à \mathbb{C}^4 . Si

$$|\psi\rangle = \alpha |0.0\rangle + \beta |0.1\rangle + \gamma |1.0\rangle + \delta |1.1\rangle.$$

Alors :

$$\tilde{F}(|\psi\rangle) = \alpha \tilde{F}(|0.0\rangle) + \beta \tilde{F}(|0.1\rangle) + \gamma \tilde{F}(|1.0\rangle) + \delta \tilde{F}(|1.1\rangle).$$

Exemple.

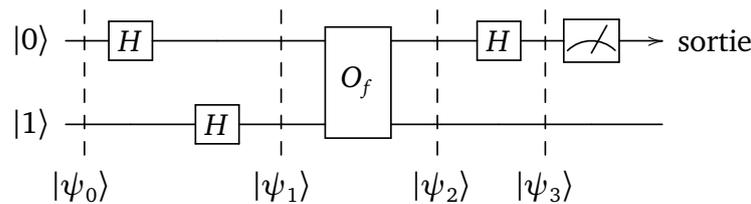
Toujours sur le même exemple, cela donne :

$$|0.0\rangle \xrightarrow{\tilde{F}} |0.1\rangle \quad |0.1\rangle \xrightarrow{\tilde{F}} |0.0\rangle \quad |1.0\rangle \xrightarrow{\tilde{F}} |1.0\rangle \quad |1.1\rangle \xrightarrow{\tilde{F}} |1.1\rangle$$

Et ainsi :

$$\tilde{F}(|\psi\rangle) = \beta |0.0\rangle + \alpha |0.1\rangle + \gamma |1.0\rangle + \delta |1.1\rangle.$$

2.3. Preuve



Nous détaillons les calculs en suivant l'évolution des qubits au fil du circuit.

Qubit initial $|\psi_0\rangle$.

$$|\psi_0\rangle = |0\rangle \otimes |1\rangle = |0.1\rangle$$

Qubit $|\psi_1\rangle$ obtenu après transformation de Hadamard.

On applique une porte de Hadamard sur la première ligne : $H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, et une autre sur la seconde ligne $H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. Ainsi :

$$\begin{aligned} |\psi_1\rangle &= \frac{1}{2}(|0\rangle + |1\rangle)(|0\rangle - |1\rangle) \\ &= \frac{1}{2}(|0.0\rangle - |0.1\rangle + |1.0\rangle - |1.1\rangle). \end{aligned}$$

Pour simplifier l'écriture des calculs dans la suite, on va « oublier » le coefficient $\frac{1}{2}$ et écrire 0.0 au lieu de $|0.0\rangle$, 0.1 au lieu de $|0.1\rangle$,... Ainsi on note :

$$|\psi_1\rangle \equiv 0.0 - 0.1 + 1.0 - 1.1$$

Qubit $|\psi_2\rangle$ obtenu après l'oracle.

$$|\psi_2\rangle \equiv 0.(0 \oplus f(0)) - 0.(1 \oplus f(0)) + 1.(0 \oplus f(1)) - 1.(1 \oplus f(1))$$

En effet, l'oracle envoie x sur x pour la première ligne et y sur $y \oplus f(x)$ pour la seconde. Attention « \oplus » est l'addition binaire et doit être effectuée en priorité. Il ne faut pas la confondre avec l'addition de qubits, notée « $+$ » : $x.(y \oplus f(x))$ n'a rien à voir avec $x.(y + f(x))$.

On regroupe les termes commençant par le même qubit :

$$|\psi_2\rangle \equiv \underbrace{0.(0 \oplus f(0)) - 0.(1 \oplus f(0))}_A + \underbrace{1.(0 \oplus f(1)) - 1.(1 \oplus f(1))}_B.$$

Calculons le terme A en fonction de $f(0)$:

$$A = \begin{cases} 0.0 - 0.1 & \text{si } f(0) = 0 \\ -(0.0 - 0.1) & \text{si } f(0) = 1 \end{cases} \quad \text{donc} \quad A = (-1)^{f(0)}(0.0 - 0.1).$$

On rappelle que $(-1)^k$ est juste une façon d'obtenir $+1$ ou -1 selon la parité de k :

$$(-1)^k = \begin{cases} +1 & \text{si } k = 0 \text{ (ou si } k \text{ est pair)} \\ -1 & \text{si } k = 1 \text{ (ou si } k \text{ est impair)} \end{cases}$$

On calcule de façon similaire B . Ainsi :

$$|\psi_2\rangle \equiv (-1)^{f(0)}(0.0 - 0.1) + (-1)^{f(1)}(1.0 - 1.1).$$

Qubit $|\psi_3\rangle$ obtenu après une porte de Hadamard.

Après l'oracle on applique une porte de Hadamard sur la première ligne. Ainsi :

$$\begin{aligned}
|\psi_3\rangle &\equiv (-1)^{f(0)}((0+1).0 - (0+1).1) \\
&\quad + (-1)^{f(1)}((0-1).0 - (0-1).1) \\
&\equiv (-1)^{f(0)}(0.0 - 0.1 + 1.0 - 1.1) \\
&\quad + (-1)^{f(1)}(0.0 - 0.1 - 1.0 + 1.1) \\
&\equiv ((-1)^{f(0)} + (-1)^{f(1)}) 0.0 \\
&\quad + (-(-1)^{f(0)} - (-1)^{f(1)}) 0.1 \\
&\quad + ((-1)^{f(0)} - (-1)^{f(1)}) 1.0 \\
&\quad + (-(-1)^{f(0)} + (-1)^{f(1)}) 1.1
\end{aligned}$$

Le coefficient que l'on a omis devant tous les qubits est $\frac{1}{2\sqrt{2}}$ et correspond aux trois portes de Hadamard (chacune apportant un facteur $\frac{1}{\sqrt{2}}$) :

$$|\psi_3\rangle = \frac{1}{2\sqrt{2}}((-1)^{f(0)} + (-1)^{f(1)}) |0.0\rangle + \dots$$

Discutons maintenant selon la catégorie de f .

Si f est constante. Alors $f(0) = f(1)$, donc

$$\begin{aligned}
(-1)^{f(0)} + (-1)^{f(1)} &= \begin{cases} +2 \\ \text{ou } -2 \end{cases} \\
\text{et } (-1)^{f(0)} - (-1)^{f(1)} &= 0.
\end{aligned}$$

Ainsi :

$$|\psi_3\rangle = \pm \frac{1}{\sqrt{2}}(|0.0\rangle - |0.1\rangle)$$

donc la mesure sur le premier qubit donne 0 dans tous les cas, car les seuls 2-qubits présents sont $|0.0\rangle$ et $|0.1\rangle$.

Si f est équilibrée. Alors $f(0) \neq f(1)$, donc

$$(-1)^{f(0)} + (-1)^{f(1)} = 0 \quad \text{et} \quad (-1)^{f(0)} - (-1)^{f(1)} = \pm 2$$

alors :

$$|\psi_3\rangle = \pm \frac{1}{\sqrt{2}}(|1.0\rangle - |1.1\rangle).$$

La mesure sur le premier qubit donne donc 1 dans tous les cas (car les 2-qubits présents sont $|1.0\rangle$ et $|1.1\rangle$).

Conclusion. Si f est constante la mesure du premier qubit donne 0, si f est équilibrée cette mesure donne 1. Ainsi le circuit répond bien au problème posé et l'oracle associé à f n'a été appelé qu'une seule fois.

2.4. Réalisation des oracles

C'est à celui qui utilise l'algorithme de fournir l'oracle, sorte de boîte noire, utilisée par l'algorithme. Voyons quel circuit quantique permet de réaliser l'oracle O_f pour chacune des quatre possibilités de la fonction f . Notons au préalable que x s'envoie sur x , donc pour la première ligne quantique il n'y a rien à faire.

Fonction constante égale à 0

$$f_0 \begin{cases} 0 \mapsto 0 \\ 1 \mapsto 0 \end{cases}$$

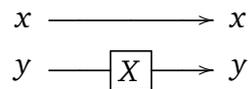
Comme $y \oplus f(x) = y \oplus 0 = y$ alors l'oracle envoie y sur y . Il n'y a rien à faire comme circuit quantique.

$$\begin{array}{c}
x \longrightarrow x \\
y \longrightarrow y
\end{array}$$

Fonction constante égale à 1

$$f_1 \begin{cases} 0 \mapsto 1 \\ 1 \mapsto 1 \end{cases}$$

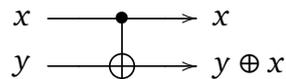
Comme $y \oplus f(x) = y \oplus 1 = NOT(y)$ alors l'oracle envoie y sur $NOT(y)$, que l'on peut réaliser par une porte X .



Fonction équilibrée identité

$$f_2 \begin{cases} 0 \mapsto 0 \\ 1 \mapsto 1 \end{cases}$$

Alors $f_2(x) = x$ et $y \oplus f(x) = y \oplus x$, c'est donc y si $x = 0$ et $NOT(y)$ si $x = 1$. C'est exactement l'action d'une porte $CNOT$:



Fonction équilibrée f_3

$$f_3 \begin{cases} 0 \mapsto 1 \\ 1 \mapsto 0 \end{cases}$$

À vous de chercher en exercice un circuit qui réalise cet oracle en vous aidant des deux circuits précédents.

3. Cas de deux variables

3.1. Problème

On considère maintenant une fonction de deux variables :

$$\begin{aligned} f : \{0, 1\}^2 &\longrightarrow \{0, 1\} \\ (x, y) &\longmapsto f(x, y) \end{aligned}$$

On ne s'intéresse qu'à deux catégories de fonctions.

Fonctions constantes. Il y a en a deux :

- f est constante égale à 0 : $f(x, y) = 0 \forall x, y \in \{0, 1\}$,
- f est constante égale à 1 : $f(x, y) = 1 \forall x, y \in \{0, 1\}$.

Fonctions équilibrées. Pour ces fonctions, il y a autant de valeurs (x, y) avec $f(x, y) = 0$ que de valeurs avec $f(x, y) = 1$. Il y a 6 fonctions possibles. Voici un exemple :

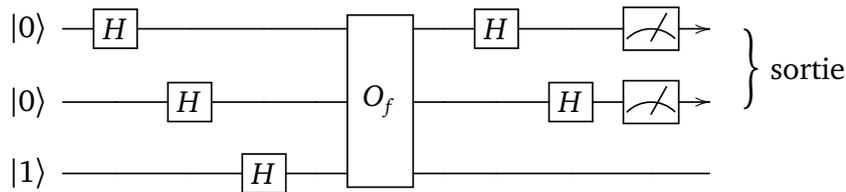
$$(0, 0) \xrightarrow{f} 1 \quad (0, 1) \xrightarrow{f} 0 \quad (1, 0) \xrightarrow{f} 0 \quad (1, 1) \xrightarrow{f} 1$$

Attention ! Il existe des fonctions qui ne sont ni constantes, ni équilibrées. Par exemple, la fonction qui vaut 0 partout, sauf en $(1, 1)$ ($f(1, 1) = 1$).

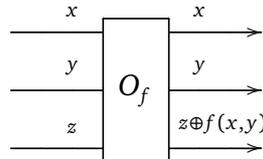
Problème. On nous donne une fonction $f : \{0, 1\}^2 \longrightarrow \{0, 1\}$ qui a la propriété d'être soit constante, soit équilibrée, mais on ne nous dit pas à quelle catégorie elle appartient. Comment déterminer cette catégorie constante ou équilibrée ?

La solution classique est de calculer plusieurs valeurs. Parfois calculer deux valeurs suffit, par exemple si $f(0, 0) \neq f(0, 1)$ alors la fonction n'est pas constante, elle est donc équilibrée. Mais si $f(0, 0) = f(0, 1)$ alors il faut calculer une troisième valeur $f(1, 0)$ pour pouvoir conclure. La complexité de l'algorithme classique est de 3 évaluations (on retient toujours le pire cas).

3.2. Circuit solution



Encore une fois, le circuit fait intervenir des portes de Hadamard et un oracle O_f qui dépend de la fonction f dont le circuit quantique est fourni par celui qui pose le problème. Pour nous, c'est une boîte noire :



où x, y, z sont des bits classiques 0 ou 1. La sortie de la troisième ligne est $z \oplus f(x, y)$.

Noter que la mesure se fait sur les deux premières lignes quantiques seulement. La réponse au problème est donnée par cette mesure :

- si la mesure est 0.0 alors la fonction est constante,
- sinon la fonction est équilibrée.

On rappelle que la fonction f doit par hypothèse être dans l'une des deux catégories ci-dessus.

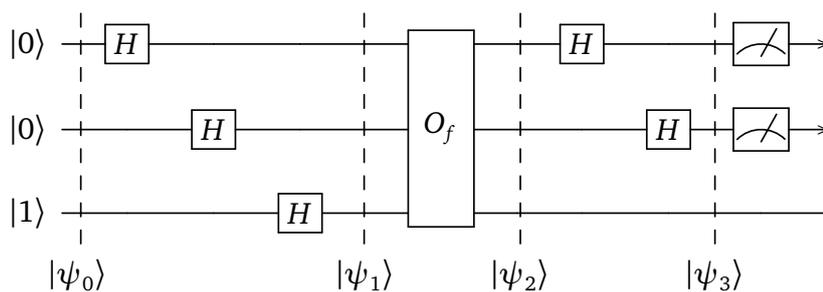
Le circuit quantique n'effectue qu'une seule évaluation de f (plus précisément qu'un seul appel au circuit de l'oracle) et donc la solution proposée est de complexité 1. Cette évaluation correspond à

$$f\left(\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\right),$$

qui est une façon d'évaluer f sur les quatre qubits de base $|0.0\rangle, |0.1\rangle, |1.0\rangle$ et $|1.1\rangle$ simultanément.

3.3. Calcul et preuve

Les calculs et la preuve peuvent être omis lors d'une première lecture, d'une part ils sont similaires à ceux pour une variable (mais un peu plus compliqués) et d'autre part les calculs seront faits dans le cas général d'une fonction de n variables dans le chapitre « Algorithme de Deutsch–Jozsa ».



Qubit initial $|\psi_0\rangle$.

$$|\psi_0\rangle = |0\rangle \otimes |0\rangle \otimes |1\rangle = |0.0.1\rangle$$

Qubit $|\psi_1\rangle$. Pour simplifier l'écriture des calculs on « oublie » le coefficient constant commun à tous les

qubits.

$$\begin{aligned}
 |\psi_1\rangle &\equiv (|0\rangle + |1\rangle)(|0\rangle + |1\rangle)(|0\rangle - |1\rangle) \\
 &\equiv |0.0.0\rangle - |0.0.1\rangle \\
 &\quad + |0.1.0\rangle - |0.1.1\rangle \\
 &\quad + |1.0.0\rangle - |1.0.1\rangle \\
 &\quad + |1.1.0\rangle - |1.1.1\rangle \\
 &\equiv \sum_{x,y \in \{0,1\}} |x.y.0\rangle - |x.y.1\rangle
 \end{aligned}$$

Qubit $|\psi_2\rangle$. On applique l'oracle et on va remarquer que

$$|x.y.(0 \oplus f(x,y))\rangle - |x.y.(1 \oplus f(x,y))\rangle = (-1)^{f(x,y)}(|x.y.0\rangle - |x.y.1\rangle).$$

Ainsi

$$\begin{aligned}
 |\psi_2\rangle &\equiv \sum_{x,y \in \{0,1\}} |x.y.(0 \oplus f(x,y))\rangle - |x.y.(1 \oplus f(x,y))\rangle \\
 &\equiv \sum_{x,y \in \{0,1\}} (-1)^{f(x,y)}(|x.y.0\rangle - |x.y.1\rangle) \\
 &\equiv \sum_{x,y \in \{0,1\}} (-1)^{f(x,y)} |x.y\rangle |0-1\rangle
 \end{aligned}$$

Qubit $|\psi_3\rangle$.

$$\begin{aligned}
 |\psi_3\rangle &\equiv \sum_{x,y \in \{0,1\}} (-1)^{f(x,y)} |H(x).H(y)\rangle |0-1\rangle \\
 &\equiv (-1)^{f(0,0)} |0+1\rangle |0+1\rangle |0-1\rangle \\
 &\quad + (-1)^{f(0,1)} |0+1\rangle |0-1\rangle |0-1\rangle \\
 &\quad + (-1)^{f(1,0)} |0-1\rangle |0+1\rangle |0-1\rangle \\
 &\quad + (-1)^{f(1,1)} |0-1\rangle |0-1\rangle |0-1\rangle
 \end{aligned}$$

Le troisième qubit est toujours $|0-1\rangle$. On ne va pas expliciter tous les termes mais seulement le coefficient devant le qubit $|0.0.(0-1)\rangle$. On en profite pour remettre les coefficients oubliés :

$$|\psi_3\rangle = \underbrace{\frac{1}{4}((-1)^{f(0,0)} + (-1)^{f(0,1)} + (-1)^{f(1,0)} + (-1)^{f(1,1)})}_{\alpha} |0.0\rangle \left| \frac{1}{\sqrt{2}}(0-1) \right\rangle + \dots$$

Conclusion.

Si f est constante alors $\alpha = \pm 1$ (ce qui fait qu'il n'y a pas d'autres qubits) et

$$|\psi_3\rangle = \pm |0.0\rangle \left| \frac{1}{\sqrt{2}}(0-1) \right\rangle.$$

Ainsi toute mesure sur les deux premiers qubits donne 0.0.

Si f est équilibrée alors il y a autant de valeurs en lesquelles f vaut 0 que de valeurs en lesquelles f vaut 1, donc

$$\alpha = \frac{1}{4}((-1)^{f(0,0)} + (-1)^{f(0,1)} + (-1)^{f(1,0)} + (-1)^{f(1,1)}) = 0.$$

Ainsi $|\psi_3\rangle$ n'a pas de qubits commençant par 0.0. Donc aucune mesure sur les deux premiers qubits ne peut donner 0.0.

Nous avons donc bien résolu notre problème : si la mesure des deux premiers qubits donne 0.0 alors f est constante, sinon f est équilibrée.

Portes quantiques

Vidéo ■ partie 9.1. Portes quantiques - Théorie

Vidéo ■ partie 9.2. Portes quantiques - Oracle

Nous approfondissons nos connaissances théoriques des portes quantiques en étudiant ce qu'elles peuvent réaliser (ou pas!).

1. La porte de Toffoli est universelle

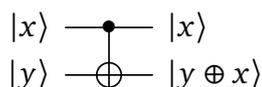
1.1. Quelques portes quantiques

Nous présentons de nouvelles portes et leur lien avec des portes déjà rencontrées.

Porte CNOT. Nous connaissons bien la porte *CNOT*.



On rappelle que la porte *CNOT* est une porte *NOT* conditionnelle, si sur la première ligne on a le qubit $|0\rangle$ alors la seconde ligne est inchangée ; par contre si le qubit de la première ligne est $|1\rangle$ alors la seconde échange le qubit $|0\rangle$ en $|1\rangle$ et inversement. Si x, y ont pour valeurs 0 ou 1, alors l'action sur la seconde ligne est en fait $y \oplus x$ où « \oplus » est l'addition binaire (et ne doit pas être confondue avec l'addition de qubits).

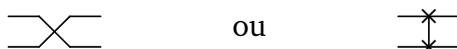


$$|0.0\rangle \xrightarrow{CNOT} |0.0\rangle \quad |0.1\rangle \xrightarrow{CNOT} |0.1\rangle \quad |1.0\rangle \xrightarrow{CNOT} |1.1\rangle \quad |1.1\rangle \xrightarrow{CNOT} |1.0\rangle$$

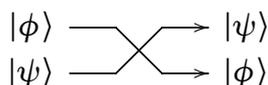
Voici la matrice de la transformation de *CNOT* (dans la base $(|0.0\rangle, |0.1\rangle, |1.0\rangle, |1.1\rangle)$) :

$$M = \left(\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{array} \right)$$

Porte SWAP. La porte *SWAP* échange deux qubits. Voici sa notation :



Comme on l'a dit, cette porte échange deux qubits :

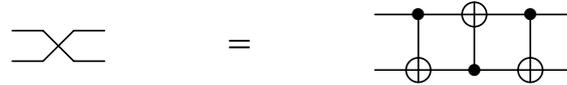


Exercice.

Calculer l'image des 2-qubits de la base canonique ($|0.0\rangle, |0.1\rangle, |1.0\rangle, |1.1\rangle$) et en déduire la matrice 4×4 de la porte *SWAP*.

Exercice.

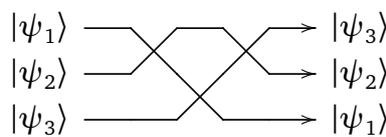
Montrer qu'une porte *SWAP* est équivalente à un circuit réalisé à partir de trois portes *CNOT*.



Indication. Il suffit de le vérifier sur les 2-qubits de la base canonique.

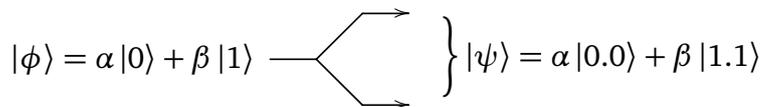
Exercice.

Montrer que le circuit suivant, construit à partir de trois portes *SWAP* correspond à une porte *SWAP*₃ qui renverse l'ordre de 3 qubits, c'est-à-dire $(|\psi_1\rangle, |\psi_2\rangle, |\psi_3\rangle) \mapsto (|\psi_3\rangle, |\psi_2\rangle, |\psi_1\rangle)$.



Il existe un circuit qui, à partir de portes *SWAP*, réalise une porte *SWAP*_n renversant l'ordre de n qubits, c'est-à-dire $(|\psi_1\rangle, |\psi_2\rangle, \dots, |\psi_n\rangle) \mapsto (|\psi_n\rangle, \dots, |\psi_2\rangle, |\psi_1\rangle)$. Construire un tel circuit pour $n = 4$.

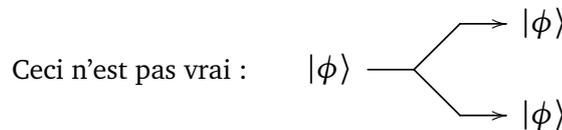
Porte FANOUT. La porte *FANOUT* transforme un 1-qubit en un 2-qubit. Dans un circuit quantique, cela permet d'augmenter le nombre de lignes quantiques.



Piège. La porte *FANOUT* envoie $|0\rangle$ sur $|0.0\rangle$ et $|1\rangle$ sur $|1.1\rangle$.



Cependant, il faut bien comprendre la porte *FANOUT* ne réalise pas un copier-coller du 1-qubit d'entrée.



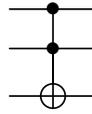
D'ailleurs, une telle porte ne peut pas exister ! Ce sera prouvé par le théorème de non-clonage quantique en fin de chapitre.

Exercice.

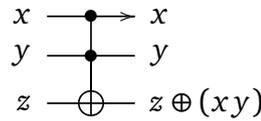
Montrer qu'une porte *FANOUT* peut être réalisée à partir d'une porte *CNOT* initialisée par $|0\rangle$ sur sa seconde ligne.



Porte de Toffoli (CCNOT). La porte de Toffoli est similaire à une porte *CNOT* mais avec trois lignes. Si les deux premiers qubits sont $|1\rangle$, alors on applique une porte *X* (c'est-à-dire *NOT*) au troisième qubit.



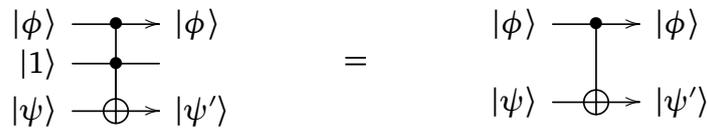
Voici l'action d'une porte de Toffoli lorsque x, y, z sont des bits 0 ou 1 (noter que $xy = 1$ si et seulement si $x = 1$ et $y = 1$ et alors $1 \oplus z = NOT(z)$).



$$M = \begin{pmatrix} 1 & 0 & | & & | & & | & & | \\ 0 & 1 & | & & | & & | & & | \\ \hline & & 1 & 0 & | & & | & & | \\ & & 0 & 1 & | & & | & & | \\ \hline & & & & 1 & 0 & | & & | \\ & & & & 0 & 1 & | & & | \\ \hline & & & & & & 0 & 1 & | \\ & & & & & & 1 & 0 & | \end{pmatrix}$$

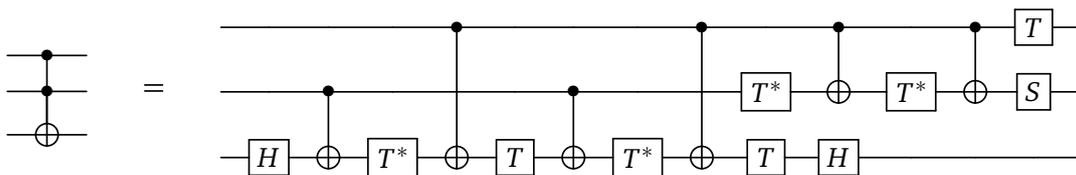
Exercice.

Montrer qu'une porte Toffoli permet de réaliser une porte *CNOT*. Il suffit d'imposer le qubit $|1\rangle$ en entrée de la seconde ligne.



Exercice (Difficile).

On peut réaliser une porte de Toffoli à partir de plusieurs portes *CNOT* et de portes élémentaires *S*, *H*, *T* et son adjointe T^* .



Essayer de prouver cette construction, soit par un calcul théorique, soit expérimentalement à l'aide d'un ordinateur (voir le chapitre « Utiliser un ordinateur quantique (avec Qiskit) »).

On rappelle qu'une porte *H* de Hadamard est définie par la matrice :

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

La porte *S* est appelé « porte phase » et est définie par la matrice :

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

La porte $\frac{\pi}{8}$ est définie par la matrice unitaire :

$$T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{4}} \end{pmatrix}$$

et donc

$$T^* = \begin{pmatrix} 1 & 0 \\ 0 & e^{-i\frac{\pi}{4}} \end{pmatrix}.$$

1.2. Théorème d'universalité

Théorème 1.

La porte de Toffoli est universelle : n'importe quelle fonction logique $f : \{0, 1\}^n \rightarrow \{0, 1\}$ peut être réalisée par un circuit quantique ne comportant que des portes de Toffoli.

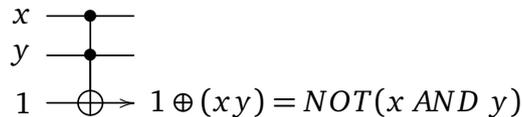
Remarque. La réalisation pratique requiert l'ajout de lignes auxiliaires.

Preuve.

- On sait que n'importe quelle fonction logique $f : \{0, 1\}^n \rightarrow \{0, 1\}$ peut être réalisée par un circuit classique ne comportant que des portes *NAND* (voir le chapitre « Informatique classique »).



- On réalise facilement l'équivalent d'une porte *NAND* à l'aide d'une porte de Toffoli en l'initialisant avec un 1 sur la troisième ligne. L'entrée correspond aux deux premières lignes et la sortie à la troisième ligne.



Pour vérifier que cela fonctionne, il faut remarquer que pour des bits x, y valant 0 ou 1 alors xy est la même chose que $x \text{ AND } y$, et donc $1 \oplus (xy) = \text{NOT}(x \text{ AND } y)$.

- Conclusion : on réalise la fonction $f : \{0, 1\}^n \rightarrow \{0, 1\}$ en substituant chaque porte *NAND* par une porte de Toffoli, avec un 1 sur sa troisième ligne.

2. Oracle

2.1. Définition

Le groupe $\mathbb{Z}/n\mathbb{Z}$.

Le groupe $(\mathbb{Z}/n\mathbb{Z}, +)$ correspond à l'ensemble des entiers modulo n . On peut représenter ce groupe par l'ensemble $\{0, 1, \dots, n-1\}$, avec la convention que $n \equiv 0, n+1 \equiv 1, \dots$. La loi de ce groupe est l'addition. On a déjà rencontré le groupe $\mathbb{Z}/2\mathbb{Z}$ (cas $n = 2$) qui est l'ensemble $\{0, 1\}$ muni de l'addition binaire notée « \oplus » (en préférence à « $+$ ») qui vérifie $1 \oplus 1 = 0$, ce qui est cohérent car $1 + 1 = 2 \equiv 0$ modulo 2.

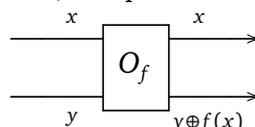
Oracle.

Nous allons associer à une fonction f un *oracle*. L'oracle d'une fonction f est un circuit quantique dont on explicite seulement l'entrée et la sortie (qui dépend de f). C'est une sorte de boîte noire, car nous n'avons pas besoin de connaître les détails du circuit qui réalise un oracle.

Cas $f : \mathbb{Z}/2\mathbb{Z} \rightarrow \mathbb{Z}/2\mathbb{Z}$.

C'est le cas déjà rencontré dans le chapitre « Un premier algorithme quantique », la fonction était alors notée $f : \{0, 1\} \rightarrow \{0, 1\}$.

Voici la transformation effectuée par un oracle, lorsque les entrées sont des bits classiques 0 ou 1 :



Il y a deux lignes pour l'entrée de l'oracle et deux lignes pour la sortie. La première sortie laisse la première entrée inchangée. Pour la seconde sortie : si x et y sont 0 ou 1 alors la seconde sortie est $y \oplus f(x)$; c'est donc y si $f(x) = 0$ et $NON(y)$ si $f(x) = 1$.

Ainsi l'oracle associé à f fournit une fonction

$$F : \mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z} \longrightarrow \mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z}$$

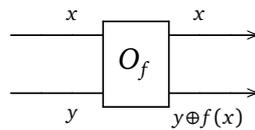
$$(x, y) \longmapsto (x, y \oplus f(x))$$

Nous verrons plus tard comment cela définit naturellement une transformation quantique sur les 2-qubits. Pour l'instant nous généralisons l'oracle au cas d'autres fonctions.

Cas $f : \mathbb{Z}/n\mathbb{Z} \rightarrow \mathbb{Z}/2\mathbb{Z}$.

Cette situation correspondra à l'algorithme de Grover. On fixe $n \geq 2$ et on considère une fonction quelconque $f : \mathbb{Z}/n\mathbb{Z} \rightarrow \mathbb{Z}/2\mathbb{Z}$ que l'on peut aussi voir comme une fonction $f : \{0, 1, \dots, n-1\} \rightarrow \{0, 1\}$.

La transformation de l'oracle, pour $x \in \mathbb{Z}/n\mathbb{Z}$ et $y \in \mathbb{Z}/2\mathbb{Z}$, renvoie une nouvelle fois x (élément de $\mathbb{Z}/n\mathbb{Z}$) et $y \oplus f(x)$ (élément de $\mathbb{Z}/2\mathbb{Z}$).



On obtient ainsi :

$$F : \mathbb{Z}/n\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z} \longrightarrow \mathbb{Z}/n\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z}$$

$$(x, y) \longmapsto (x, y \oplus f(x))$$

Exemple.

Fixons $\ell \in \{0, \dots, n-1\}$ un entier et $f : \mathbb{Z}/n\mathbb{Z} \rightarrow \mathbb{Z}/2\mathbb{Z}$ tel que $f(x) = 0$ pour tout x , sauf $f(\ell) = 1$. Alors :

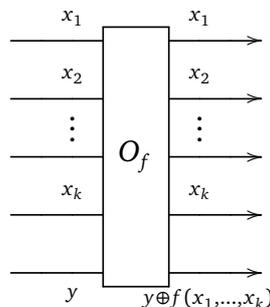
- pour $x \neq \ell$ et $y = 0$ on a $y \oplus f(x) = 0$,
- pour $x \neq \ell$ et $y = 1$ on a $y \oplus f(x) = 1$,
- pour $x = \ell$ et $y = 0$ on a $y \oplus f(x) = 1$,
- pour $x = \ell$ et $y = 1$ on a $y \oplus f(x) = 1 \oplus 1 = 0$.

Cas $f : (\mathbb{Z}/2\mathbb{Z})^k \rightarrow \mathbb{Z}/2\mathbb{Z}$.

Cette situation correspondra à l'algorithme de Deutsch–Jozsa.

On fixe $k \geq 1$ et on considère une fonction quelconque $f : (\mathbb{Z}/2\mathbb{Z})^k \rightarrow \mathbb{Z}/2\mathbb{Z}$ que l'on peut aussi voir comme une fonction $f : \{0, 1\}^k \rightarrow \{0, 1\}$.

La transformation de l'oracle, pour $x = (x_1, \dots, x_k) \in (\mathbb{Z}/2\mathbb{Z})^k$ et $y \in \mathbb{Z}/2\mathbb{Z}$, renvoie $x = (x_1, \dots, x_k)$ (élément de $(\mathbb{Z}/2\mathbb{Z})^k$) et $y \oplus f(x)$ (élément de $\mathbb{Z}/2\mathbb{Z}$).



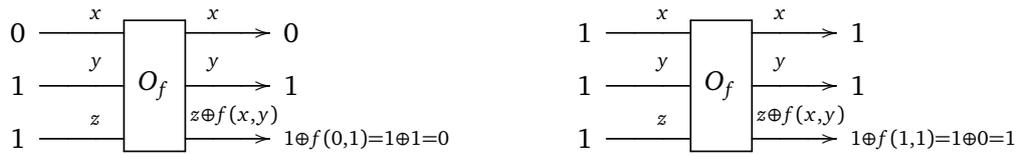
On obtient ainsi :

$$F : (\mathbb{Z}/2\mathbb{Z})^k \times \mathbb{Z}/2\mathbb{Z} \longrightarrow (\mathbb{Z}/2\mathbb{Z})^k \times \mathbb{Z}/2\mathbb{Z}$$

$$(x_1, \dots, x_k, y) \longmapsto (x_1, \dots, x_k, y \oplus f(x_1, \dots, x_k))$$

Exemple.

On considère $f : (\mathbb{Z}/2\mathbb{Z})^2 \rightarrow \mathbb{Z}/2\mathbb{Z}$ définie par $f(x, y) = x \text{ XOR } y$. Voici quelques exemples d'action de l'oracle :



Autrement dit $F(0, 1, 1) = (0, 1, 0)$ et $F(1, 1, 1) = (1, 1, 1)$.

On pourrait généraliser l'oracle au cas d'une fonction $f : E \rightarrow E'$ pour lequel l'oracle associé serait une fonction $F : E \times E' \rightarrow E \times E'$ défini par $F(x, y) = (x, y \oplus f(x))$ où « \oplus » est une addition dans E' .

2.2. L'oracle est bijectif

Quel peut être l'intérêt d'un oracle ? Plus précisément quel est l'avantage de la fonction F par rapport à f ? Considérons une fonction $f : E \rightarrow \mathbb{Z}/2\mathbb{Z}$ quelconque. En particulier elle n'est pas supposée bijective, par contre la fonction F associée à l'oracle va l'être.

Lemme 1.

Soit $f : E \rightarrow \mathbb{Z}/2\mathbb{Z}$ une fonction quelconque, alors la fonction $F : E \times \mathbb{Z}/2\mathbb{Z} \rightarrow E \times \mathbb{Z}/2\mathbb{Z}$ définie par $F(x, y) = (x, y \oplus f(x))$ est bijective.

Démonstration. Il suffit de trouver la bijection réciproque de F : nous allons montrer que cette réciproque est F elle-même.

Partons de

$$F(x, y) = (x, y \oplus f(x))$$

donc

$$F(F(x, y)) = F(x, y \oplus f(x)) = (x, y \oplus f(x) \oplus f(x)) = (x, y).$$

En effet, pour $a \in \mathbb{Z}/2\mathbb{Z}$ on a $a \oplus a = 2a = 0$ (car $0 \oplus 0 = 0$ et $1 \oplus 1 = 0$), donc $x \oplus a \oplus a = x$. Ainsi F est bijective et de plus $F^{-1} = F$ (autrement dit $F \circ F = \text{id}$). □

Exemple.

Reprenons $f : (\mathbb{Z}/2\mathbb{Z})^2 \rightarrow \mathbb{Z}/2\mathbb{Z}$ définie par $f(x, y) = x \text{ XOR } y$. Alors F est bien bijective :

$(0, 0, 0)$	\xrightarrow{F}	$(0, 0, 0)$	$(0, 0, 1)$	\xrightarrow{F}	$(0, 0, 1)$
$(0, 1, 0)$	\xrightarrow{F}	$(0, 1, 1)$	$(0, 1, 1)$	\xrightarrow{F}	$(0, 1, 0)$
$(1, 0, 0)$	\xrightarrow{F}	$(1, 0, 1)$	$(1, 0, 1)$	\xrightarrow{F}	$(1, 0, 0)$
$(1, 1, 0)$	\xrightarrow{F}	$(1, 1, 0)$	$(1, 1, 1)$	\xrightarrow{F}	$(1, 1, 1)$

Exemple.

Soit $f : \mathbb{Z}/n\mathbb{Z} \rightarrow \mathbb{Z}/2\mathbb{Z}$ telle que $f(x) = 0$ pour tout x , sauf $f(\ell) = 1$, pour un entier $\ell \in \{0, \dots, n-1\}$ fixé.

- Pour $x \neq \ell$, $f(x) = 0$ donc $F(x, y) = (x, y \oplus f(x)) = (x, y)$.
- pour $x = \ell$, $f(x) = 1$ donc $F(x, y) = (x, y \oplus 1) = (x, \text{NON}(y))$.

L'application F est bijective.

2.3. Transformation quantique

Considérons le cas d'une fonction $f : (\mathbb{Z}/2\mathbb{Z})^k \rightarrow \mathbb{Z}/2\mathbb{Z}$. L'oracle fournit une fonction $F : (\mathbb{Z}/2\mathbb{Z})^{k+1} \rightarrow (\mathbb{Z}/2\mathbb{Z})^{k+1}$ en ayant considéré $(\mathbb{Z}/2\mathbb{Z})^k \times \mathbb{Z}/2\mathbb{Z} = (\mathbb{Z}/2\mathbb{Z})^{k+1}$. Voyons la transformation quantique associée sur les $(k + 1)$ -qubits.

Les $(k + 1)$ -qubits sont engendrés par la base canonique formée des 2^{k+1} qubits de base :

$$\underbrace{|0.0 \dots 0\rangle}_{k+1 \text{ bits}} \quad |0.0 \dots 1\rangle \quad \dots \quad |1.1 \dots 1\rangle.$$

La fonction F (définie sur des $(k + 1)$ -bits) s'étend naturellement en une fonction \tilde{F} sur les vecteurs de la base des $(k + 1)$ -qubits :

$$\begin{aligned} |e_0\rangle = |0.0 \dots 0\rangle &\xrightarrow{\tilde{F}} |F(0, 0, \dots, 0)\rangle = |f_0\rangle \\ |e_1\rangle = |0.0 \dots 1\rangle &\xrightarrow{\tilde{F}} |F(0, 0, \dots, 1)\rangle = |f_1\rangle \\ &\dots \\ |e_{2^{k+1}-1}\rangle = |1.1 \dots 1\rangle &\xrightarrow{\tilde{F}} |F(1, 1, \dots, 1)\rangle = |f_{2^{k+1}-1}\rangle \end{aligned}$$

Maintenant que \tilde{F} est définie sur les vecteurs de la base par la relation $\tilde{F}(|e_i\rangle) = |F(e_i)\rangle = |f_i\rangle$, elle s'étend par linéarité à tous les $(k + 1)$ -qubits. Ainsi on obtient

$$\tilde{F} : \mathbb{C}^{2^{k+1}} \longrightarrow \mathbb{C}^{2^{k+1}}$$

et pour un $(k + 1)$ -qubit

$$|\psi\rangle = \sum_{i=0}^{2^{k+1}-1} \alpha_i |e_i\rangle,$$

avec $\alpha_i \in \mathbb{C}$, on obtient le $(k + 1)$ -qubit :

$$\tilde{F}(|\psi\rangle) = \sum_{i=0}^{2^{k+1}-1} \alpha_i |f_i\rangle.$$

Comme F est bijective alors \tilde{F} envoie l'ensemble des vecteurs de la base canonique sur ces mêmes vecteurs de la base canonique (autrement dit \tilde{F} permute les vecteurs de la base). Ainsi \tilde{F} est une transformation unitaire (voir la section 3).

Exemple.

Reprenons l'exemple de la fonction $f : (\mathbb{Z}/2\mathbb{Z})^2 \rightarrow \mathbb{Z}/2\mathbb{Z}$ définie par $f(x, y) = x \text{ XOR } y$. Nous avons déjà calculé F , ce qui donne les valeurs de \tilde{F} sur les 3-qubits de base. Par exemple $\tilde{F} |0.1.0\rangle = |0.1.1\rangle$, $\tilde{F} |0.1.1\rangle = |0.1.0\rangle, \dots$ Pour un 3-qubit quelconque :

$$|\psi\rangle = \alpha_0 |0.0.0\rangle + \alpha_1 |0.0.1\rangle + \alpha_2 |0.1.0\rangle + \alpha_3 |0.1.1\rangle + \alpha_4 |1.0.0\rangle + \alpha_5 |1.0.1\rangle + \alpha_6 |1.1.0\rangle + \alpha_7 |1.1.1\rangle$$

alors

$$\tilde{F} |\psi\rangle = \alpha_0 |0.0.0\rangle + \alpha_1 |0.0.1\rangle + \alpha_3 |0.1.0\rangle + \alpha_2 |0.1.1\rangle + \alpha_5 |1.0.0\rangle + \alpha_4 |1.0.1\rangle + \alpha_6 |1.1.0\rangle + \alpha_7 |1.1.1\rangle.$$

2.4. Matrice de l'oracle

Nous allons calculer la matrice de l'oracle, c'est-à-dire la matrice de l'application \tilde{F} .

On rappelle que

$$(x, y) \xrightarrow{F} (x, y \oplus f(x))$$

où $x = (x_1, \dots, x_k)$ est un k -bit et y un 1-bit. La fonction F est naturellement étendue aux qubits de base par la formule :

$$|x.y\rangle \xrightarrow{\tilde{F}} |x.y \oplus f(x)\rangle$$

Calculons explicitement l'image de chacun des vecteurs $|e_i\rangle$ de la base canonique de $(k + 1)$ -qubits.

$$|e_0\rangle = |\underbrace{0 \dots 0}_x \cdot \underbrace{0}_y\rangle \xrightarrow{\tilde{F}} |\underbrace{0 \dots 0}_x \cdot \underbrace{0 \oplus f(0, \dots, 0)}_{0 \text{ ou } 1}\rangle = \begin{cases} |e_0\rangle & \text{si } f(0, \dots, 0) = 0 \\ |e_1\rangle & \text{si } f(0, \dots, 0) = 1 \end{cases}$$

De même

$$|e_1\rangle = |\underbrace{0 \dots 0}_x \cdot \underbrace{1}_y\rangle \xrightarrow{\tilde{F}} |\underbrace{0 \dots 0}_x \cdot \underbrace{1 \oplus f(0, \dots, 0)}_{1 \text{ ou } 0}\rangle = \begin{cases} |e_1\rangle & \text{si } f(0, \dots, 0) = 0 \\ |e_0\rangle & \text{si } f(0, \dots, 0) = 1 \end{cases}$$

De façon générale

$$\begin{cases} |e_{2i}\rangle \xrightarrow{\tilde{F}} |e_{2i}\rangle \\ |e_{2i+1}\rangle \xrightarrow{\tilde{F}} |e_{2i+1}\rangle \end{cases} \quad \text{ou} \quad \begin{cases} |e_{2i}\rangle \xrightarrow{\tilde{F}} |e_{2i+1}\rangle \\ |e_{2i+1}\rangle \xrightarrow{\tilde{F}} |e_{2i}\rangle \end{cases}$$

La sous-matrice de \tilde{F} dans la base (e_{2i}, e_{2i+1}) est donc

$$I_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \text{ou} \quad J_2 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

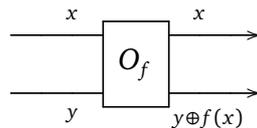
Ainsi la matrice de \tilde{F} dans la base canonique des $(k + 1)$ -qubits est la matrice suivante, qui est une matrice diagonale par blocs, chaque bloc étant I_2 ou J_2 :

$$A = \begin{pmatrix} I_2/J_2 & & & \\ & I_2/J_2 & & \\ & & \ddots & \\ & & & \ddots & \\ & & & & I_2/J_2 \end{pmatrix} \in M_{2^{k+1}}(\mathbb{C}).$$

On a bien sûr $I_2^2 = I_2$, mais aussi $J_2^2 = I_2$ et $J_2^* = J_2$, donc $A^*A = I$, ce qui prouve que A est une matrice unitaire. (On le savait déjà car l'application associée \tilde{F} est unitaire, voir ci-dessus.)

2.5. Oracle pour $f = NOT$

Considérons $f : \mathbb{Z}/2\mathbb{Z} \rightarrow \mathbb{Z}/2\mathbb{Z}$ défini par $f(0) = 1$ et $f(1) = 0$. C'est donc la négation : $f(x) = NOT(x) = 1 \oplus x$. Décrivons l'oracle de f .



L'application F est :

$$F : \mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z} \longrightarrow \mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z} \\ (x, y) \longmapsto (x, y \oplus f(x)) = (x, 1 \oplus x \oplus y).$$

ce qui donne concrètement :

$$\begin{array}{ccc} (0, 0) \xrightarrow{F} (0, 1) & & |0.0\rangle \xrightarrow{\tilde{F}} |0.1\rangle \\ (0, 1) \xrightarrow{F} (0, 0) & \text{donc} & |0.1\rangle \xrightarrow{\tilde{F}} |0.0\rangle \\ (1, 0) \xrightarrow{F} (1, 0) & & |1.0\rangle \xrightarrow{\tilde{F}} |1.0\rangle \\ (1, 1) \xrightarrow{F} (1, 1) & & |1.1\rangle \xrightarrow{\tilde{F}} |1.1\rangle \end{array}$$

Ainsi la matrice de l'oracle est :

$$\begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

2.6. Oracle pour $f = AND$

Exercice.

Effectuer le même travail mais cette fois avec la fonction de deux variables f définie par AND .

Soit $f : (\mathbb{Z}/2\mathbb{Z})^2 \rightarrow \mathbb{Z}/2\mathbb{Z}$ définie par $f(x, y) = x AND y = xy$.

Nous allons voir que l'oracle associé à ce f est exactement une porte de Toffoli ($CCNOT$).



En détails :

1. Calculer l'image par l'oracle de chacun des vecteurs de la base des 3-qubits : $|0.0.0\rangle, |0.0.1\rangle, \dots, |1.1.1\rangle$.
2. En déduire que l'oracle associé à ce f est équivalent à une porte de Toffoli, en vérifiant que le résultat est le même que l'action de la porte de Toffoli sur les 3-qubits de base.
3. Vérifier que l'application F (ou \tilde{F}) est bijective alors que f ne l'est pas.
4. Calculer la matrice de l'oracle (et retrouver la matrice de la porte de Toffoli).

3. Matrices unitaires

Nous reprenons l'étude des matrices unitaires, ici de taille quelconque, les matrices 2×2 ayant déjà été étudiées dans le chapitre « Vecteurs et matrices ».

3.1. Produit scalaire hermitien

Rappelons quelques définitions et propriétés.

- Le **produit scalaire hermitien** $\langle u|v\rangle$ des deux vecteurs u et v est défini par :

$$u = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \quad v = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} \quad \langle u|v\rangle = x_1^* \cdot y_1 + x_2^* \cdot y_2 + \dots + x_n^* \cdot y_n = \sum_{i=1}^n x_i^* \cdot y_i.$$

- Le produit scalaire permet de calculer la norme : $\|u\|^2 = \langle u|u\rangle$.
- Le produit scalaire est anti-linéaire à gauche et linéaire à droite. Pour $\lambda \in \mathbb{C}$:

$$\langle \lambda u|v\rangle = \lambda^* \langle u|v\rangle \quad \text{et} \quad \langle u|\lambda v\rangle = \lambda \langle u|v\rangle.$$

- La **matrice adjointe** de A est la matrice A^* obtenue par transposition et conjugaison complexe :

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1p} \\ a_{21} & a_{22} & \dots & a_{2p} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \dots & a_{np} \end{pmatrix} \quad A^* = \begin{pmatrix} a_{11}^* & a_{21}^* & \dots & a_{n1}^* \\ a_{12}^* & a_{22}^* & \dots & a_{n2}^* \\ \vdots & \vdots & & \vdots \\ a_{1p}^* & a_{2p}^* & \dots & a_{np}^* \end{pmatrix}.$$

- La notation « ket » $|\phi\rangle$ désigne un vecteur écrit sous forme colonne :

$$|\phi\rangle = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}.$$

- La notation « bra » $\langle \phi |$ désigne un vecteur ligne, obtenu comme l'adjoint :

$$\text{si } |\phi\rangle = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \quad \text{alors} \quad \langle \phi | = |\phi\rangle^* = (x_1^* \quad \cdots \quad x_n^*).$$

- Ainsi l'écriture $\langle \cdot | \cdot \rangle$ désigne de façon cohérente à la fois le produit scalaire hermitien et la multiplication matricielle d'un vecteur ligne par un vecteur colonne (qui donne un scalaire) :

$$|\phi\rangle = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \quad |\psi\rangle = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \quad \langle \phi | \psi \rangle = x_1^* \cdot y_1 + x_2^* \cdot y_2 + \cdots + x_n^* \cdot y_n.$$

Proposition 1.

$$\boxed{\langle Au | v \rangle = \langle u | A^* v \rangle}$$

Démonstration. Si $A = (a_{ij})$ et $u = (x_i)$ alors Au est un vecteur dont le terme de rang i est $(Au)_i = \sum_{j=1}^n a_{ij}x_j$. Ainsi si $v = (y_i)$ alors $\langle Au | v \rangle$ est la somme sur i de termes

$$\left(\sum_{j=1}^n a_{ij}x_j \right)^* y_i,$$

et donc

$$\langle Au | v \rangle = \sum_{i=1}^n \sum_{j=1}^n a_{ij}^* x_j^* y_i.$$

D'autre part $\langle u | A^* v \rangle$ est la somme sur i de termes

$$x_i^* \left(\sum_{j=1}^n a_{ij}^* y_j \right),$$

ce qui prouve que $\langle u | A^* v \rangle = \langle Au | v \rangle$. □

3.2. Caractérisation des matrices unitaires

Définition.

Une matrice $A \in M_n(\mathbb{C})$ est **unitaire** si :

$$\boxed{A^* A = I}$$

On note U_n l'ensemble des matrices unitaires.

Si A est une matrice unitaire alors on a

$$A^{-1} = A^* \quad \text{et} \quad AA^* = I.$$

Proposition 2.

L'ensemble des matrices unitaires forme un groupe. En particulier $I \in U_n$ et si $A, B \in U_n$ alors $AB \in U_n$ et $A^{-1} \in U_n$.

Pour la suite nous aurons besoin de la notion suivante :

Définition.

Les vecteurs (e_1, e_2, \dots, e_n) forment une **base orthonormale** de \mathbb{C}^n si

$$\langle e_i | e_j \rangle = 0 \quad \text{pour tout } i \neq j \quad \text{et} \quad \|e_i\| = 1 \quad \text{pour tout } i = 1, \dots, n.$$

On peut rassembler les deux conditions en une seule

$$\langle e_i | e_j \rangle = \delta_{i,j}$$

où $\delta_{i,j}$ est le symbole de Kronecker :

$$\delta_{i,j} = 0 \quad \text{si } i \neq j \quad \text{et} \quad \delta_{i,i} = 1.$$

Proposition 3.

Les assertions suivantes sont équivalentes :

- (i) La matrice $A \in M_n(\mathbb{C})$ est unitaire.
- (ii) A préserve le produit scalaire hermitien : $\langle Au | Av \rangle = \langle u | v \rangle$ quels que soient $u, v \in \mathbb{C}^n$.
- (iii) A préserve les longueurs : $\|Au\| = \|u\|$ quel que soit $u \in \mathbb{C}^n$.
- (iv) Si (e_i) est une base orthonormale de \mathbb{C}^n , alors (Ae_i) est aussi une base orthonormale.
- (v) Les vecteurs colonnes (f_i) de A forment une base orthonormale de \mathbb{C}^n .

Démonstration.

- (i) \implies (ii) $\langle Au | Av \rangle = \langle u | A^* Av \rangle = \langle u | v \rangle$
- (ii) \implies (iii) $\|Au\|^2 = \langle Au | Au \rangle = \langle u | u \rangle = \|u\|^2$
- (iii) \implies (iv) Notons $f_i = Ae_i$. Alors $\|f_i\| = \|e_i\| = 1$. Soit $i \neq j$, comme $\|e_i + e_j\|^2 = 2$ alors $\|f_i + f_j\|^2 = \|A(e_i + e_j)\|^2 = \|e_i + e_j\|^2 = 2$. Or

$$\begin{aligned} \|f_i + f_j\|^2 &= \langle f_i + f_j | f_i + f_j \rangle = \langle f_i | f_i \rangle + \langle f_i | f_j \rangle + \langle f_j | f_i \rangle + \langle f_j | f_j \rangle \\ &= \|f_i\|^2 + \langle f_i | f_j \rangle + \langle f_i | f_j \rangle^* + \|f_j\|^2 = \|f_i\|^2 + 2 \operatorname{Re}(\langle f_i | f_j \rangle) + \|f_j\|^2 \end{aligned}$$

Comme $\|f_i + f_j\|^2 = 2$, $\|f_i\|^2 = 1$ et $\|f_j\|^2 = 1$ alors $2 \operatorname{Re}(\langle f_i | f_j \rangle) = 0$.

De même

$$\|f_i + if_j\|^2 = \|f_i\|^2 - 2 \operatorname{Im}(\langle f_i | f_j \rangle) + \|f_j\|^2$$

alors $2 \operatorname{Im}(\langle f_i | f_j \rangle) = 0$. Ainsi $\langle f_i | f_j \rangle = 0$ et (f_i) forment une base orthonormée.

- (iv) \implies (v) Soit (e_i) la base canonique, alors les $f_i = Ae_i$ sont les vecteurs colonnes de A . Comme (e_i) est une base orthonormale, alors (f_i) l'est aussi.
- (v) \implies (i) Notons $M = AA^* - I$. Notons (e_i) la base canonique et $(f_i) = (Ae_i)$ les vecteurs colonnes de A .

$$\langle Me_i | e_j \rangle = \langle AA^* e_i - e_i | e_j \rangle = \langle AA^* e_i | e_j \rangle - \langle e_i | e_j \rangle = \langle Ae_i | Ae_j \rangle - \delta_{i,j} = \langle f_i | f_j \rangle - \delta_{i,j} = \delta_{i,j} - \delta_{i,j} = 0.$$

Fixons i , comme $\langle Me_i | e_j \rangle = 0$ (scalaire nul) pour tout vecteur e_j de la base, alors $Me_i = 0$ (vecteur nul). Maintenant comme $Me_i = 0$ pour tout vecteur e_i de la base, alors $M = 0$ (matrice nulle). Ainsi $AA^* = I$ donc A est unitaire. □

3.3. Porte quantique

Nous avons vu différentes portes quantiques, voici maintenant la définition générale :

Définition.

Une **porte quantique** est la transformation $|\psi\rangle \mapsto A|\psi\rangle$ où A est une matrice unitaire.

$$|\psi\rangle \longrightarrow \boxed{A} \longrightarrow A|\psi\rangle$$

Si l'entrée $|\psi\rangle$ est un n -qubit, alors A une matrice de taille 2^n (donc $A \in U_{2^n}$), la sortie est un n -qubit.

Comme la matrice A est unitaire alors en particulier la transformation est inversible. C'est une différence majeure par rapport à une porte de l'informatique classique (par exemple une porte AND n'est pas inversible).

3.4. Théorèmes de réalisation

Même si une porte quantique est donnée par une matrice unitaire A quelconque, cette porte quantique peut être réalisée de façon équivalente par un circuit composé de portes bien connues. Nous allons voir plusieurs résultats de réalisations que nous énonçons sans démonstration.

Théorème 2.

Toute porte quantique à n -qubits peut être réalisée de façon équivalente par un circuit ne comportant que des portes CNOT et des portes à 1-qubit.

Ainsi l'étude de n'importe quel circuit quantique se ramène à l'étude de deux types de portes et à leur composition. Le défaut de ce résultat, c'est que la réalisation se fait à l'aide de portes parmi une infinité de possibilités. En effet, une porte à 1-qubit est définie par une matrice unitaire $A \in M_2(\mathbb{C})$, et il y a une infinité de telles matrices.

Le résultat suivant construit des circuits avec seulement trois types de portes, la contrepartie c'est que l'on n'obtient pas exactement le circuit voulu, mais une approximation.

Théorème 3.

Toute porte quantique à n -qubits peut être approchée d'autant près que l'on veut par un circuit ne comportant que des portes H de Hadamard, des portes T (dite « porte $\frac{\pi}{8}$ ») et des portes CNOT.

On rappelle qu'une porte H de Hadamard est définie par la matrice :

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

La porte $\frac{\pi}{8}$ est définie par la matrice unitaire :

$$T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{4}} \end{pmatrix}$$

qui tient son nom de son écriture sous la forme :

$$T = e^{i\frac{\pi}{8}} \begin{pmatrix} e^{-i\frac{\pi}{8}} & 0 \\ 0 & e^{i\frac{\pi}{8}} \end{pmatrix}.$$

Le théorème de Solovay–Kitaev est une version améliorée du théorème précédent et affirme de plus qu'on peut réaliser le circuit en utilisant assez peu de portes.

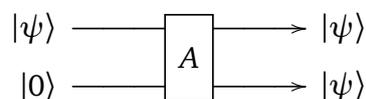
4. Théorème de non-clonage quantique

Un ordinateur classique est modélisé par une machine de Turing et est capable de lire une série de bits et de les dupliquer à un autre endroit. Nous allons voir que ce n'est pas le cas pour un ordinateur quantique. En fait on peut copier un qubit, mais en créant la copie on perd l'original. On parle ainsi de « non-clonage quantique ».

4.1. Non-clonage des 1-qubits

Théorème 4.

Il n'existe pas de porte quantique qui réalise le clonage des 1-qubits, c'est-à-dire telle que pour tout 1-qubit $|\psi\rangle$ on ait :



Démonstration. Raisonnons par l'absurde et supposons que le clonage quantique soit possible. Cela signifie qu'il existe une porte quantique qui réalise ce clonage, c'est-à-dire qu'il existe une matrice $A \in M_4(\mathbb{C})$ unitaire

telle que :

$$A|\psi.0\rangle = |\psi.\psi\rangle \quad \text{pour tout 1-qubit } |\psi\rangle.$$

Comme cette égalité est vraie pour tous les 1-qubits, c'est également le cas :

- pour $|\psi_0\rangle = |0\rangle$, donc $A|0.0\rangle = |0.0\rangle$,
- pour $|\psi_1\rangle = |1\rangle$, donc $A|1.0\rangle = |1.1\rangle$,
- et pour $|\psi_2\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$.

Nous allons détailler ce que cela implique pour $|\psi_2\rangle$.

- D'une part comme $|\psi_2\rangle = \frac{1}{\sqrt{2}}(|\psi_0\rangle + |\psi_1\rangle)$.

$$\begin{aligned} A|\psi_2.0\rangle &= A\left(\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle).|0\rangle\right) \\ &= \frac{1}{\sqrt{2}}A|0.0\rangle + \frac{1}{\sqrt{2}}A|1.0\rangle \\ &= \frac{1}{\sqrt{2}}|0.0\rangle + \frac{1}{\sqrt{2}}|1.1\rangle \end{aligned}$$

On retient que :

$$A|\psi_2.0\rangle = \frac{1}{\sqrt{2}}(|0.0\rangle + |1.1\rangle). \tag{1}$$

- D'autre part par le clonage de $|\psi_2\rangle$:

$$\begin{aligned} A|\psi_2.0\rangle &= |\psi_2.\psi_2\rangle \\ &= \frac{1}{2}(|0\rangle + |1\rangle)(|0\rangle + |1\rangle) \\ &= \frac{1}{2}(|0.0\rangle + |0.1\rangle + |1.0\rangle + |1.1\rangle) \end{aligned}$$

On a prouvé :

$$A|\psi_2.0\rangle = \frac{1}{2}(|0.0\rangle + |0.1\rangle + |1.0\rangle + |1.1\rangle). \tag{2}$$

Nous pouvons maintenant conclure à partir des équations (1) et (2) :

$$\frac{1}{\sqrt{2}}(|0.0\rangle + |1.1\rangle) = \frac{1}{2}(|0.0\rangle + |0.1\rangle + |1.0\rangle + |1.1\rangle)$$

donc

$$\left(\frac{1}{2} - \frac{1}{\sqrt{2}}\right)|0.0\rangle + \frac{1}{2}|0.1\rangle + \frac{1}{2}|1.0\rangle + \left(\frac{1}{2} - \frac{1}{\sqrt{2}}\right)|1.1\rangle = 0. \tag{3}$$

Souvenons-nous que $(|0.0\rangle, |0.1\rangle, |1.0\rangle, |1.1\rangle)$ est une base de \mathbb{C}^4 , donc

$$\text{si } \alpha|0.0\rangle + \beta|0.1\rangle + \gamma|1.0\rangle + \delta|1.1\rangle = 0 \quad \text{alors} \quad \alpha = 0, \beta = 0, \gamma = 0, \delta = 0.$$

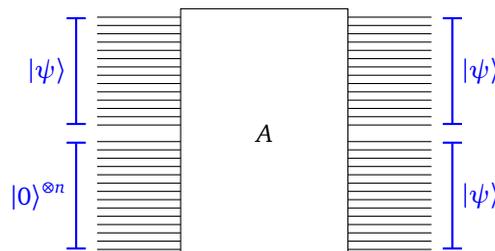
Dans notre cas cela implique que les coefficients de l'équation (3) sont tous nuls, et donc par exemple $\frac{1}{2} = 0$ ce qui fournit une contradiction.

Conclusion : une telle matrice A qui réalise le clonage ne peut exister. □

4.2. Cas général

Théorème 5.

Il n'existe pas de porte quantique qui réalise le clonage d'un n-qubit, c'est-à-dire telle que pour tout n-qubit $|\psi\rangle$ on ait :



La preuve pour le cas général des n -qubits est le même calcul que pour le cas des 1-qubits. On note $|e_0\rangle, |e_1\rangle, \dots, |e_{2^n-1}\rangle$ les vecteurs de la base canonique des n -qubits. On raisonne par l'absurde en supposant qu'il existe une matrice unitaire A telle que $A|\psi \cdot 0^{\otimes n}\rangle = |\psi \cdot \psi\rangle$ quel que soit le n -qubit $|\psi\rangle$ (ici $|0\rangle^{\otimes n} = |0.0 \dots 0\rangle$).

Ensuite on pose :

- pour $|\psi_0\rangle = |e_0\rangle$, donc $A|e_0 \cdot 0^{\otimes n}\rangle = |e_0 \cdot e_0\rangle$,
- pour $|\psi_1\rangle = |e_1\rangle$, donc $A|e_1 \cdot 0^{\otimes n}\rangle = |e_1 \cdot e_1\rangle$.
- Pour $|\psi_2\rangle = \frac{1}{\sqrt{2}}(|e_0\rangle + |e_1\rangle)$, on obtient une contradiction en écrivant d'une part

$$A|\psi_2 \cdot 0^{\otimes n}\rangle = \frac{1}{\sqrt{2}}(A|0 \cdot 0^{\otimes n}\rangle + A|e_1 \cdot 0^{\otimes n}\rangle)$$

et d'autre part

$$A|\psi_2 \cdot 0^{\otimes n}\rangle = |\psi_2 \cdot \psi_2\rangle.$$

Algorithme de Deutsch–Jozsa

Vidéo ■ partie 10. Algorithme de Deutsch–Jozsa

Nous expliquons et prouvons l'algorithme de Deutsch–Jozsa dans le cas général. C'est notre premier algorithme quantique qui supprime les algorithmes classiques et c'est aussi l'occasion d'introduire plusieurs notions utiles pour la suite.

1. Algorithme

1.1. Problème

Le problème à résoudre est la généralisation du problème rencontré dans le chapitre introductif « Un premier algorithme quantique ».

Terminologie. Soit une fonction $f : (\mathbb{Z}/2\mathbb{Z})^n \rightarrow \mathbb{Z}/2\mathbb{Z}$, que l'on peut aussi voir comme une fonction $f : \{0, 1\}^n \rightarrow \{0, 1\}$. Il y a 2^{2^n} fonctions différentes possibles mais nous n'allons considérer que deux types de fonctions.

- f est **constante** si pour tout (x_1, \dots, x_n) on a $f(x_1, \dots, x_n) = 0$ ou bien si pour tout (x_1, \dots, x_n) on a $f(x_1, \dots, x_n) = 1$. Il existe donc deux fonctions constantes.
- f est **équilibrée** s'il y a autant de n -uplets (x_1, \dots, x_n) tels que $f(x_1, \dots, x_n) = 0$ que de n -uplets (x_1, \dots, x_n) tels que $f(x_1, \dots, x_n) = 1$, autrement dit

$$\begin{aligned} & \text{Card} \{ (x_1, \dots, x_n) \in (\mathbb{Z}/2\mathbb{Z})^n \mid f(x_1, \dots, x_n) = 0 \} \\ &= \text{Card} \{ (x_1, \dots, x_n) \in (\mathbb{Z}/2\mathbb{Z})^n \mid f(x_1, \dots, x_n) = 1 \}. \end{aligned}$$

Il y a en tout $\binom{2^n}{2^{n-1}} = \frac{2^n}{(2^{n-1})^2}$ telles fonctions.

Pour $n > 1$ il existe beaucoup de fonctions qui ne sont ni constantes, ni équilibrées, par exemple une fonction qui prend une seule fois la valeur 1 et 0 ailleurs.

Problème. On nous donne une fonction $f : (\mathbb{Z}/2\mathbb{Z})^n \rightarrow \mathbb{Z}/2\mathbb{Z}$ en nous certifiant qu'elle est soit constante, soit équilibrée. C'est à vous de déterminer dans quelle catégorie elle se situe : constante ou équilibrée.

1.2. Solution classique

Pour ce problème, la complexité des algorithmes se mesure par le nombre d'évaluations $f(x_1, \dots, x_n)$ effectuées. Avec un ordinateur classique, la complexité du meilleur algorithme est exponentielle, d'ordre $O(2^n)$. Notons qu'il y a en tout $2^n = \text{Card}((\mathbb{Z}/2\mathbb{Z})^n)$ éléments dans l'ensemble de départ. Voici un algorithme classique dont la complexité est $2^{n-1} + 1$.

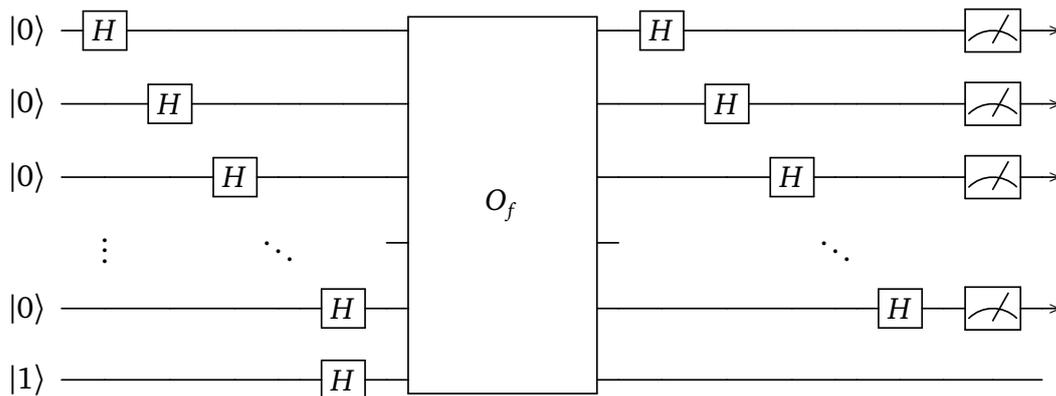
Algorithme.

- On évalue f sur $2^{n-1} + 1$ termes.
- Si toutes ces valeurs sont égales alors f est constante, sinon elle est équilibrée.

Plus précisément, aucun algorithme classique ne peut faire moins que $\frac{2^n}{2}$ évaluations. Bien sûr pour certaines fonctions f on pourrait obtenir une réponse plus rapide (par exemple dès que l'on obtient deux valeurs différentes, la fonction doit être équilibrée), mais dans le pire des cas il faut évaluer f sur plus de la moitié des éléments pour pouvoir conclure. En effet, si par exemple f s'annule sur la moitié des éléments, on ne peut pas déjà savoir si elle est constante ou équilibrée car les deux conclusions sont encore possibles.

1.3. Circuit quantique

Voici le circuit qui fournit l'algorithme quantique répondant au problème.



Les n premières lignes du circuit sont initialisées par $|0\rangle$, suivi de la transformation de Hadamard. La dernière ligne est initialisée par $|1\rangle$, suivi d'une porte de Hadamard. Ensuite on applique l'oracle associé à f . Enfin, on applique de nouveau une transformation de Hadamard sur les n premières lignes, suivi d'une mesure uniquement sur les n premières lignes.

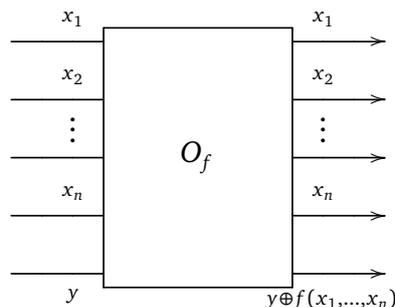
Le circuit n'est exécuté qu'une seule fois, autrement dit l'algorithme quantique est de complexité 1 car la fonction associée à l'oracle O_f n'est évaluée qu'une seule fois.

L'algorithme consiste simplement à exécuter le circuit.

Algorithme. Si la mesure des n premiers qubits de sortie est $0.0 \dots 0$ alors la fonction est constante, sinon la fonction est équilibrée.

Pour les exemples où $n = 1$ ou $n = 2$, nous renvoyons au chapitre « Un premier algorithme quantique ». La complexité de l'algorithme quantique est donc $O(1)$. On a vu que l'algorithme classique est de complexité exponentielle. En effet $O(2^{n-1}) = O(2^n)$ et $2^n = e^{n \ln 2}$. L'algorithme quantique est donc une amélioration exponentielle de l'algorithme classique ! Bien sûr le problème résolu ici est artificiel et assez peu intéressant mais nous avons maintenant la preuve que l'informatique quantique peut aller plus vite que l'informatique classique.

On rappelle que l'oracle associé à la fonction f agit ainsi : sur les n premières lignes $x_i \mapsto x_i$, sur la dernière ligne $y \mapsto y \oplus f(x_1, \dots, x_n)$.



2. Notation entière des qubits

2.1. Notation

La notation $|0.0\dots 0\rangle, |0.0\dots 1\rangle, \dots, |1.1\dots 1\rangle$ pour les n -qubits de la base canonique n'est pas pratique pour les calculs généraux et les preuves. En particulier comment noter un n -qubit quelconque de cette base? Nous allons introduire l'écriture d'un n -qubit de base par un seul entier : c'est tout simplement l'opération inverse de l'écriture binaire.

On fixe un entier $n \geq 1$. Soit $0 \leq k \leq 2^n - 1$. Notons \underline{k} l'écriture binaire de l'entier k sur n bits. L'*écriture entière* $|\underline{k}\rangle$ désigne le n -qubit de la base canonique associé à l'écriture binaire de k .

$$\begin{aligned} |\underline{0}\rangle &= |0.0\dots 0.0.0\rangle \\ |\underline{1}\rangle &= |0.0\dots 0.0.1\rangle \\ |\underline{2}\rangle &= |0.0\dots 0.1.0\rangle \\ |\underline{3}\rangle &= |0.0\dots 0.1.1\rangle \\ &\vdots \\ |\underline{2^n - 2}\rangle &= |1.1\dots 1.1.0\rangle \\ |\underline{2^n - 1}\rangle &= |1.1\dots 1.1.1\rangle \end{aligned}$$

On peut ainsi écrire facilement certains énoncés. Par exemple une fonction $f : (\mathbb{Z}/2\mathbb{Z})^n \rightarrow \mathbb{Z}/2\mathbb{Z}$ est constante si $f(\underline{k}) = 0$ pour tout k variant de 0 à $2^n - 1$ ou si $f(\underline{k}) = 1$ pour tout k variant de 0 à $2^n - 1$.

2.2. Exemples

Pour $n = 1$, il y a seulement deux qubits de base et on a $|\underline{0}\rangle = |0\rangle$ et $|\underline{1}\rangle = |1\rangle$.

Exemple.

Pour $n = 2$.

$$\begin{aligned} |\underline{0}\rangle &= |0.0\rangle \\ |\underline{1}\rangle &= |0.1\rangle \\ |\underline{2}\rangle &= |1.0\rangle \\ |\underline{3}\rangle &= |1.1\rangle \end{aligned}$$

Exemple.

Pour $n = 3$.

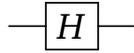
$$\begin{array}{ll} |\underline{0}\rangle = |0.0.0\rangle & |\underline{4}\rangle = |1.0.0\rangle \\ |\underline{1}\rangle = |0.0.1\rangle & |\underline{5}\rangle = |1.0.1\rangle \\ |\underline{2}\rangle = |0.1.0\rangle & |\underline{6}\rangle = |1.1.0\rangle \\ |\underline{3}\rangle = |0.1.1\rangle & |\underline{7}\rangle = |1.1.1\rangle \end{array}$$

3. Transformation de Hadamard

3.1. Définition

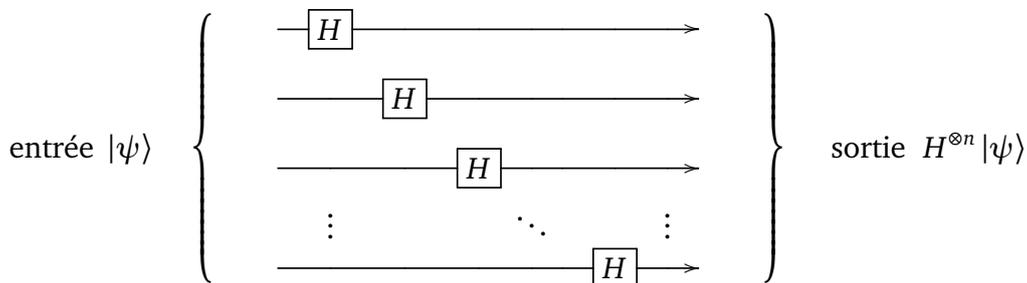
On rappelle que la porte de Hadamard est définie pour les 1-qubits par la formule :

$$H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \quad \text{et} \quad H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle).$$



La **transformation de Hadamard** d'un n -qubit $|\psi\rangle$ est l'application d'une porte de Hadamard sur chacun des 1-qubits le constituant. On note cette transformation $H^{\otimes n}$.

Le circuit est simplement composé de n lignes, avec une porte de Hadamard par ligne (l'ordre de ces portes n'a pas d'importance).



Exemple.

Pour $n = 2$.

$$\begin{aligned} |0\rangle &= |0.0\rangle \xrightarrow{H^{\otimes 2}} \frac{1}{2}(|0+1\rangle|0+1\rangle) = \frac{1}{2}(|0.0\rangle + |0.1\rangle + |1.0\rangle + |1.1\rangle) = \frac{1}{2}(|\underline{0}\rangle + |\underline{1}\rangle + |\underline{2}\rangle + |\underline{3}\rangle) \\ |1\rangle &= |0.1\rangle \xrightarrow{H^{\otimes 2}} \frac{1}{2}(|0+1\rangle|0-1\rangle) = \frac{1}{2}(|0.0\rangle - |0.1\rangle + |1.0\rangle - |1.1\rangle) = \frac{1}{2}(|\underline{0}\rangle - |\underline{1}\rangle + |\underline{2}\rangle - |\underline{3}\rangle) \\ |2\rangle &= |1.0\rangle \xrightarrow{H^{\otimes 2}} \frac{1}{2}(|0-1\rangle|0+1\rangle) = \frac{1}{2}(|0.0\rangle + |0.1\rangle - |1.0\rangle - |1.1\rangle) = \frac{1}{2}(|\underline{0}\rangle + |\underline{1}\rangle - |\underline{2}\rangle - |\underline{3}\rangle) \\ |3\rangle &= |1.1\rangle \xrightarrow{H^{\otimes 2}} \frac{1}{2}(|0-1\rangle|0-1\rangle) = \frac{1}{2}(|0.0\rangle - |0.1\rangle - |1.0\rangle + |1.1\rangle) = \frac{1}{2}(|\underline{0}\rangle - |\underline{1}\rangle - |\underline{2}\rangle + |\underline{3}\rangle) \end{aligned}$$

3.2. Formule de la transformation de Hadamard

Quelle est la formule qui permet de calculer la transformation de Hadamard sur les qubits de base ?

Exemple.

Calculons $H^{\otimes n}|\underline{0}\rangle$ pour n quelconque. Comme $|\underline{0}\rangle = |0.0\dots 0\rangle$ alors $H^{\otimes n}|\underline{0}\rangle = \frac{1}{\sqrt{2^n}}|0+1\rangle \cdot |0+1\rangle \cdots |0+1\rangle$. En développant ce produit, on obtient toutes les combinaisons possibles de 0 et de 1 :

$$H^{\otimes n}|\underline{0}\rangle = \frac{1}{\sqrt{2^n}}(|0\dots 0.0\rangle + |0\dots 0.1\rangle + |0\dots 1.0\rangle + \cdots + |1\dots 1.1\rangle)$$

Autrement dit :

$$H^{\otimes n}|\underline{0}\rangle = \frac{1}{\sqrt{2^n}} \sum_{\ell=0}^{2^n-1} |\underline{\ell}\rangle$$

La formule générale est donnée par la proposition suivante.

Proposition 1.

Pour $0 \leq k \leq 2^n - 1$, on a :

$$H^{\otimes n} |\underline{k}\rangle = \frac{1}{\sqrt{2^n}} \sum_{\ell=0}^{2^n-1} (-1)^{\underline{k} \odot \underline{\ell}} |\underline{\ell}\rangle$$

Notation. Pour l'écriture binaire $\underline{k} = k_1.k_2\dots k_n$ et l'écriture binaire $\underline{\ell} = \ell_1.\ell_2\dots \ell_n$ (avec $k_i, \ell_i \in \{0, 1\}$) alors

$$\underline{k} \odot \underline{\ell} = k_1\ell_1 \oplus k_2\ell_2 \oplus \dots \oplus k_n\ell_n \in \{0, 1\}.$$

C'est comme un produit scalaire modulo 2.

Par exemple si $|\underline{k}\rangle = |0.1.0.1.1\rangle$ et $|\underline{\ell}\rangle = |1.1.0.0.1\rangle$ alors

$$\underline{k} \odot \underline{\ell} = 0 \cdot 1 \oplus 1 \cdot 1 \oplus 0 \cdot 0 \oplus 1 \cdot 0 \oplus 1 \cdot 1 = 0 \oplus 1 \oplus 0 \oplus 0 \oplus 1 = 0.$$

Démonstration. $H^{\otimes n} |\underline{k}\rangle$ est un produit de termes ayant la forme $|0+1\rangle$ ou $|0-1\rangle$. En développant ce produit on obtient une expression faisant intervenir tous les n -qubits de la base canonique, donc tous les $|\underline{\ell}\rangle$, avec $\ell = 0, \dots, 2^n - 1$.

On ne change le signe qu'en présence d'un 1 (donc il faut $\ell_i = 1$) et du signe « $-$ » (donc $k_i = 1$). Une preuve détaillée se fait par récurrence. \square

3.3. Exemple

Exemple.

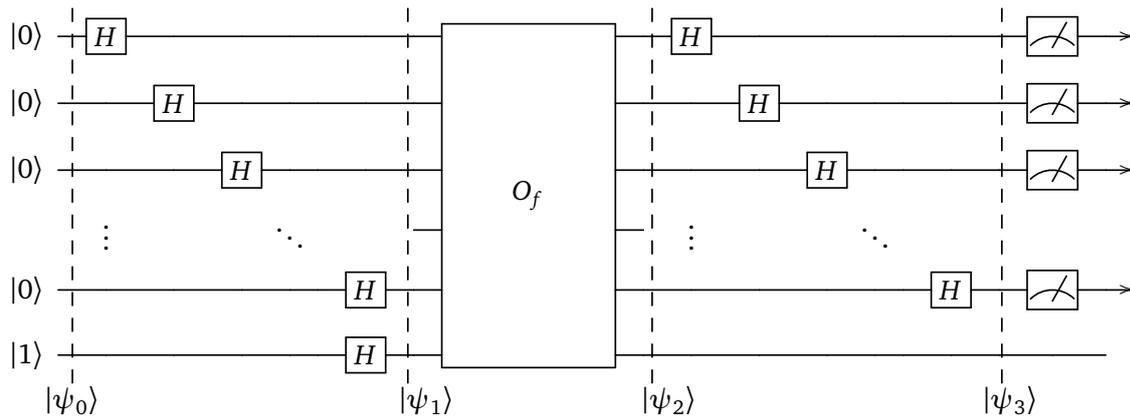
Soit $n = 3$ et $|\underline{k}\rangle = |\underline{5}\rangle = |1.0.1\rangle$, alors un calcul direct donne :

$$\begin{aligned} H^{\otimes 3} |\underline{5}\rangle &= H^{\otimes 3} |1.0.1\rangle \\ &= \frac{1}{2\sqrt{2}} |(0-1).(0+1).(0-1)\rangle \\ &= \frac{1}{2\sqrt{2}} (|0.0.0\rangle - |0.0.1\rangle + |0.1.0\rangle - |0.1.1\rangle - |1.0.0\rangle + |1.0.1\rangle - |1.1.0\rangle + |1.1.1\rangle) \end{aligned}$$

On retrouve bien la formule annoncée car avec $\underline{k} = \underline{5} = 1.0.1$ on a :

- pour $\ell = 0$: $\underline{\ell} = 0.0.0$, $\underline{k} \odot \underline{\ell} = 0$, donc terme $+ |0.0.0\rangle$,
- pour $\ell = 1$: $\underline{\ell} = 0.0.1$, $\underline{k} \odot \underline{\ell} = 1$, donc terme $- |0.0.1\rangle$,
- pour $\ell = 2$: $\underline{\ell} = 0.1.0$, $\underline{k} \odot \underline{\ell} = 0$, donc terme $+ |0.1.0\rangle$,
- pour $\ell = 3$: $\underline{\ell} = 0.1.1$, $\underline{k} \odot \underline{\ell} = 1$, donc terme $- |0.1.1\rangle$,
- pour $\ell = 4$: $\underline{\ell} = 1.0.0$, $\underline{k} \odot \underline{\ell} = 1$, donc terme $- |1.0.0\rangle$,
- pour $\ell = 5$: $\underline{\ell} = 1.0.1$, $\underline{k} \odot \underline{\ell} = 0$, donc terme $+ |1.0.1\rangle$,
- pour $\ell = 6$: $\underline{\ell} = 1.1.0$, $\underline{k} \odot \underline{\ell} = 1$, donc terme $- |1.1.0\rangle$,
- pour $\ell = 7$: $\underline{\ell} = 1.1.1$, $\underline{k} \odot \underline{\ell} = 0$, donc terme $+ |1.1.1\rangle$.

4. Preuve de l'algorithme de Deutsch–Jozsa



- Initialisation.

$$|\psi_0\rangle = |0 \dots 0\rangle \cdot |1\rangle = |\underline{0}\rangle \cdot |1\rangle.$$

On mélange les deux écritures : la notation entière qui regroupe les n premiers qubits et la notation classique pour le dernier qubit.

- Transformation de Hadamard.

$$\begin{aligned} |\psi_1\rangle &= H^{\otimes n+1} |\psi_0\rangle \\ &= H^{\otimes n} |\underline{0}\rangle \cdot H |1\rangle \\ &= \frac{1}{\sqrt{2^n}} \sum_{\ell=0}^{2^n-1} |\underline{\ell}\rangle \cdot \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle) \end{aligned}$$

- Oracle.

$$\begin{aligned} |\psi_2\rangle &= O_f |\psi_1\rangle \\ &= \frac{1}{\sqrt{2^n}} \sum_{\ell=0}^{2^n-1} |\underline{\ell}\rangle \cdot \frac{1}{\sqrt{2}} (|0 \oplus f(\ell)\rangle - |1 \oplus f(\ell)\rangle) \\ &= \frac{1}{\sqrt{2^n}} \sum_{\ell=0}^{2^n-1} (-1)^{f(\ell)} |\underline{\ell}\rangle \cdot \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle) \end{aligned}$$

On a utilisé :

$$|0 \oplus f(\ell)\rangle - |1 \oplus f(\ell)\rangle = \begin{cases} |0\rangle - |1\rangle & \text{si } f(\ell) = 0 \\ -(|0\rangle - |1\rangle) & \text{si } f(\ell) = 1 \end{cases} = (-1)^{f(\ell)} (|0\rangle - |1\rangle).$$

- Transformation de Hadamard.

$$\begin{aligned}
|\psi_3\rangle &= H^{\otimes n} \left(\frac{1}{\sqrt{2^n}} \sum_{\ell=0}^{2^n-1} (-1)^{f(\ell)} |\underline{\ell}\rangle \right) \cdot \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle) \\
&= \frac{1}{\sqrt{2^n}} \sum_{\ell=0}^{2^n-1} (-1)^{f(\ell)} H^{\otimes n} (|\underline{\ell}\rangle) \cdot \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle) \\
&= \frac{1}{\sqrt{2^n}} \sum_{\ell=0}^{2^n-1} (-1)^{f(\ell)} \left(\frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} (-1)^{\underline{\ell} \circ k} |\underline{k}\rangle \right) \cdot \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle) \\
&= \frac{1}{2^n} \sum_{k=0}^{2^n-1} |\underline{k}\rangle \sum_{\ell=0}^{2^n-1} (-1)^{f(\ell) + \underline{\ell} \circ k} \cdot \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle)
\end{aligned}$$

Quelle est la probabilité de mesurer $0.0 \dots 0$ pour les n premiers qubits ? Il s'agit de trouver le coefficient $\alpha \in \mathbb{C}$ devant le qubit $|0.0 \dots 0\rangle \cdot \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle)$ (le dernier qubit ne sera pas mesuré). Autrement dit il s'agit de trouver le coefficient correspondant à $\underline{k} = \underline{0}$:

$$\alpha = \frac{1}{2^n} \sum_{\ell=0}^{2^n-1} (-1)^{f(\ell) + \underline{\ell} \circ \underline{0}}.$$

Comme $\underline{\ell} \circ \underline{0} = 0$ alors :

$$\alpha = \frac{1}{2^n} \sum_{\ell=0}^{2^n-1} (-1)^{f(\ell)}.$$

- Si f est constante, par exemple $f(\underline{\ell}) = 0$, pour tout ℓ , alors :

$$\alpha = \frac{1}{2^n} \sum_{\ell=0}^{2^n-1} (-1)^0 = \frac{1}{2^n} \sum_{\ell=0}^{2^n-1} 1 = 1.$$

Comme le qubit $|\psi_3\rangle$ est normalisé, et que $\alpha = 1$ alors les autres coefficients des termes de $|\psi_3\rangle$ sont tous nuls. Ainsi dans ce cas $|\psi_3\rangle = |0.0 \dots 0\rangle \cdot \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle)$ et la probabilité de mesurer $0.0 \dots 0$ sur les n premiers qubits est 1.

De même si f était constante égale à 1, alors on trouverait $\alpha = -1$ et $|\psi_3\rangle = -|0.0 \dots 0\rangle \cdot \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle)$ et la probabilité de mesurer $0.0 \dots 0$ sur les n premiers qubits est également 1.

- Si f est équilibrée, il y a autant de ℓ avec $f(\underline{\ell}) = 0$ que de ℓ avec $f(\underline{\ell}) = 1$, ainsi il y a autant de ℓ avec $(-1)^{f(\underline{\ell})} = +1$ que $(-1)^{f(\underline{\ell})} = -1$. (On rappelle $(-1)^p = \pm 1$.) Ainsi la somme des $(-1)^{f(\underline{\ell})}$ est nulle et donc $\alpha = 0$. La probabilité de mesurer $0.0 \dots 0$ sur les n premiers qubits est donc 0.

Conclusion : si la mesure sur les n premiers qubits est $0.0 \dots 0$ alors la fonction f est constante, sinon c'est qu'elle est équilibrée.

Algorithme de Grover

Vidéo ■ partie 11. Algorithme de Grover

L'algorithme de Grover est un algorithme de recherche d'un élément dans une liste qui est plus efficace que les algorithmes classiques. Son principe est simple, même si sa mise en œuvre est un peu complexe. L'algorithme de Grover ne fournit pas un résultat sûr à 100 %, mais une réponse qui a de grandes chances d'être la bonne.

1. Recherche dans une liste

1.1. Idée de l'algorithme

Expliquons l'algorithme de Grover avec des dessins.

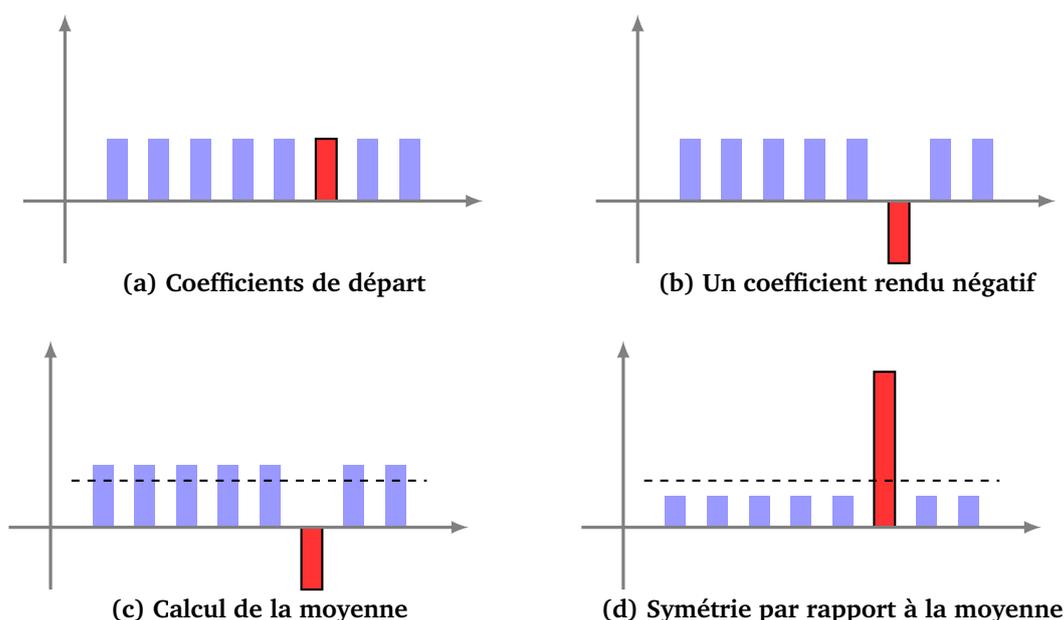


Figure (a). Il s'agit de distinguer un rang parmi les autres, ici le rang du rectangle rouge. On considère que les hauteurs des rectangles représentent les coefficients d'un qubit : ici il y a 8 coefficients pour l'expression d'un 3-qubit comme somme des 8 états de base. Une mesure de ce qubit ne donnerait aucune information, chacun des rangs s'obtenant avec la même probabilité, car les hauteurs des rectangles sont égales.

Figure (b). On rend le coefficient du rang qui nous intéresse négatif. (Cela peut se faire sans connaître le rang : je regarde la couleur du rectangle, s'il est rouge je change le signe). Une mesure de nouveau qubit ne donnerait toujours aucune information, car en valeur absolue les hauteurs des rectangles sont encore égales.

Figure (c). On calcule la moyenne des coefficients

Figure (d). On effectue une symétrie par rapport à la moyenne. Les rectangles bleus ont maintenant des hauteurs petites alors que le rectangle rouge a une grande hauteur. Que donne une mesure de ce nouveau qubit ? Il y a beaucoup plus de chances d'obtenir l'état de base correspondant au rectangle rouge et donc d'obtenir le rang souhaité.

L'algorithme de Grover est l'itération de ce procédé : à partir du dernier état obtenu avant mesure, on recommence les étapes (b), (c) et (d). Le rectangle rouge devient de plus en plus grand et les rectangles bleus de plus en plus petits. Ainsi après plusieurs itérations, une mesure donne avec une très forte probabilité, le rang du rectangle rouge.

1.2. Recherche dans une liste ordonnée

On dispose d'une liste et on nous donne un élément. Il s'agit de trouver s'il est présent dans la liste et de déterminer son rang. Le problème est donc : trouver i tel que $\text{liste}[i] = \text{mon-élément}$.

Supposons d'abord que les éléments sont classés par ordre (par exemple les mots d'un dictionnaire par ordre alphabétique, ou bien les numéros de cartes d'étudiants classés du plus petit au plus grand). Alors un algorithme de recherche classique est la recherche par dichotomie (on coupe au milieu, on regarde si l'élément cherché est avant ou après et on recommence). C'est une méthode très efficace : si la liste comporte N éléments alors la complexité est $O(\ln_2(N))$. La complexité est mesurée comme le nombre de comparaisons entre l'élément au rang i et l'élément recherché. Par exemple si $N = 1024 = 2^{10}$, alors il faut environ moins de 10 comparaisons pour trouver l'élément et si la liste contient un milliard de données, il faut moins de 30 comparaisons pour conclure !

1.3. Recherche dans une liste non ordonnée

Pour certaines listes, il n'est pas possible d'ordonner des éléments ou bien on ne souhaite pas le faire car ordonner une liste est assez long. Comment chercher un élément dans une liste non ordonnée ? Il n'y a pas d'autre choix que de parcourir la liste ! On peut par exemple parcourir la liste en partant du premier élément, ou bien en choisissant les éléments au hasard. Dans les deux méthodes, la complexité dans le pire des cas est N (si l'élément cherché est le dernier à être testé). En moyenne, on trouvera l'élément cherché au bout de $\frac{N}{2}$ tests, mais cela n'améliore pas l'ordre de grandeur de la complexité qui est donc $O(N)$ (car $O(N) = O(\frac{N}{2})$).

1.4. Complexité de l'algorithme de Grover

L'algorithme de Grover qui sera étudié dans ce chapitre a une complexité d'ordre $O(\sqrt{N})$, c'est donc un gros progrès par rapport aux algorithmes classiques. Par exemple, dans une liste de $N = 1024$ personnes, il suffira d'environ 30 tests ; pour une liste d'un milliard de données, la complexité est d'environ 30 000. C'est évidemment beaucoup plus que l'algorithme de la dichotomie sur une liste ordonnée, mais beaucoup moins que la recherche séquentielle qui est de complexité N .

1.5. Algorithmes probabilistes

L'algorithme de Grover est rapide mais ne renvoie malheureusement pas toujours le bon résultat ! C'est un algorithme probabiliste. L'algorithme de Grover renvoie le bon résultat dans la plupart des cas (on verra que pour une liste de longueur N , l'algorithme se trompe avec une probabilité inférieure à $\frac{4}{N}$).

Pourquoi un algorithme probabiliste, ne donnant donc pas toujours la réponse attendue, peut quand même être un bon algorithme ? Tout d'abord, pour certains problèmes, ne pas avoir la bonne réponse n'est pas trop grave. Par exemple, si un algorithme vous fournit le plus court chemin dans 99 cas sur 100 et que vous faites quelques kilomètres en plus de temps en temps, cela peut vous convenir. D'autre part il est souvent

facile de vérifier si la réponse donnée est correcte, donc si la réponse obtenue ne vous convient pas, vous relancez l'algorithme. Même un algorithme qui ne donne la bonne réponse qu'une fois sur deux peut être utile ! Imaginez un algorithme qui donne tous les bons numéros du loto seulement une fois sur deux, est-ce que cela vous intéresserait ?

1.6. Application au hachage

Certaines sécurités informatiques sont basées sur des fonctions de hachage. Par exemple une fonction de hachage permet de vérifier qu'un fichier téléchargé n'a pas été compromis (*checksum*). D'autres exemples sont les *bitcoins* qui utilisent une « preuve de travail », de même que certaines méthodes de cryptographie (par exemple pour ne pas sauvegarder vos mots de passe en clair sur votre disque dur).

Considérons l'exemple d'une fonction de hachage qui à un entier k codé sur n bits (la clé) associe un entier $h(k)$ (le *hash*). J'utilise cette fonction ainsi :

- je choisis une clé secrète par exemple $k_0 = 1.0.1.0.1$ (avec ici $n = 5$),
- je calcule $h(k_0)$, par exemple $h(k_0) = 12\,575\,302$,
- $h(k_0)$ est mon mot de passe.

Imaginons un pirate qui voudrait attaquer mon compte. Il n'a pas d'autre choix que de tester toutes les clés possibles $0.0.0.0.0$, puis $0.0.0.0.1, \dots$ afin d'obtenir la bonne clé, donc le bon mot de passe. Il y a en tout $N = 2^n$ (ici $n = 5$) clés possibles à tester dans le pire des cas avec l'informatique classique.

Mais trouver cette clé revient à trouver le bon élément parmi une liste de $N = 2^n$ éléments, ce que fait l'algorithme de Grover avec une complexité d'ordre $O(\sqrt{N})$, c'est-à-dire $O(2^{n/2})$, ce qui est beaucoup plus rapide que la solution classique.

À la suite de la découverte de Grover, il a été recommandé de doubler la longueur des clés de certains protocoles (par exemple passer de AES-128 à AES-256). En effet imaginons une clé de longueur n bits, un algorithme classique nécessite de l'ordre de $N = 2^n$ tests et l'algorithme quantique seulement $\sqrt{N} = 2^{n/2}$. Si la clé est doublée à une longueur $2n$, alors l'algorithme de Grover nécessite maintenant $\sqrt{2^{2n}} = 2^n$ tests et donc le niveau de sécurité initial est maintenu.

1.7. Image réciproque

Élargissons la situation précédente à un problème plus général. Soit $f : E \rightarrow F$ une fonction. Étant donné $x \in E$, il est généralement facile de calculer son image $y = f(x)$. Par contre le problème inverse de trouver un antécédent est souvent délicat : étant donné $y \in F$, trouver $x \in E$ tel que $y = f(x)$. Parfois la seule solution est de tester tous les x possibles, et là encore l'algorithme de Grover permet de le faire plus rapidement.

Voici un exemple de fonction où il est difficile de calculer un antécédent. Soit p un (grand) nombre premier et $f : \mathbb{Z} \rightarrow \mathbb{Z}$ la fonction définie par $f(x) = x^2 \pmod{p}$. Bien sûr, pour x donné, il est facile de calculer $y = f(x)$. Par contre y étant fixé, il est difficile de trouver un antécédent, c'est-à-dire un x tel que $x^2 \pmod{p} = y$. On doit alors se rabattre sur des techniques de force brute et tester $x = 0, x = 1, \dots, x = p - 1$.

2. Principe et circuit

2.1. Problème

Nous modélisons la recherche dans une liste non ordonnée à l'aide d'une fonction mathématique. Soit N un entier fixé et soit k_0 un entier avec $0 \leq k_0 \leq N - 1$. Définissons alors la fonction $f : \{0, 1, \dots, N - 1\} \rightarrow \{0, 1\}$ par

$$f(k_0) = 1 \quad \text{et} \quad f(k) = 0 \quad \text{pour tout } k \neq k_0.$$

Problème. Étant donnée une telle fonction f , trouver la valeur k_0 telle que $f(k_0) = 1$.

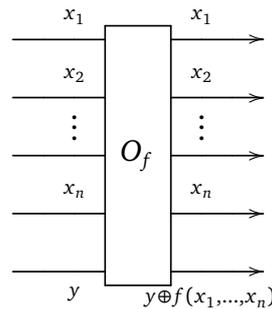
Remarques.

- Il s'agit donc de trouver l'antécédent de 1 par f .
- On peut identifier $\{0, 1, \dots, N - 1\}$ à $\mathbb{Z}/N\mathbb{Z}$ et $\{0, 1\}$ à $\mathbb{Z}/2\mathbb{Z}$ et donc considérer $f : \mathbb{Z}/N\mathbb{Z} \rightarrow \mathbb{Z}/2\mathbb{Z}$.

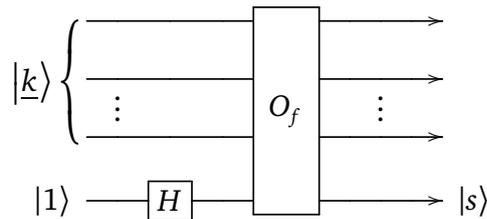
2.2. Oracle

On se place dans le cas où N est une puissance de 2 : $N = 2^n$. On rappelle que pour $0 \leq k \leq N - 1$, alors \underline{k} est l'écriture binaire de k sur n bits. Ainsi $|\underline{k}\rangle$, pour $k = 0, \dots, 2^n - 1$, désigne les n -qubits de la base canonique : $|\underline{0}\rangle = |0.0 \dots 0\rangle$, $|\underline{1}\rangle = |0.0 \dots 1\rangle$, ..., $|\underline{2^n - 1}\rangle = |1.1 \dots 1\rangle$.

Nous allons utiliser l'oracle O_f associé à la fonction f . Pour $x \in \mathbb{Z}/N\mathbb{Z}$ et $y \in \mathbb{Z}/2\mathbb{Z}$, l'oracle réalise une fonction $F(x, y) = (x, y \oplus f(x))$. On préfère écrire l'entier x à l'aide de son écriture binaire $\underline{x} = x_1.x_2 \dots x_n$, ce qui permet de récrire la fonction F sous la forme $F(x, y) = (x_1, \dots, x_n, y \oplus f(x_1, \dots, x_n))$. On rappelle que $y \oplus y'$ est l'addition dans $\mathbb{Z}/2\mathbb{Z}$ (qui vérifie $1 \oplus 1 = 0$).



Utilisons notre oracle pour réaliser le petit circuit suivant :



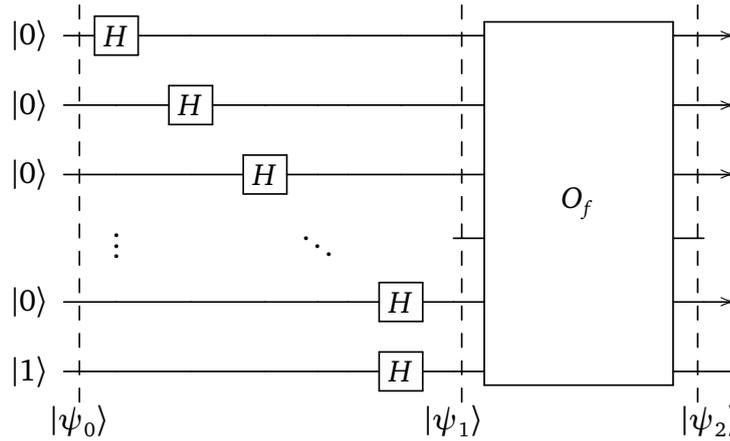
Que se passe-t-il pour notre f particulier qui vaut 1 seulement en k_0 ? L'entrée sur la dernière ligne avant l'oracle est $\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$ et la sortie est

$$s = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \oplus f(k) = (-1)^{f(k)} \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = \begin{cases} \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) & \text{si } k \neq k_0 \\ -\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) & \text{si } k = k_0 \end{cases}$$

Ainsi l'oracle permet de détecter si l'entier en entrée est k_0 ou pas. Malheureusement nous n'avons fait aucun progrès car il faut de nouveau tester tous les k de 0 à $N - 1$ pour pouvoir conclure. C'est là qu'entre en jeu la magie de l'informatique quantique et la superposition des états : il est possible de tester toutes ces valeurs en même temps !

2.3. Circuit

Voici le début du circuit de l'algorithme de Grover qui permet de comprendre l'essentiel de son fonctionnement (le circuit complet sera étudié plus tard).



- Initialisation. Le qubit en entrée est le $(n + 1)$ -qubit :

$$|\psi_0\rangle = |0 \dots 0\rangle \cdot |1\rangle = |\underline{0}\rangle \cdot |1\rangle.$$

- Transformation de Hadamard. On s'intéresse d'abord seulement aux n premières lignes. Après la transformation de Hadamard (une porte de Hadamard sur chacune des n premières lignes) alors le n -qubit est

$$|\underline{0}\rangle + |\underline{1}\rangle + \dots + |\underline{k_0}\rangle + \dots + |\underline{2^n - 1}\rangle$$

(à un facteur multiplicatif près). Ainsi tous les qubits $|\underline{k}\rangle$ se retrouvent simultanément en entrée de l'oracle !

Voici les calculs complets, en intégrant tous les qubits :

$$\begin{aligned} |\psi_1\rangle &= H^{\otimes n+1} |\psi_0\rangle \\ &= H^{\otimes n} |\underline{0}\rangle \cdot H |1\rangle \\ &= \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} |\underline{k}\rangle \cdot \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle) \end{aligned}$$

- Oracle. Nous avons vu que l'oracle fait apparaître un signe « - » devant le terme correspond à k_0 . Ainsi la sortie de l'oracle est

$$|\underline{0}\rangle + |\underline{1}\rangle + \dots - |\underline{k_0}\rangle + \dots + |\underline{2^n - 1}\rangle$$

Noter le signe « - » devant $|\underline{k_0}\rangle$ uniquement.

En détail, sachant que $(-1)^{f(k)} = 1$, sauf $(-1)^{f(k_0)} = -1$:

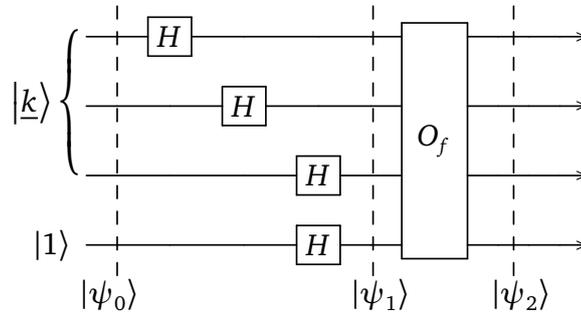
$$\begin{aligned} |\psi_2\rangle &= O_f |\psi_1\rangle \\ &= \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} (-1)^{f(k)} |\underline{k}\rangle \cdot \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle) \\ &= \frac{1}{\sqrt{2^n}} \left(|\underline{0}\rangle + |\underline{1}\rangle + \dots - |\underline{k_0}\rangle + \dots + |\underline{2^n - 1}\rangle \right) \cdot \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle) \end{aligned}$$

- Bilan. En une seule évaluation de l'oracle, on arrive à distinguer le terme de rang k_0 des autres termes. L'idée essentielle est ici. Cependant on n'a pas complètement terminé : il reste à déterminer précisément ce rang, connaissant la somme. Ce sera le travail assez technique du reste de ce chapitre.

2.4. Exemple

Exemple.

Prenons $n = 3$ et $k_0 = 5$ qui caractérisent la fonction $f : \mathbb{Z}/8\mathbb{Z} \rightarrow \mathbb{Z}/2\mathbb{Z}$ définie par $f(k) = 0$ pour tout $k \neq 5$ et $f(5) = 1$.



Reprenons les calculs dans ce cas :

- Initialisation. $|\psi_0\rangle = |0.0.0\rangle \cdot |1\rangle = |0\rangle \cdot |1\rangle$.
- Transformation de Hadamard.

$$\begin{aligned} |\psi_1\rangle &= \frac{1}{4} |(0+1).(0+1).(0+1)|0-1\rangle \\ &= \frac{1}{4} (|0.0.0\rangle + |0.0.1\rangle + |0.1.0\rangle + |0.1.1\rangle + |1.0.0\rangle + |1.0.1\rangle + |1.1.0\rangle + |1.1.1\rangle) |0-1\rangle \\ &= \frac{1}{4} (|0\rangle + |1\rangle + |2\rangle + |3\rangle + |4\rangle + |5\rangle + |6\rangle + |7\rangle) |0-1\rangle \end{aligned}$$

- Oracle. Comme $f(5) = 1$ alors nous avons vu que l'oracle fait apparaître un signe « - » devant le terme correspond à $|5\rangle = |1.0.1\rangle$.

$$\begin{aligned} |\psi_2\rangle &= \frac{1}{4} (|0.0.0\rangle + |0.0.1\rangle + |0.1.0\rangle + |0.1.1\rangle + |1.0.0\rangle - |1.0.1\rangle + |1.1.0\rangle + |1.1.1\rangle) |0-1\rangle \\ &= \frac{1}{4} (|0\rangle + |1\rangle + |2\rangle + |3\rangle + |4\rangle - |5\rangle + |6\rangle + |7\rangle) |0-1\rangle \end{aligned}$$

- Bilan. On trouve bien une somme des qubits de base avec un signe « - » au rang $k_0 = 5$ (on commence à compter au rang 0).

3. Transformations géométriques

Le reste du chapitre est dédié à la détection du rang k_0 . C'est une partie assez technique, mais on comprend mieux les calculs à l'aide d'une interprétation géométrique un peu plus sophistiquée que celle de l'introduction de ce chapitre.

3.1. Symétrie de l'oracle

Que fait l'oracle sur les n -qubits des n premières lignes ? L'oracle change $|k_0\rangle$ en $-|k_0\rangle$, et laisse inchangé $|k\rangle$ pour $k \neq k_0$:

$$\begin{cases} |k_0\rangle \xrightarrow{O_f} -|k_0\rangle \\ |k\rangle \xrightarrow{O_f} |k\rangle \quad \text{si } k \neq k_0. \end{cases}$$

On va isoler le qubit de base $|k_0\rangle$ d'un côté et regrouper tous les autres qubits de base, ainsi n'importe quel

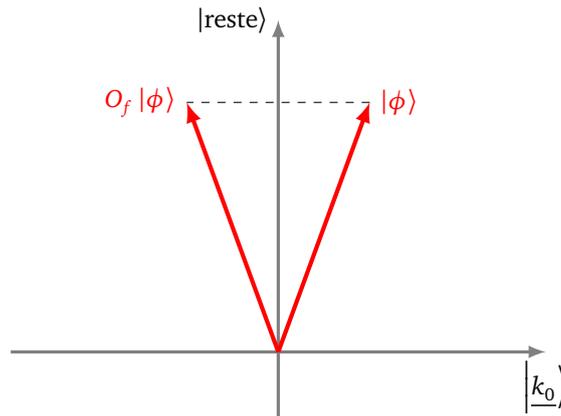
n -qubit $|\phi\rangle$ s'écrit :

$$|\phi\rangle = \alpha |k_0\rangle + \sum_{k \neq k_0} \alpha_k |k\rangle$$

L'action de l'oracle O_f donne :

$$O_f |\phi\rangle = -\alpha |k_0\rangle + \sum_{k \neq k_0} \alpha_k |k\rangle$$

Géométriquement cette transformation est une symétrie par rapport à l'axe formé des qubits de base autres que $|k_0\rangle$ que l'on regroupe schématiquement par l'axe $|\text{reste}\rangle$ ci-dessous. Sur la figure, la transformation est représentée comme une symétrie par rapport à une droite (mais en réalité c'est une symétrie par rapport à un hyperplan de dimension $2^n - 1$).



3.2. Symétrie S_0

Considérons la transformation S_0 définie sur les n -qubits de la base canonique par :

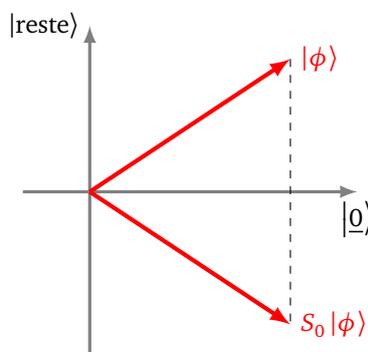
$$\begin{cases} |0\rangle \xrightarrow{S_0} |0\rangle \\ |k\rangle \xrightarrow{S_0} -|k\rangle \quad \text{si } k \neq 0 \end{cases}$$

Cette fois c'est seulement le qubit de base $|0\rangle$ qui reste inchangé.

Par exemple : pour $n = 2$, on a $S_0 |0.0\rangle = |0.0\rangle$ alors que $S_0 |0.1\rangle = -|0.1\rangle$, $S_0 |1.0\rangle = -|1.0\rangle$, $S_0 |1.1\rangle = -|1.1\rangle$. Comme d'habitude on étend S_0 par linéarité à tous les n -qubits. Ainsi :

$$S_0(\alpha |0.0\rangle + \beta |0.1\rangle + \gamma |1.0\rangle + \delta |1.1\rangle) = \alpha |0.0\rangle - \beta |0.1\rangle - \gamma |1.0\rangle - \delta |1.1\rangle.$$

Géométriquement S_0 est une symétrie par rapport à l'axe $|0\rangle$ (attention les axes ne sont pas les mêmes que dans la figure précédente).



Voici l'écriture algébrique de S_0 .

Lemme 1.

$$S_0 = 2|\underline{0}\rangle\langle\underline{0}| - I$$

I désigne l'application identité. Ainsi cette formule signifie que pour un qubit $|\phi\rangle$ on a :

$$S_0|\phi\rangle = 2|\underline{0}\rangle\langle\underline{0}|\phi\rangle - |\phi\rangle$$

L'écriture $|\underline{0}\rangle\langle\underline{0}|\phi\rangle$ est bien qubit car $\langle\underline{0}|\phi\rangle$ est un scalaire (i.e. un nombre complexe).

Démonstration. Il suffit de vérifier que cette formule est vraie pour les $|\phi\rangle$ parcourant les qubits de base. Pour $|\phi\rangle = |\underline{0}\rangle$, alors

$$(2|\underline{0}\rangle\langle\underline{0}| - I)|\underline{0}\rangle = 2|\underline{0}\rangle\langle\underline{0}|\underline{0}\rangle - |\underline{0}\rangle = 2|\underline{0}\rangle - |\underline{0}\rangle = |\underline{0}\rangle,$$

car $\langle\underline{0}|\underline{0}\rangle = 1$.

Pour $|\phi\rangle$ vérifiant $\langle\underline{0}|\phi\rangle = 0$, alors

$$(2|\underline{0}\rangle\langle\underline{0}| - I)|\phi\rangle = 2|\underline{0}\rangle\langle\underline{0}|\phi\rangle - |\phi\rangle = 2|\underline{0}\rangle \cdot 0 - |\phi\rangle = -|\phi\rangle.$$

□

Notons que la matrice de S_0 est une matrice diagonale, avec +1 comme premier élément et des -1 ailleurs.

$$S_0 = \begin{pmatrix} 1 & & & & \\ & -1 & & & \\ & & -1 & & \\ & & & \ddots & \\ & & & & -1 \end{pmatrix}$$

Il est clair que cette matrice est unitaire.

3.3. Transformation S_ψ

Nous allons généraliser la transformation S_0 . Fixons un n -qubit $|\psi\rangle$ de norme 1. Nous définissons la transformation S_ψ sur les n -qubits par la formule

$$S_\psi = 2|\psi\rangle\langle\psi| - I$$

Autrement dit, pour n'importe quel n -qubit $|\phi\rangle$, on a :

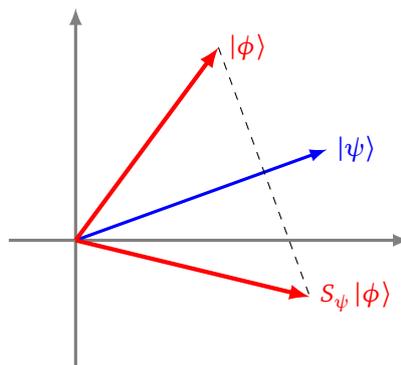
$$S_\psi|\phi\rangle = 2|\psi\rangle\langle\psi|\phi\rangle - |\phi\rangle.$$

Cette transformation vérifie :

$$\begin{cases} |\psi\rangle \xrightarrow{S_\psi} |\psi\rangle \\ |\phi\rangle \xrightarrow{S_\psi} -|\phi\rangle \quad \text{si } |\phi\rangle \text{ est orthogonal à } |\psi\rangle. \end{cases}$$

Rappelons que « $|\phi\rangle$ est orthogonal à $|\psi\rangle$ » signifie « $\langle\psi|\phi\rangle = 0$ ».

Géométriquement S_ψ est une symétrie par rapport à l'axe dirigé par $|\psi\rangle$.



3.4. Transformation S_{ψ_H}

Notons $|\psi_H\rangle$ le n -qubit formé par la somme de tous les qubits de la base canonique (normalisé de façon à avoir une norme 1) :

$$|\psi_H\rangle = \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} |k\rangle.$$

Ce qubit $|\psi_H\rangle$ est aussi l'image du qubit $|0\dots 0\rangle$ par la transformation de Hadamard :

$$|\psi_H\rangle = H^{\otimes n} |0\rangle.$$

Proposition 1.

La transformation S_{ψ_H} est définie par l'une des caractérisations équivalentes suivantes :

(i) $S_{\psi_H} = 2|\psi_H\rangle\langle\psi_H| - I$, c'est-à-dire $S_{\psi_H}|\phi\rangle = 2|\psi_H\rangle\langle\psi_H|\phi\rangle - |\phi\rangle$, pour tout qubit $|\phi\rangle$.

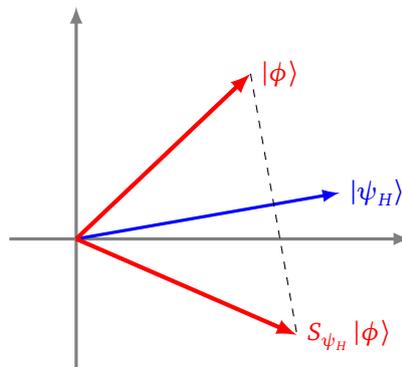
(ii) $\begin{cases} |\psi_H\rangle \xrightarrow{S_{\psi_H}} |\psi_H\rangle \\ |\phi\rangle \xrightarrow{S_{\psi_H}} -|\phi\rangle \quad \text{si } |\phi\rangle \text{ est orthogonal à } |\psi_H\rangle \end{cases}$

(iii) $S_{\psi_H} = H^{\otimes n} \cdot S_0 \cdot H^{\otimes n}$

(iv) S_{ψ_H} a pour matrice

$$\frac{2}{2^n}U - I \quad \text{où} \quad U = \begin{pmatrix} 1 & 1 & \dots & 1 \\ 1 & \ddots & & 1 \\ \vdots & & \ddots & \vdots \\ 1 & \dots & 1 & 1 \end{pmatrix} \in M_{2^n}$$

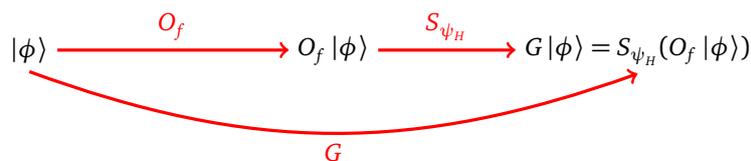
Nous prouverons cette proposition un peu plus loin. On retient que S_{ψ_H} est une symétrie par rapport à l'axe dirigé par $|\psi_H\rangle$.



3.5. Transformation de Grover

La *transformation de Grover* est l'application

$$G = S_{\psi_H} \circ O_f.$$



Nous allons voir quelle est l'action géométrique de G sur les qubits. Reprenons le qubit $|\psi_H\rangle$ obtenu comme la somme de tous les qubits de base :

$$|\psi_H\rangle = \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} |k\rangle.$$

Dans cette somme nous mettons à part le qubit correspondant à l'indice k_0 , qui est le rang que l'on doit déterminer :

$$|\psi_H\rangle = \sqrt{\frac{N-1}{N}} |\chi\rangle + \frac{1}{\sqrt{N}} |k_0\rangle$$

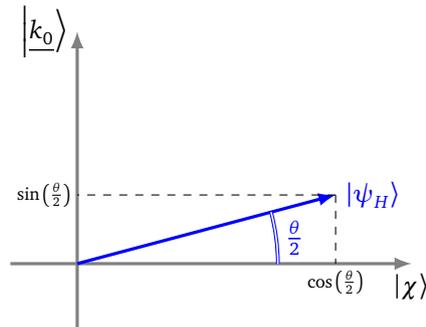
où

$$N = 2^n \quad \text{et} \quad |\chi\rangle = \frac{1}{\sqrt{N-1}} \sum_{k \neq k_0} |k\rangle.$$

Nous récrivons maintenant ψ_H à l'aide d'une écriture trigonométrique :

$$|\psi_H\rangle = \cos\left(\frac{\theta}{2}\right) |\chi\rangle + \sin\left(\frac{\theta}{2}\right) |k_0\rangle$$

où $\frac{\theta}{2}$ est l'angle entre $|\chi\rangle$ et $|\psi_H\rangle$.



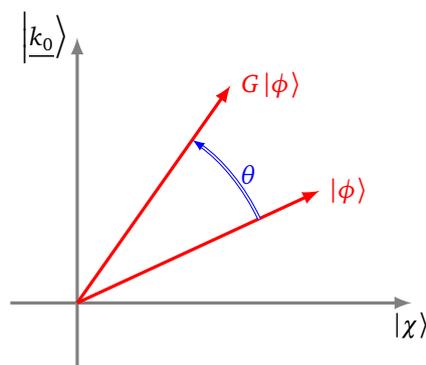
Comme dans la pratique $N = 2^n$ est grand, alors l'angle $\frac{\theta}{2}$ est petit. Pour plus de lisibilité le dessin ne reflète pas à quel point $\frac{\theta}{2}$ est petit. Ainsi $\frac{\theta}{2}$ est l'angle défini par :

$$\cos\left(\frac{\theta}{2}\right) = \sqrt{\frac{N-1}{N}} \quad \text{et} \quad \sin\left(\frac{\theta}{2}\right) = \frac{1}{\sqrt{N}}.$$

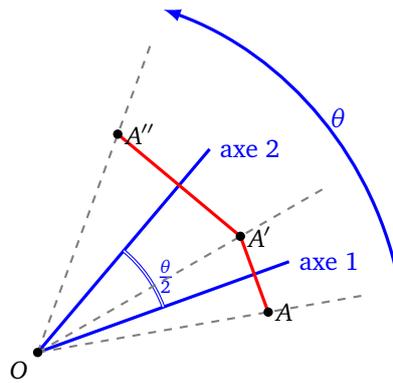
Proposition 2.

La transformation de Grover est une rotation d'angle θ (centrée à l'origine).

Autrement dit, pour tout qubit $|\phi\rangle$, $G|\phi\rangle$ est obtenu à partir de $|\phi\rangle$, par une rotation d'angle θ (encore une fois l'angle θ est en réalité beaucoup plus petit que sur le dessin ci-dessous).



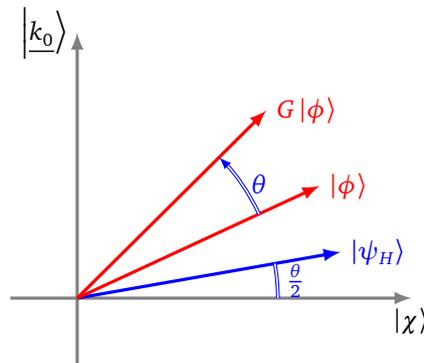
Démonstration. Un résultat géométrique dit que la composition de deux symétries axiales est une rotation, l'angle θ de cette rotation étant le double de l'angle entre les axes.



Ici G est la composition de deux symétries :

- la symétrie O_f d'axe $|\chi\rangle$,
- la symétrie S_{ψ_H} d'axe $|\psi_H\rangle$,
- l'angle entre $|\chi\rangle$ et $|\psi_H\rangle$ est $\frac{\theta}{2}$.

Ainsi $G = S_{\psi_H} \circ O_f$ est la rotation d'angle θ (centrée à l'origine).



□

3.6. Idée de l'algorithme

Le but de l'algorithme de Grover est de déterminer le rang k_0 . Ce rang est repérable après l'application de l'oracle O_f . En effet partons de

$$|\psi_H\rangle = \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} |k\rangle = \sqrt{\frac{N-1}{N}} |\chi\rangle + \frac{1}{\sqrt{N}} |k_0\rangle,$$

alors

$$O_f |\psi_H\rangle = \sqrt{\frac{N-1}{N}} |\chi\rangle - \frac{1}{\sqrt{N}} |k_0\rangle.$$

Mais attention ceci est une écriture quantique qui n'est pas mesurable. Souvenez-vous que nous n'avons pas accès aux coefficients d'un qubit.

Voici la seule certitude qu'une mesure puisse nous donner : si je sais par avance qu'un qubit $|\phi\rangle$ est un des qubits de base $|0\dots 0,0\rangle, |0\dots 0,1\rangle, \dots, |1\dots 1,1\rangle$, alors la mesure de ce n -qubit permet d'identifier ce qubit de base $|\phi\rangle$.

Prenons l'exemple des 1-qubits : si mon qubit $|\phi\rangle$ est $|0\rangle$ ou $|1\rangle$, alors une mesure permet d'identifier si on avait $|\phi\rangle = |0\rangle$ ou bien $|\phi\rangle = |1\rangle$. Noter que ceci ne fonctionnerait pas pour un état superposé $|\phi\rangle = \alpha|0\rangle + \beta|1\rangle$. De même avec un 2-qubit $|\phi\rangle$ parmi $|0,0\rangle, |0,1\rangle, |1,0\rangle, |1,1\rangle$ par une double mesure.

La transformation de Grover va nous permettre de transformer le qubit $|\psi_H\rangle$ (obtenu par transformation de Hadamard de $|0\dots 0,0\rangle$) en un qubit très proche du qubit de base $|k_0\rangle$. Il ne reste plus qu'à effectuer une mesure pour obtenir (presque à coup sûr) la valeur de k_0 .

Idée de l’algorithme de Grover.

- La transformation de Hadamard envoie l’état initial $|0.0 \dots 0\rangle$ sur $|\psi_H\rangle$.
- On part du qubit $|\psi_H\rangle$ qui est la superposition de tous les qubits de base.
- Ce qubit forme un angle $\frac{\theta}{2}$ avec l’axe $|\chi\rangle$. (L’angle $\frac{\theta}{2}$ est petit car $N = 2^n$ est grand.)
- La transformation de Grover est une rotation d’angle θ et conduit donc au qubit $G|\psi_H\rangle$ qui forme un angle $\frac{\theta}{2} + \theta$ avec l’axe $|\chi\rangle$.
- On itère la transformation de Grover jusqu’à obtenir un qubit $G^\ell |\psi_H\rangle$ qui forme un angle d’environ $\frac{\pi}{2}$ avec l’axe $|\chi\rangle$. (Ce nombre d’itérations ℓ est environ $\frac{\pi}{2\theta}$.)
- Le qubit $G^\ell |\psi_H\rangle$ obtenu est proche de $|k_0\rangle$.
- La mesure de ce qubit conduit très probablement à $\underline{k_0}$ (avec une probabilité d’erreur très petite, d’ordre $\frac{4}{N}$).

3.7. Portes quantiques

La transformation de Grover est la composition de l’oracle O_f et de la transformation S_{ψ_H} . La transformation de l’oracle est réalisable par un circuit quantique (voir le chapitre « Portes quantiques »). Nous montrons ici comment réaliser le circuit pour la transformation S_{ψ_H} , en nous limitant au cas des 2-qubits.

Porte Z. Tout d’abord rappelons l’action de la porte Z et sa matrice :

$$\text{---} \boxed{Z} \text{---} \quad \begin{cases} |0\rangle \mapsto |0\rangle \\ |1\rangle \mapsto -|1\rangle \end{cases} \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Porte CZ. La porte CZ (*Controlled Z*) fonctionne sur le même principe qu’une porte CNOT : si l’entrée de la première ligne est $|0\rangle$, alors la seconde ligne est inchangée, par contre si l’entrée de la première ligne est $|1\rangle$, alors on fait agir une porte Z sur la seconde ligne.

$$\begin{array}{c} \text{---} \bullet \text{---} \\ | \\ \text{---} \boxed{Z} \text{---} \end{array} \quad \begin{cases} |0.0\rangle \mapsto |0.0\rangle \\ |0.1\rangle \mapsto |0.1\rangle \\ |1.0\rangle \mapsto |1.0\rangle \\ |1.1\rangle \mapsto -|1.1\rangle \end{cases} \quad CZ = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$$

Circuit pour S_0 .

Voici un circuit qui permet de réaliser la transformation S_0 , dans le cas des 2-qubits. Noter bien que la partie droite du circuit est un porte CZ.

$$\begin{array}{c} \text{---} \boxed{Z} \text{---} \bullet \text{---} \\ | \\ \text{---} \boxed{Z} \text{---} \boxed{Z} \text{---} \end{array} \quad \begin{cases} |0.0\rangle \mapsto |0.0\rangle \\ |0.1\rangle \mapsto -|0.1\rangle \\ |1.0\rangle \mapsto -|1.0\rangle \\ |1.1\rangle \mapsto -|1.1\rangle \end{cases} \quad S_0 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$$

Circuit pour S_{ψ_H} .

On sait par la proposition 1 que $S_{\psi_H} = H^{\otimes n} \cdot S_0 \cdot H^{\otimes n}$, il suffit juste d’appliquer la transformation de Hadamard avant et après le circuit précédent.

$$\text{---} \boxed{S_{\psi_H}} \text{---} = \begin{array}{c} \text{---} \boxed{H} \text{---} \boxed{Z} \text{---} \bullet \text{---} \boxed{H} \text{---} \\ | \\ \text{---} \boxed{H} \text{---} \boxed{Z} \text{---} \boxed{Z} \text{---} \boxed{H} \text{---} \end{array}$$

3.8. Preuve autour de la transformation S_{ψ_H}

Cette section peut être passée lors d'une première lecture. Il s'agit de prouver la proposition 1 énoncée auparavant et que l'on rappelle ci-dessous.

Proposition 3.

La transformation S_{ψ_H} est définie par l'une des caractérisations équivalentes suivantes :

- (i) $S_{\psi_H} = 2|\psi_H\rangle\langle\psi_H| - I$, c'est-à-dire $S_{\psi_H}|\phi\rangle = 2|\psi_H\rangle\langle\psi_H|\phi\rangle - |\phi\rangle$, pour tout qubit $|\phi\rangle$.
- (ii)
$$\begin{cases} |\psi_H\rangle \xrightarrow{S_{\psi_H}} |\psi_H\rangle \\ |\phi\rangle \xrightarrow{S_{\psi_H}} -|\phi\rangle \end{cases} \quad \text{si } |\phi\rangle \text{ est orthogonal à } |\psi_H\rangle$$
- (iii) $S_{\psi_H} = H^{\otimes n} \cdot S_0 \cdot H^{\otimes n}$
- (iv) S_{ψ_H} a pour matrice

$$\frac{2}{2^n}U - I \quad \text{où} \quad U = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ 1 & \ddots & & 1 \\ \vdots & & \ddots & \vdots \\ 1 & \cdots & 1 & 1 \end{pmatrix} \in M_{2^n}$$

Démonstration. Définissons S_{ψ_H} par $S_{\psi_H} = 2|\psi_H\rangle\langle\psi_H| - I$. On rappelle que $|\psi_H\rangle$ est de norme 1.

- Preuve que (i) \iff (ii).

On a $S_{\psi_H}|\psi_H\rangle = 2|\psi_H\rangle\langle\psi_H|\psi_H\rangle - |\psi_H\rangle = 2|\psi_H\rangle - |\psi_H\rangle = |\psi_H\rangle$ et $S_{\psi_H}|\phi\rangle = 2|\psi_H\rangle\langle\psi_H|\phi\rangle - |\phi\rangle = 2|\psi_H\rangle \cdot 0 - |\phi\rangle = -|\phi\rangle$ pour tout $|\phi\rangle$ orthogonal à $|\psi_H\rangle$, c'est-à-dire $\langle\psi_H|\phi\rangle = 0$.

Réciproquement si on complète le vecteur $|\psi_H\rangle$ en une base orthogonale, alors la relation (ii) définit une unique application linéaire, qui est donc S_{ψ_H} .

- Preuve que (ii) \iff (iii).

Notons $T = H^{\otimes n} \cdot S_0 \cdot H^{\otimes n}$. On va montrer que $T = S_{\psi_H}$ en vérifiant que T et S_{ψ_H} vérifient les mêmes relations que celles vues en (ii).

On sait d'une part que $|\psi_H\rangle = H^{\otimes n}|\underline{0}\rangle$, mais $H^{\otimes n}$ est unitaire alors $(H^{\otimes n})^{-1} = (H^{\otimes n})^* = H^{\otimes n}$, d'où $|\underline{0}\rangle = H^{\otimes n}|\psi_H\rangle$. Ainsi :

$$\begin{aligned} T|\psi_H\rangle &= H^{\otimes n} \cdot S_0 \cdot H^{\otimes n}|\psi_H\rangle \\ &= H^{\otimes n} \cdot S_0|\underline{0}\rangle \\ &= H^{\otimes n}|\underline{0}\rangle \quad \text{car } S_0|\underline{0}\rangle = |\underline{0}\rangle \\ &= |\psi_H\rangle \end{aligned}$$

Considérons maintenant un qubit $|\phi\rangle$ orthogonal à $|\psi_H\rangle$, alors $\langle\psi_H|\phi\rangle = 0$ et comme $H^{\otimes n}$ est unitaire alors il préserve le produit scalaire, donc on a aussi $\langle H^{\otimes n}|\psi_H\rangle | H^{\otimes n}|\phi\rangle \rangle = 0$, donc $|\underline{0}\rangle$ et $H^{\otimes n}|\phi\rangle$ sont orthogonaux. Ainsi

$$\begin{aligned} T|\phi\rangle &= H^{\otimes n} \cdot S_0(H^{\otimes n}|\phi\rangle) \\ &= H^{\otimes n}(-H^{\otimes n}|\phi\rangle) \quad \text{car } |\underline{0}\rangle \text{ et } H^{\otimes n}|\phi\rangle \text{ sont orthogonaux, donc } S_0(H^{\otimes n}|\phi\rangle) = -H^{\otimes n}|\phi\rangle \\ &= -H^{\otimes n} \cdot H^{\otimes n}|\phi\rangle \\ &= -|\phi\rangle \quad \text{car } H^{\otimes n} \cdot H^{\otimes n} = \text{id parce que } H^{\otimes n} \text{ est unitaire} \end{aligned}$$

Conclusion : T et S_{ψ_H} vérifient la même relation vue en (ii). Les applications sont donc égales : $T = S_{\psi_H}$.

- Preuve que (i) \iff (iv). Par construction l'état $|\psi_H\rangle$ est la superposition de tous les états de la base canonique, donc il s'écrit sous forme de vecteur $\begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}$ à un facteur de normalisation près. Plus

précisément :

$$|\psi_H\rangle = \frac{1}{\sqrt{2^n}} \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} \quad \text{donc} \quad \langle \psi_H | = |\psi_H\rangle^* = \frac{1}{\sqrt{2^n}} (1 \quad 1 \quad \dots \quad 1)$$

Ainsi :

$$|\psi_H\rangle \langle \psi_H| = \frac{1}{\sqrt{2^n}} \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} \times \frac{1}{\sqrt{2^n}} (1 \quad 1 \quad \dots \quad 1) = \frac{1}{2^n} \begin{pmatrix} 1 & 1 & \dots & 1 \\ 1 & \ddots & & 1 \\ \vdots & & \ddots & \vdots \\ 1 & \dots & 1 & 1 \end{pmatrix} = \frac{1}{2^n} U$$

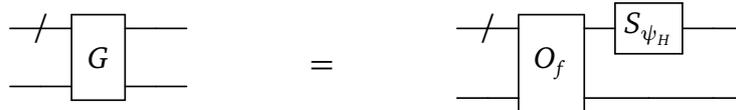
Conclusion : $S_{\psi_H} = 2|\psi_H\rangle \langle \psi_H| - I$ a pour matrice $\frac{2}{2^n}U - I$. (Réciproquement une matrice définit une unique application linéaire.)

□

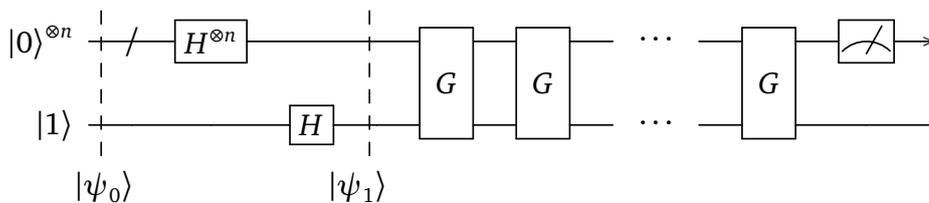
4. Étapes de l'algorithme de Grover

4.1. Circuit

On note G la transformation de Grover, elle prend en entrée un $(n + 1)$ -qubit, et est formée par la porte O_f de l'oracle, suivie d'une porte associée à la transformation S_{ψ_H} . On représente cette porte G avec 2 lignes seulement, la première ligne correspond à un n -qubit (ligne symbolisée avec « / »), la seconde à un 1-qubit.



Voici le circuit de l'algorithme de Grover.



La porte G est itérée ℓ fois avec $\ell \simeq \frac{\pi}{4} \sqrt{N}$ où $N = 2^n$. La complexité de l'algorithme est d'ordre ℓ , donc d'ordre $O(\sqrt{N})$. La mesure finale est la mesure d'un n -qubit (et correspond donc à n mesures de 1-qubits). Le circuit renvoie donc un n -bit classique \underline{k} avec $0 \leq k < 2^n$. Nous allons justifier que cet entier est très probablement le rang k_0 cherché.

4.2. Données

Soit $n \geq 1$ et $N = 2^n$. Supposons donné un entier k_0 vérifiant $0 \leq k_0 < N$. On considère la fonction $f : \mathbb{Z}/N\mathbb{Z} \rightarrow \mathbb{Z}/2\mathbb{Z}$, avec $f(k_0) = 1$ et $f(k) = 0$ pour tout $k \neq k_0$.

4.3. Initialisation et transformation de Hadamard

Le circuit quantique est initialisé par le $(n + 1)$ -qubit

$$|\psi_0\rangle = |0 \dots 0\rangle \cdot |1\rangle = |0\rangle \cdot |1\rangle.$$

Ensuite on applique la transformation de Hadamard pour obtenir le qubit

$$\begin{aligned}
|\psi_1\rangle &= H^{\otimes n+1} |\psi_0\rangle \\
&= H^{\otimes n} |0\rangle \cdot H |1\rangle \\
&= \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} |k\rangle \cdot \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle) \\
&= |\psi_H\rangle \cdot \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle)
\end{aligned}$$

Dans la suite on oublie le dernier qubit et on s'intéresse seulement au n -qubit formé par les n premières lignes.

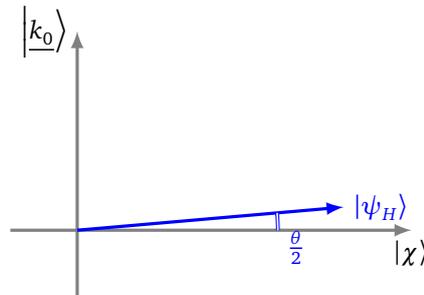
Dans $|\psi_H\rangle$ distinguons le qubit de base $|k_0\rangle$:

$$|\psi_H\rangle = \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} |k\rangle = \sqrt{\frac{N-1}{N}} |\chi\rangle + \frac{1}{\sqrt{N}} |k_0\rangle$$

que l'on récrit sous forme trigonométrique :

$$|\psi_H\rangle = \cos\left(\frac{\theta}{2}\right) |\chi\rangle + \sin\left(\frac{\theta}{2}\right) |k_0\rangle$$

où $\frac{\theta}{2}$ est l'angle entre $|\chi\rangle$ et $|\psi_H\rangle$, également défini par la relation $\sin\left(\frac{\theta}{2}\right) = \frac{1}{\sqrt{N}}$.



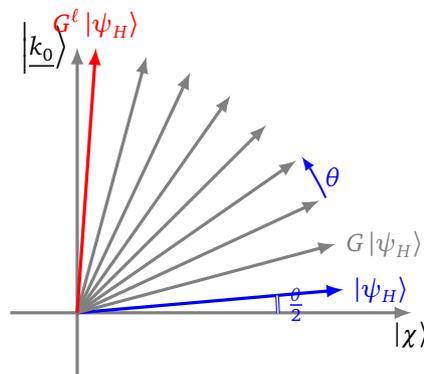
4.4. Itérations de la transformation de Grover

La transformation de Grover $G = S_{\psi_H} \circ O_f$, est une rotation d'angle θ . Donc après ℓ itérations on obtient le n -qubit

$$G^\ell |\psi_H\rangle = \cos(\theta_\ell) |\chi\rangle + \sin(\theta_\ell) |k_0\rangle$$

avec $\theta_\ell = \frac{\theta}{2} + \ell\theta$.

On veut $\theta_\ell \simeq \frac{\pi}{2}$, c'est-à-dire $\frac{\theta}{2} + \ell\theta \simeq \frac{\pi}{2}$. Ainsi ℓ est défini comme l'entier le plus proche de $\frac{\pi}{2\theta} - \frac{1}{2}$.



Donnons une approximation du nombre ℓ d'itérations nécessaires. Pour cela nous considérons que $N = 2^n$ est grand, et donc θ est petit. Comme $\sin\left(\frac{\theta}{2}\right) = \frac{1}{\sqrt{N}}$ alors $\frac{\theta}{2} \simeq \frac{1}{\sqrt{N}}$ (car pour x proche de 0, $\sin(x) \simeq x$). On veut $\ell\theta \simeq \frac{\pi}{2}$ donc $\ell \simeq \frac{\pi}{2\theta}$ et ainsi

$$\ell \simeq \frac{\pi}{4} \sqrt{N}$$

4.5. Mesure

Après ces ℓ itérations nous avons $\theta_\ell \simeq \frac{\pi}{2}$, donc

$$G^\ell |\psi_H\rangle = \cos(\theta_\ell) |\chi\rangle + \sin(\theta_\ell) |k_0\rangle \simeq |k_0\rangle.$$

La mesure de ce n -qubit conduit donc probablement au n -bit k_0 et permet alors d'identifier le rang k_0 . Les détails des probabilités sont donnés ci-dessous.

5. Probabilités

Nous avons construit un qubit $G^\ell |\psi_H\rangle$ qui est proche de $|k_0\rangle$. Ce qubit a donc de grandes chances d'être mesuré en k_0 et donc on retrouve le rang cherché k_0 , mais ce n'est pas une certitude. Avec quelle probabilité obtient-on le résultat correct ? Nous allons calculer cette probabilité d'obtenir le bon résultat.

Proposition 4.

L'algorithme de Grover renvoie le rang correct k_0 avec une probabilité supérieure à $1 - \frac{4}{N}$.

Ainsi la probabilité que l'algorithme renvoie un mauvais résultat est inférieure à $\frac{4}{N}$. Prenons par exemple $n = 10$, alors $N = \frac{1}{2^n} = 1024$ et l'algorithme fournit le résultat correct dans plus de 99,6% des cas.

Démonstration.

- Le qubit final obtenu par l'algorithme de Grover est

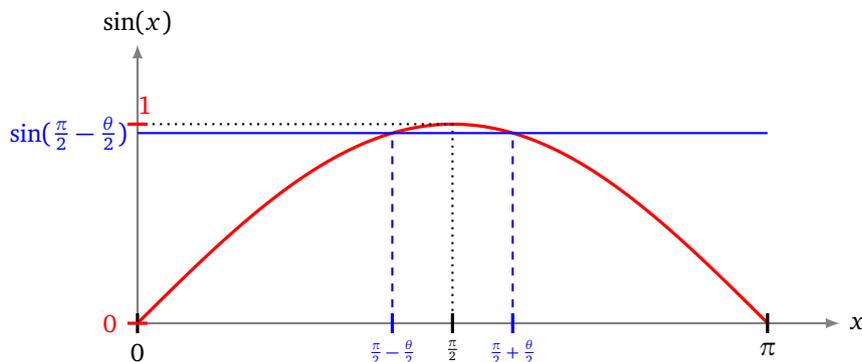
$$G^\ell |\psi_H\rangle = \cos(\theta_\ell) |\chi\rangle + \sin(\theta_\ell) |k_0\rangle.$$

Donc, lors de la mesure, la probabilité d'obtenir la bonne réponse k_0 est $p = |\sin(\theta_\ell)|^2$.

- Nous savons que la transformation de Grover G est une rotation d'angle θ et nous avons itéré cette transformation ℓ fois de façon à construire un angle $\theta_\ell = \frac{\theta}{2} + \ell\theta$ le plus proche possible de l'angle $\frac{\pi}{2}$. Ainsi l'angle θ_ℓ est dans un intervalle d'amplitude θ centré en $\frac{\pi}{2}$:

$$\frac{\pi}{2} - \frac{\theta}{2} < \theta_\ell \leq \frac{\pi}{2} + \frac{\theta}{2}.$$

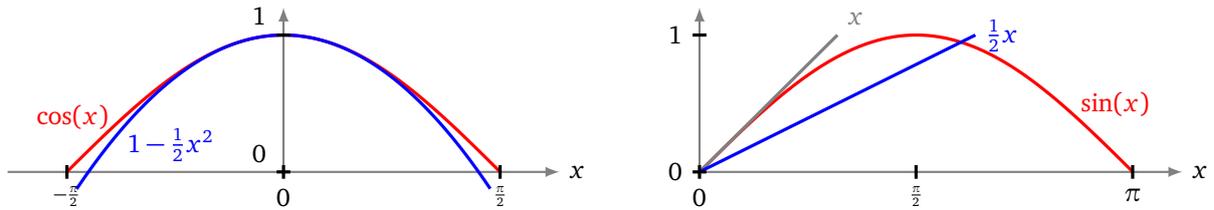
- Ainsi $\sin(\theta_\ell) \geq \sin\left(\frac{\pi}{2} - \frac{\theta}{2}\right)$ (voir la figure ci-dessous).



Donc :

$$\sin(\theta_\ell) \geq \sin\left(\frac{\pi}{2} - \frac{\theta}{2}\right) = \cos\left(\frac{\theta}{2}\right) \geq 1 - \frac{1}{2} \left(\frac{\theta}{2}\right)^2.$$

Pour la dernière inégalité on connaît le développement limité $\cos(x) \simeq 1 - \frac{x^2}{2}$ (pour x proche de 0), mais on a en plus l'inégalité $\cos(x) \geq 1 - \frac{x^2}{2}$ (figure de gauche ci-dessous).



- L'angle θ est défini avec la relation $\sin\left(\frac{\theta}{2}\right) = \frac{1}{\sqrt{N}}$. On sait que, pour x proche de 0, on a $\sin(x) \simeq x$, mais on a en plus l'inégalité $\sin(x) \geq \frac{x}{2}$ (figure de droite ci-dessus). Ainsi, comme $\sin\left(\frac{\theta}{2}\right) = \frac{1}{\sqrt{N}}$, alors $\frac{1}{\sqrt{N}} \geq \frac{\theta}{4}$ donc $\frac{4}{N} \geq \left(\frac{\theta}{2}\right)^2$ et alors en reprenant les inégalités ci-dessus :

$$\sin(\theta_\ell) \geq 1 - \frac{1}{2} \left(\frac{\theta}{2}\right)^2 \geq 1 - \frac{2}{N}.$$

Enfin on a $(1-x)^2 = 1 - 2x + x^2 \geq 1 - 2x$ quel que soit x , donc

$$p = |\sin(\theta_\ell)|^2 \geq \left(1 - \frac{2}{N}\right)^2 \geq 1 - \frac{4}{N}.$$

□

TROISIÈME PARTIE

$|1.0\rangle$

$|1.0\rangle$

ALGORITHME DE SHOR

Vidéo ■ partie 12.1. Éléments d'arithmétique

Vidéo ■ partie 12.2. Cryptographie RSA

La sécurité des communications sur internet est basée sur l'arithmétique et en particulier sur le système de cryptographie RSA qui repose sur la difficulté de factoriser de très grands entiers avec un ordinateur classique. Nous présentons dans ce chapitre les notions essentielles d'arithmétique afin de comprendre plus tard l'algorithme de Shor qui permet de factoriser rapidement un entier à l'aide d'un ordinateur quantique.

1. Division et pgcd

1.1. Divisibilité

Définition.

Soient $a, b \in \mathbb{Z}$ avec b non nul. On dit que b **divise** a s'il existe un entier $k \in \mathbb{Z}$ tel que $a = kb$.

On note alors $b|a$. On dit aussi que a est divisible par b ou encore que b est un diviseur de a .

Par exemple :

- $3|12$ (« 3 divise 12 » ou bien « 12 est divisible par 3 »).
- Plus généralement les diviseurs positifs de 12 sont 1, 2, 3, 4, 6, 12.
- Quel que soit $b \in \mathbb{Z}$, non nul, on a $b|0$.

1.2. Division euclidienne

La **division euclidienne** permet de généraliser la notion de divisibilité.

Soient $a \in \mathbb{Z}$ et $b \in \mathbb{N} \setminus \{0\}$. Il existe des entiers $q, r \in \mathbb{Z}$, uniques, tels que

$$a = bq + r \quad \text{et} \quad 0 \leq r < b$$

- L'entier q est le **quotient** et r est le **reste**.
- Exemple : $a = 101$, $b = 7$ alors $q = 14$ et $r = 3$ car $101 = 7 \times 14 + 3$.
- Le reste est nul si et seulement si b divise a .
- On note aussi (de façon un peu abusive) $r = a \pmod{b}$.
- Avec *Python* on calcule le quotient par $q = a // b$ (à noter la double barre de division) et le reste $r = a \% b$.

1.3. Pgcd

Soient $a, b \in \mathbb{Z}$ (non tous les deux nuls). Le **pgcd** de a et b est le plus grand entier qui divise à la fois a et b . Par exemple avec $a = 42$ et $b = 24$, les diviseurs positifs communs à a et b sont $\{1, 2, 3, 4, 6\}$, donc $\text{pgcd}(42, 24) = 6$.

1.4. Nombres premiers entre eux

Les entiers a et b sont **premiers entre eux** si leur pgcd vaut 1.

Par exemple $a = 20$ et $b = 33$ sont premiers entre eux, car le seul diviseur positif de ces deux entiers est 1. Autre exemple : deux entiers consécutifs sont toujours premiers entre eux. Preuve : si $d > 0$ divise a et $a + 1$ alors d divise $(a + 1) - a$, donc d divise 1, donc d égal 1.

1.5. Théorème de Bézout

Théorème 1 (Théorème de Bézout).

Soient a, b des entiers non nuls. Il existe des entiers $u, v \in \mathbb{Z}$ tels que

$$au + bv = \text{pgcd}(a, b)$$

On a même une équivalence lorsque les entiers sont premiers entre eux :

Corollaire 1.

Soient a, b deux entiers non nuls. a et b sont premiers entre eux si et seulement si il existe $u, v \in \mathbb{Z}$ tels que

$$au + bv = 1$$

Exemples :

- $a = 42$ et $b = 24$. On a vu $\text{pgcd}(42, 24) = 6$. Avec $u = -1$ et $v = 2$, on obtient $42 \times (-1) + 24 \times 2 = 6$.
- $a = 20$ et $b = 33$. On a vu $\text{pgcd}(20, 33) = 1$. Avec $u = 5$ et $v = -3$, on obtient $20 \times 5 + 33 \times (-3) = 1$.

1.6. Algorithme d'Euclide

L'algorithme d'Euclide est une méthode efficace pour calculer le pgcd et sa version étendue permet de trouver des coefficients u, v du théorème de Bézout.

Soient $a, b \in \mathbb{N}^*$. Considérons la division euclidienne $a = bq + r$, où r est le reste. Alors

$$\text{pgcd}(a, b) = \text{pgcd}(b, r)$$

Pour en faire un algorithme on calcule des divisions euclidiennes successives. Le pgcd sera le dernier reste non nul car on sait que $\text{pgcd}(a, 0) = a$.

Exemple.

Calculons le pgcd d de $a = 11\,466$ et $b = 1\,656$.

- Division euclidienne de a par b : $11\,466 = 6 \times 1\,656 + 1\,530$, donc le reste est $r_1 = 1\,530$. On utilise alors $d = \text{pgcd}(a, b) = \text{pgcd}(b, r_1)$, donc $\text{pgcd}(11\,466, 1\,656) = \text{pgcd}(1\,656, 1\,530)$.
- Division euclidienne de $b = 1\,656$ par $r_1 = 1\,530$: $1\,656 = 1 \times 1\,530 + 126$, donc le reste est $r_2 = 126$. Notre pgcd d vaut maintenant $\text{pgcd}(1\,530, 126)$.
- Division euclidienne de $1\,530$ par 126 : $1\,530 = 12 \times 126 + 18$, donc le reste est $r_3 = 18$ et $d = \text{pgcd}(126, 18)$.
- Division euclidienne de 126 par 18 : $126 = 7 \times 18 + 0$, donc le reste est nul. Or $\text{pgcd}(18, 0) = 18$.

- Le pgcd est le dernier reste non nul, ainsi $\text{pgcd}(11\,466, 1656) = 18$.

L'algorithme d'Euclide est l'un des plus anciens algorithmes mais il est cependant très efficace ! Le nombre d'étapes dans l'algorithme est assez faible : si a et b s'écrivent (en base 10) avec moins de n chiffres, alors il y a au plus $5n$ étapes dans l'algorithme d'Euclide. Donc par exemple, avec des entiers à 100 chiffres, il y a au plus 500 étapes.

1.7. Lemme de Gauss

Proposition 1 (Lemme de Gauss).

Soient $a, b, c \in \mathbb{Z}$ (avec a non nul).

$$\boxed{\text{Si } a|bc \text{ et } \text{pgcd}(a, b) = 1 \text{ alors } a|c}$$

Exemple : si un entier a divise $(a + 1)c$ alors $a|c$ (c'est le lemme de Gauss, sachant qu'on a toujours $\text{pgcd}(a, a + 1) = 1$).

Attention aux hypothèses : 6 divise 4×9 , mais 6 ne divise ni 4 ni 9. Cela ne contredit pas le lemme de Gauss car 6 n'est premier ni avec 4, ni avec 9.

La preuve découle du théorème de Bézout : comme par hypothèse $\text{pgcd}(a, b) = 1$, il existe $u, v \in \mathbb{Z}$ tels que $au + bv = 1$. On multiplie cette égalité par c pour obtenir $acu + bcv = c$. Mais $a|acu$ et par hypothèse $a|bcv$ donc a divise $acu + bcv = c$.

2. Nombres premiers

2.1. Définition

Définition.

Un **nombre premier** p est un entier supérieur ou égal à 2 dont les seuls diviseurs positifs sont 1 et p .

Exemples.

- 2, 3, 5, 7, 11, 13, ... sont des nombres premiers.
- Un théorème d'Euclide nous dit qu'il y a une infinité de nombres premiers.
- Par définition, 1 n'est pas un nombre premier.

2.2. Décomposition en facteurs premiers

Théorème 2.

Tout entier $n \geq 2$ se décompose en produit de facteurs premiers :

$$n = p_1^{\alpha_1} \times p_2^{\alpha_2} \times \cdots \times p_r^{\alpha_r}$$

où les p_i sont des nombres premiers, et les exposants $\alpha_i \geq 1$ sont des entiers. De plus, cette décomposition est unique (à l'ordre des facteurs près).

2.3. Petit théorème de Fermat

Théorème 3 (Petit théorème de Fermat).

Si p est un nombre premier ne divisant pas a alors

$$\boxed{a^{p-1} \equiv 1 \pmod{p}}$$

Une variante : pour a un entier quelconque et p un nombre premier :

$$a^p \equiv a \pmod{p}$$

Nous reviendrons sur les congruences dans la section suivante.

2.4. Algorithmes et nombres premiers

Nous allons discuter de plusieurs algorithmes qui permettent de décider si un entier n donné est un nombre premier ou pas.

Tester les diviseurs un par un. On teste si d divise n pour $d = 2, d = 3, d = 4, \dots$. Cela se fait par un calcul de division euclidienne. Si on obtient un diviseur strictement inférieur à n , alors n n'est pas premier. Si on ne trouve pas de diviseur alors n est premier.

Améliorations possibles : on peut tester $d = 2$, et ensuite ne tester que des entiers d impairs ; on peut aussi limiter la recherche des diviseurs à ceux vérifiant $d \leq \sqrt{n}$ (critère de Napoléon).

Crible d'Ératosthène. On peut aussi dresser une longue liste de nombres premiers, au delà de l'entier n . L'avantage est qu'il suffit alors de vérifier si l'entier n est dans la liste pour savoir s'il est premier. Mais le crible est une méthode lente et ne permet pas d'obtenir de très grands nombres premiers.

Test probabiliste de Fermat. Le petit théorème de Fermat nous dit que si p est un nombre premier et a est un entier avec $1 \leq a < p$ alors $a^{p-1} \equiv 1 \pmod{p}$.

Pour tester si un entier n est premier, on fixe un entier $1 < a < n$, on calcule $a^{n-1} \pmod{n}$ (c'est très facile grâce à l'exponentiation rapide, voir plus loin).

- Si $a^{n-1} \not\equiv 1 \pmod{n}$ alors on est sûr que n n'est pas un nombre premier.
- Si $a^{n-1} \equiv 1 \pmod{n}$ alors on dit que n valide le test de Fermat pour l'entier a et qu'il est probablement premier. Cependant il existe des exceptions : certains entiers valident le test de Fermat mais ne sont pas des nombres premiers. Par exemple parmi tous les entiers $n \leq 1\,000\,000$, tous ceux qui passent le test de Fermat à la fois pour $a = 2, a = 3, a = 5$ et $a = 7$ sont des nombres premiers (il y en a 78 498) à l'exception de 19 entiers (le premier de la liste est $n = 29\,341 = 13 \times 37 \times 61$).

Le test de Fermat permet de produire des entiers très grands qui sont probablement des nombres premiers : on choisit un entier impair n au hasard, on effectue un test de Fermat, si le test est concluant alors n est probablement un nombre premier, sinon on essaie l'entier $n + 2$...

Algorithmes modernes. Une amélioration du test de Fermat est l'algorithme de Miller-Rabin. Par ailleurs, il a été récemment démontré par Agrawal–Kayal–Saxena que le test de primalité peut être effectué en temps polynomial (algorithme AKS). Même si dans la pratique l'algorithme n'est pas très utile, c'est une grande avancée théorique. L'algorithme est basé sur le fait que si p est un nombre premier alors on a l'égalité polynomiale : $(X + 1)^p \equiv X^p + 1 \pmod{p}$.

2.5. Algorithmes et factorisation

Tester si un entier est premier ou donner sa factorisation sont au final deux problèmes distincts. On a vu, grâce au petit théorème de Fermat, qu'on peut décider qu'un entier n'est pas premier sans lui avoir trouvé de facteur. Le problème de factoriser un entier n , ou au moins de trouver un facteur non trivial, est donc plus difficile.

Tester les diviseurs un par un. Comme auparavant on peut tester les diviseurs un par un jusqu'à \sqrt{n} , la complexité est en $O(\sqrt{n})$. On peut bien sûr ne tester que les diviseurs premiers (ce qui est plus rapide) mais cela demande au préalable d'avoir une liste des premiers nombres premiers (ce qui est long). C'est donc une méthode efficace uniquement pour trouver les petits diviseurs.

Facteurs de Fermat. On peut essayer d'exprimer n comme différence de deux carrés, on obtient alors une factorisation. En effet, si c'est le cas :

$$n = a^2 - b^2 = (a - b)(a + b).$$

Réciproquement tout entier impair non premier est la différence de deux carrés. En effet, si $n = cd$ alors $n = \left(\frac{c+d}{2}\right)^2 - \left(\frac{c-d}{2}\right)^2$. Cela fournit un algorithme de recherche d'un facteur de n : prendre un entier a (généralement proche de \sqrt{n}), calculer si $b' = n - a^2$ est un carré de la forme b^2 , si c'est le cas, on obtient une factorisation $(a - b)(a + b)$, sinon on recommence avec $a + 1$. C'est une méthode efficace si les facteurs premiers sont grands, donc cette méthode est complémentaire de la précédente.

Algorithmes modernes.

Aucun algorithme connu n'a de complexité polynomiale. Plus précisément, on ne connaît pas d'algorithme de factorisation ayant une complexité polynomiale, mais on ne sait pas non plus prouver qu'un tel algorithme n'existe pas.

Le meilleur algorithme connu pour factoriser des grands entiers est l'algorithme GNFS (pour *General Number Field Sieve*), sa complexité est environ $O(n^{\frac{1}{3}})$ (c'est donc une complexité exponentielle par rapport à la taille de n qui est d'ordre $\ln(n)$).

3. Congruence modulo n

3.1. Modulo n

Définition.

Soit $n \geq 1$ un entier. On dit $a \equiv b \pmod{n}$ s'il existe $k \in \mathbb{Z}$ tel que $a = kn + b$, autrement dit si $b - a$ est divisible par n . On dira *a est congru à b modulo n* .

Proposition 2.

Si $a \equiv b \pmod{n}$ et $a' \equiv b' \pmod{n}$ alors :

$$a + a' \equiv b + b' \pmod{n} \quad \text{et} \quad a \cdot a' \equiv b \cdot b' \pmod{n}.$$

De plus $a^k \equiv b^k \pmod{n}$, quel que soit $k \in \mathbb{N}$.

Exemples.

- $33 \equiv 3 \pmod{15}$ car $33 = 2 \times 15 + 3$.
- $1789 = 105 \times 17 + 4$ donc $1789 \equiv 4 \pmod{17}$, mais aussi $1789 = 104 \times 17 + 21$ et $1789 \equiv 21 \pmod{17}$.
- Un entier a est pair si et seulement si $a \equiv 0 \pmod{2}$.
- Par conséquent si a est impair alors a^k est impair. En effet a impair, implique $a \equiv 1 \pmod{2}$, donc $a^k \equiv 1^k \equiv 1 \pmod{2}$, donc a^k est impair.

3.2. Inverse modulo n

Définition.

Soit $n \geq 1$ un entier. On dit que a est *inversible* modulo n , s'il existe $b \in \mathbb{Z}$ tel que $a \cdot b \equiv 1 \pmod{n}$. On dit alors que b est l'inverse de a modulo n .

Exemples.

- Avec $n = 15$, $a = 2$ est inversible modulo 15, car avec $b = 8$ on a $a \cdot b = 2 \times 8 = 16 \equiv 1 \pmod{15}$.
- Avec $n = 15$, $a = 7$ est inversible modulo 15, car avec $b = 13$ on a $a \cdot b = 7 \times 13 = 91 \equiv 1 \pmod{15}$.
- Avec $n = 15$, $a = 3$ n'est pas inversible.

Proposition 3.

a est inversible modulo n si et seulement si $\text{pgcd}(a, n) = 1$.

Démonstration.

$$\begin{aligned} \text{pgcd}(a, n) = 1 &\iff \exists u, v \in \mathbb{Z} \quad au + nv = 1 && \text{(par le théorème de Bézout)} \\ &\iff \exists u, v \in \mathbb{Z} \quad au = 1 - nv \\ &\iff \exists u \in \mathbb{Z} \quad au \equiv 1 \pmod{n} \\ &\iff a \text{ est inversible modulo } n. \end{aligned}$$

□

La preuve justifie que l'on peut trouver l'inverse de a modulo n à l'aide des coefficients de Bézout u, v . Ces coefficients se calculent à l'aide de l'algorithme d'Euclide étendu.

3.3. Groupes

Pour être un peu plus théorique, on définit :

- $(\mathbb{Z}/n\mathbb{Z}, +)$ le groupe additif des entiers modulo n . C'est un groupe commutatif ($a + b \equiv b + a \pmod{n}$), ayant n éléments. L'élément neutre pour l'addition est 0, l'inverse d'un élément a est $-a$.
- $(\mathbb{Z}/n\mathbb{Z})^*, \times$ le groupe des inversibles modulo n . C'est un groupe multiplicatif, commutatif ($a \times b \equiv b \times a \pmod{n}$). Son élément neutre est 1, l'inverse d'un élément a est son inverse modulo n , noté b , tel que $ab \equiv 1 \pmod{n}$. Le groupe $(\mathbb{Z}/n\mathbb{Z})^*$ possède $\varphi(n)$ éléments (où $\varphi(n)$ est défini juste après).

3.4. Indicatrice d'Euler

Définition.

Soit $n \geq 1$. L'*indicatrice d'Euler* $\varphi(n)$ est le nombre d'entiers a premiers avec n , tels que $1 \leq a \leq n$.

Une conséquence immédiate est que $\varphi(n)$ est le nombre d'éléments inversibles modulo n : $\varphi(n) = \text{Card}(\mathbb{Z}/n\mathbb{Z})^*$.

Exemples.

- Soit $n = 15$. Les entiers a premiers avec 15 sont $\{1, 2, 4, 7, 8, 11, 13, 14\}$, donc $\varphi(15) = 8$.
- Si p est nombre premier alors $\varphi(p) = p - 1$ car tout entier a , avec $1 \leq a < p$, est premier avec p .

La proposition suivante permet de calculer $\varphi(n)$ à partir de la décomposition de n en facteurs premiers.

Proposition 4.

- Si $n = pq$ (avec p, q deux nombres premiers distincts) alors $\varphi(n) = (p - 1)(q - 1)$.
- Si $n = p^k$ (avec $k \geq 1$) alors $\varphi(n) = p^k - p^{k-1}$.
- Formule générale. Si $n = p_1^{\alpha_1} \cdots p_\ell^{\alpha_\ell}$ alors

$$\varphi(n) = n \prod_{i=1}^{\ell} \left(1 - \frac{1}{p_i}\right).$$

Démonstration.

- Si $n = pq$ alors les entiers a qui ne sont pas premiers avec n sont les $p, 2p, 3p, \dots, (q - 1)p$ et les $q, 2q, \dots, (p - 1)q$ et pq . Il y a en a donc $(q - 1) + (p - 1) + 1 = p + q - 1$. Les autres sont premiers avec n et sont au nombre de $pq - (p + q - 1) = (p - 1)(q - 1)$.
- Si $n = p^k$ alors les entiers qui ne sont pas premiers avec n sont les multiples de p de la forme ap avec $1 \leq a \leq p^{k-1}$. Il y en a donc p^{k-1} .
- Nous admettons la formule générale qui se prouve par récurrence à partir de la formule $\varphi(ab) = \varphi(a) \cdot \varphi(b)$ lorsque a et b sont premiers entre eux.

□

3.5. Théorème d'Euler

Le théorème d'Euler est une version généralisée du petit théorème de Fermat.

Théorème 4 (Théorème d'Euler).

Si a et n sont premiers entre eux alors :

$$a^{\varphi(n)} \equiv 1 \pmod{n}$$

On pourrait déduire ce résultat du théorème de Lagrange appliqué au groupe fini $(\mathbb{Z}/n\mathbb{Z})^*$ de cardinal $\varphi(n)$. Nous allons en donner une autre démonstration.

Démonstration. Fixons n et fixons a premier avec n . Notons $\mathcal{A} = \{a_1, a_2, \dots, a_{\varphi(n)}\}$ l'ensemble des entiers inférieurs à n et premiers avec n (notre entier a est l'un de ces éléments et en fait $\mathcal{A} = (\mathbb{Z}/n\mathbb{Z})^*$). Considérons l'application $f : \mathcal{A} \rightarrow \mathcal{A}$, définie par $f(a_i) = a \cdot a_i \pmod{n}$. Comme a est inversible modulo n , alors f est bijective (sa bijection réciproque est $f^{-1}(a_i) = b \cdot a_i \pmod{n}$ où b est l'inverse de a modulo n). Ainsi $\mathcal{A}' = \{a \cdot a_1, a \cdot a_2, \dots, a \cdot a_{\varphi(n)}\}$ contient les mêmes termes que notre ensemble $\mathcal{A} = \{a_1, \dots, a_{\varphi(n)}\}$ (mais les éléments sont permutés) : $\mathcal{A}' = \mathcal{A}$. Comme ces ensembles sont égaux, alors le produit des éléments de \mathcal{A}' est égal au produit des éléments de \mathcal{A} :

$$\prod_{i=1}^{\varphi(n)} (a \cdot a_i) \equiv \prod_{i=1}^{\varphi(n)} a_i \pmod{n}$$

donc

$$a^{\varphi(n)} \prod_{i=1}^{\varphi(n)} a_i \equiv \prod_{i=1}^{\varphi(n)} a_i \pmod{n},$$

et comme les a_i sont inversibles modulo n , alors $a^{\varphi(n)} \equiv 1 \pmod{n}$. □

3.6. Exponentiation

Fixons $n \geq 1$ et $a \in \mathbb{Z}$. Il s'agit de calculer $a^k \pmod{n}$, pour un entier $k \geq 0$.

Exponentiation classique. Si on a besoin de connaître tous les $a^k \pmod{n}$ pour $k = 1, 2, 3, \dots$ alors on les calcule successivement en utilisant la relation $a^{k+1} = a^k \cdot a$.

Exemple.

Calcul des $2^k \pmod{25}$.

$$\begin{array}{ll} 2^0 \equiv 1 \pmod{25} & 2^5 \equiv 2 \times 16 \equiv 7 \pmod{25} \\ 2^1 \equiv 2 \pmod{25} & 2^6 \equiv 2 \times 7 \equiv 14 \pmod{25} \\ 2^2 \equiv 2 \times 2 \equiv 4 \pmod{25} & 2^7 \equiv 2 \times 14 \equiv 28 \equiv 3 \pmod{25} \\ 2^3 \equiv 2 \times 4 \equiv 8 \pmod{25} & 2^8 \equiv 2 \times 3 \equiv 6 \pmod{25} \\ 2^4 \equiv 2 \times 8 \equiv 16 \pmod{25} & \dots \end{array}$$

Noter que chaque calcul est une simple multiplication par a du résultat précédent et qu'on réduit immédiatement modulo n afin que les entiers en jeu restent de petite taille.

Exponentiation rapide. Si on a besoin de connaître un seul $a^k \pmod{n}$, alors on n'est pas obligé de calculer toutes les puissances précédentes, mais seulement celles dont l'exposant k est une puissance de 2.

Exemple.

On souhaite calculer $3^{21} \pmod{31}$.

- On décompose l'exposant 21 en base 2 : $21 = 16 + 4 + 1$.

- On calcule successivement $3^1, 3^2, 3^4, 3^8, \dots$ en utilisant que $3^{2k} = (3^k)^2$. De plus tous les calculs se font modulo $n = 31$.

$$3^1 \equiv 3 \pmod{31}$$

$$3^2 \equiv 9 \pmod{31}$$

$$3^4 \equiv (3^2)^2 \equiv 9^2 \equiv 81 \equiv 19 \pmod{31}$$

$$3^8 \equiv (3^4)^2 \equiv 19^2 \equiv 361 \equiv 20 \pmod{31}$$

$$3^{16} \equiv (3^8)^2 \equiv 20^2 \equiv 400 \equiv 28 \pmod{31}$$

- On combine les résultats :

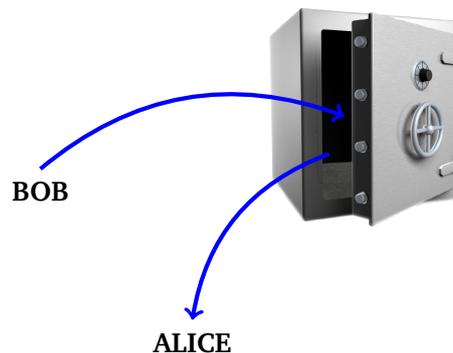
$$3^{21} = 3^{16+4+1} = 3^{16} \times 3^4 \times 3^1 \equiv 28 \times 19 \times 3 \equiv 1596 \equiv 15 \pmod{31}.$$

4. Cryptographie RSA

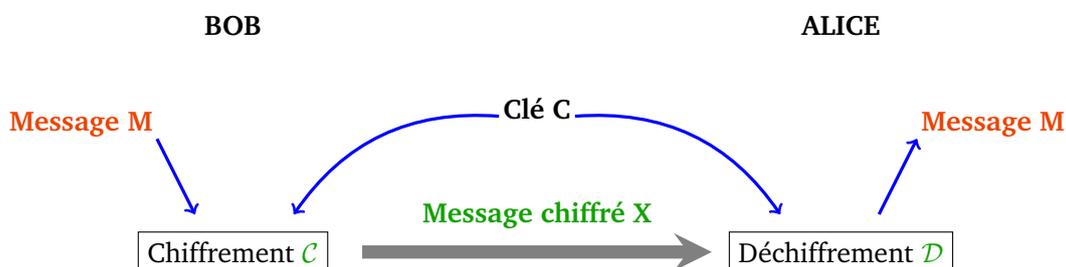
4.1. Chiffrement à clé secrète

Jusqu'à récemment pour que Bob envoie un message à Alice, sans que personne ne puisse prendre connaissance du contenu, on utilisait un *chiffrement à clé secrète*. Une méthode (très basique) consiste par exemple à décaler chaque lettre d'un certain rang C . Par exemple si $C = 3$, Bob chiffre son message **BAC** en **EDF**. Alice peut facilement déchiffrer le message si elle connaît la clé C .

On représente ce protocole ainsi : Bob dépose son message dans un coffre fort pour Alice, Alice et Bob étant les seuls à posséder la clé du coffre.



La grande difficulté est que Alice et Bob doivent d'abord se communiquer la clé.

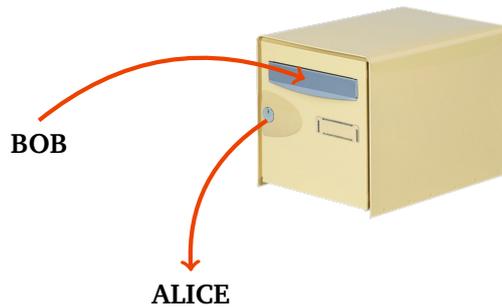


4.2. Chiffrement à clé publique

Le chiffrement à clé publique est une petite révolution : n'importe qui peut envoyer un message chiffré à Alice en utilisant la clé publique d'Alice, mais seule Alice peut déchiffrer le message à l'aide d'une clé secrète qu'elle est la seule à connaître.

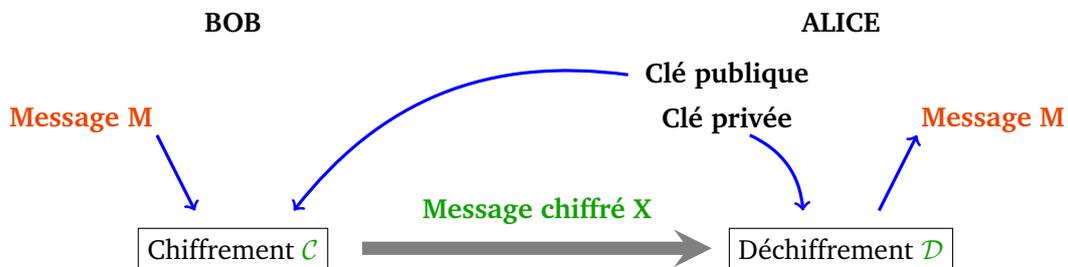
De façon imagée, si Bob veut envoyer un message à Alice, il dépose son message dans la boîte aux lettres d'Alice, seule Alice pourra ouvrir sa boîte et consulter le message. Ici la clé publique est symbolisée par la

boîte aux lettres, tout le monde peut y déposer un message, la clé qui ouvre la boîte aux lettres est la clé privée d'Alice.



Si Bob veut envoyer un message secret à Alice, le processus se décompose ainsi :

1. Alice prépare une clé publique et une clé privée,
2. Bob utilise la clé publique d'Alice pour chiffrer son message,
3. Alice reçoit le message chiffré et le déchiffre grâce à sa clé privée.



Pour chiffrer un message, on commence par le transformer en un –ou plusieurs– nombres. Dans toute la suite le message que Bob envoie est un entier.

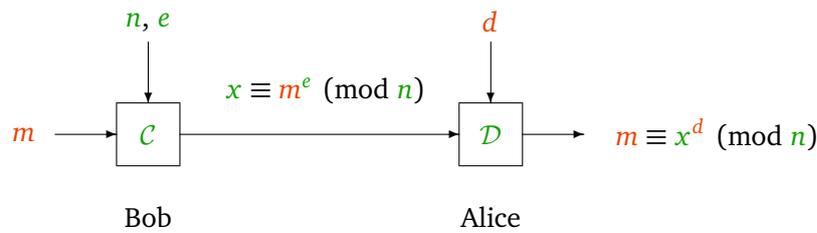
4.3. Principe du chiffrement RSA

Voici en résumé le protocole RSA.

- On choisit deux nombres premiers p et q que l'on garde secrets et on pose $n = p \times q$. Le principe étant que même connaissant n il est très difficile de retrouver p et q .
- La clé secrète et la clé publique se déterminent à l'aide de l'algorithme d'Euclide et des coefficients de Bézout.
- Les calculs de chiffrement se feront modulo n .
- Le déchiffrement fonctionne grâce au théorème d'Euler.

Et voici un schéma qui présente le chiffrement et le déchiffrement :

- n, e forment la clé publique d'Alice
- d est la clé privée d'Alice,
- m est le message secret que Bob souhaite transmettre à Alice,
- x est le message chiffré que Bob calcule à partir de la clé publique d'Alice et qu'il lui transmet,
- seule Alice peut retrouver m par un calcul à partir de x et de sa clé privée.



4.4. Protocole du chiffrement RSA

Choix de deux nombres premiers

Alice effectue, une fois pour toutes, les opérations suivantes (en secret) :

- elle choisit deux nombres premiers distincts p et q (dans la pratique ce sont de très grands nombres, jusqu'à des centaines de chiffres),
- elle calcule $n = p \times q$,
- elle calcule $\varphi(n) = (p - 1) \times (q - 1)$.

Vous noterez que le calcul de $\varphi(n)$ n'est possible que si la décomposition de n sous la forme $p \times q$ est connue. D'où le caractère secret de $\varphi(n)$ même si n est connu de tous.

Choix d'un exposant et calcul de son inverse

Alice continue :

- elle choisit un exposant e tel que $\text{pgcd}(e, \varphi(n)) = 1$,
- elle calcule l'inverse d de e modulo $\varphi(n)$: $d \times e \equiv 1 \pmod{\varphi(n)}$. Ce calcul se fait par l'algorithme d'Euclide étendu.

Clé publique/clé privée

La **clé publique** d'Alice est constituée des deux nombres :

$$n \text{ et } e$$

Et comme son nom l'indique, Alice communique sa clé publique au monde entier.

Alice garde pour elle sa **clé privée** :

$$d$$

Noter que le calcul de d nécessite $\varphi(n)$, qu'Alice est la seule à connaître. Alice peut détruire p , q et $\varphi(n)$ qui ne sont plus utiles. Elle ne conserve secrètement que sa clé privée.

Message chiffré

Le message est un entier m , tel que $0 \leq m < n$.

Bob récupère la clé publique d'Alice, n et e , avec laquelle il calcule :

$$x \equiv m^e \pmod{n}$$

Il transmet ce message x à Alice.

Déchiffrement du message

Alice reçoit le message x chiffré par Bob, elle le déchiffre à l'aide de sa clé privée d , par l'opération :

$$m \equiv x^d \pmod{n}$$

Nous allons prouver dans le lemme 1 que cette opération permet à Alice de retrouver le message original m de Bob.

4.5. Exemple

Mise en place par Alice.

- Alice choisit $p = 5$ et $q = 11$, elle calcule $n = p \times q = 55$.
- Elle calcule aussi $\varphi(n) = (p - 1)(q - 1) = 4 \times 10 = 40$.
- Alice choisit par exemple l'entier $e = 3$ qui est bien premier avec $\varphi(n)$.
- Alice calcule d , l'inverse de e modulo $\varphi(n)$, ici elle trouve $d = 27$ car $3 \times 27 = 81 \equiv 1 \pmod{40}$.
- La clé publique d'Alice est $(n, e) = (55, 3)$, sa clé privée est $d = 27$.

Envoi du message de Bob à Alice.

- Bob souhaite envoyer le message $m = 41$ à Alice.
- Bob calcule $x \equiv m^e \pmod{n}$ à l'aide de la clé publique d'Alice. Ici

$$x \equiv m^e \equiv 41^3 \equiv 68921 \equiv 6 \pmod{55}.$$

- Bob transmet $x = 6$ à Alice.
- Seule Alice peut déchiffrer le message à l'aide de sa clé secrète d , en effet le calcul de $x^d \pmod{n}$ redonne le message original m . Ici

$$x^d \equiv 6^{27} \equiv 41 \pmod{55}.$$

Ainsi Alice obtient bien le message $m = 41$. (Pour le calcul de $6^{27} \pmod{55}$ on utilise les techniques d'exponentiation vues précédemment).

4.6. Lemme de déchiffrement

Le principe de déchiffrement repose sur le théorème d'Euler.

Lemme 1.

Soit d l'inverse de e modulo $\varphi(n)$.

$$\text{Si } x \equiv m^e \pmod{n} \text{ alors } m \equiv x^d \pmod{n}.$$

Ce lemme prouve bien que le message original m de Bob, chiffré à l'aide de la clé publique d'Alice (e, n) en le message x , peut-être retrouvé par Alice à l'aide de sa clé secrète d .

Démonstration.

- Que d soit l'inverse de e modulo $\varphi(n)$ signifie $d \cdot e \equiv 1 \pmod{\varphi(n)}$. Autrement dit, il existe $k \in \mathbb{Z}$ tel que $d \cdot e = 1 + k \cdot \varphi(n)$.
- On rappelle que, par le théorème d'Euler, lorsque m et n sont premiers entre eux

$$m^{\varphi(n)} \equiv 1 \pmod{n}.$$

- Supposons $\text{pgcd}(m, n) = 1$.

Notons $x \equiv m^e \pmod{n}$ et calculons x^d :

$$x^d \equiv (m^e)^d \equiv m^{e \cdot d} \equiv m^{1+k \cdot \varphi(n)} \equiv m \cdot m^{k \cdot \varphi(n)} \equiv m \cdot (m^{\varphi(n)})^k \equiv m \cdot (1)^k \equiv m \pmod{n}.$$

- On admet que si m et n ne sont pas premiers entre eux, le résultat reste vrai (il faut adapter les arguments précédents).

□

Note. Certains passages de ce chapitre sont extraits du chapitre « Arithmétique » du livre « Algèbre » d'Exo7. La section sur le chiffrement RSA est tirée du cours « Cryptographie » écrit avec François Recher.

Algorithme de Shor

Vidéo ■ partie 13.1. Arithmétique pour l'algorithme de Shor

Vidéo ■ partie 13.2. Début de l'algorithme de Shor

Vidéo ■ partie 13.3. Transformée de Fourier discrète (pour Shor)

Vidéo ■ partie 13.4. Fin de l'algorithme de Shor

Nous détaillons le circuit et les calculs qui permettent une factorisation rapide des entiers à l'aide d'un ordinateur quantique.

1. Arithmétique pour l'algorithme de Shor

1.1. Objectif : factoriser N

Soit N un entier. Nous aimerions décomposer N en produit de facteurs premiers : $N = p_1^{\alpha_1} p_2^{\alpha_2} \cdots p_s^{\alpha_s}$. Pour cela il suffit de trouver un algorithme qui fournit un facteur d de N , avec $1 < d < N$. Ensuite pour obtenir la factorisation complète, il suffit d'appliquer itérativement cet algorithme à d et à $\frac{N}{d}$. Par exemple dans le cas $N = pq$ de la cryptographie RSA, il n'y a qu'une seule étape.

Au début de l'algorithme, on commence par choisir au hasard un entier a avec $1 < a < N$. On calcule le pgcd de a et de N par l'algorithme d'Euclide (c'est une étape très rapide).

- Si $\text{pgcd}(a, N) \neq 1$ alors $d = \text{pgcd}(a, N)$ est un facteur non-trivial de N et l'algorithme est terminé! (Par exemple dans le cas $N = pq$, cette situation est rare, car il faudrait choisir a un multiple de p ou de q .)
- Si $\text{pgcd}(a, N) = 1$, alors $a \in (\mathbb{Z}/N\mathbb{Z})^*$, c'est-à-dire a est inversible modulo N . En particulier il existe un entier $k > 0$, tel que $a^k \equiv 1 \pmod{N}$.

1.2. Ordre

Définition.

On appelle **ordre** d'un entier a modulo N , le plus petit entier r strictement positif tel que $a^r \equiv 1 \pmod{N}$:

$$r = \min \{k > 0 \mid a^k \equiv 1 \pmod{N}\}.$$

Le théorème de Lagrange pour le groupe $(\mathbb{Z}/N\mathbb{Z})^*$ de cardinal $\varphi(N)$ donne une borne sur r .

Proposition 1.

Si $\varphi(N)$ est l'indicatrice d'Euler et a est un nombre premier avec N , alors

$$a^{\varphi(N)} \equiv 1 \pmod{N}$$

et l'ordre de a modulo n divise $\varphi(N)$.

Mais attention, ni l'ordre r , ni l'indicatrice $\varphi(N)$ ne sont faciles à calculer. Par exemple dans le cas $N = pq$, alors $\varphi(N) = (p-1)(q-1)$ et on ne peut pas calculer $\varphi(N)$ sans connaître les facteurs p et q (que l'on ne connaît pas car c'est ce que l'on veut calculer).

1.3. Période

Proposition 2.

L'ordre r de l'entier a modulo N est la plus petite période de la fonction $k \mapsto a^k \pmod{N}$.

Démonstration.

- Par définition de l'ordre r , on sait $a^r \equiv 1 \pmod{N}$ et $a^k \not\equiv 1 \pmod{N}$ pour $0 < k < r$.

- r est une période :

$$a^{k+\ell r} = a^k \cdot a^{\ell r} = a^k \cdot (a^r)^\ell \equiv a^k \cdot 1^\ell \equiv a^k \pmod{N},$$

donc $f(k + \ell r) = f(k)$.

- r est la plus petite période : par l'absurde si $s < r$ était une période plus petite, alors $f(s) = f(0)$ donc $a^s \equiv a^0 \equiv 1 \pmod{N}$. Mais par définition, r est le plus petit entier tel que $a^r \equiv 1 \pmod{N}$, donc $s \geq r$ et nous avons une contradiction. □

1.4. Facteurs de N

Nous allons faire plusieurs hypothèses au cours de ce chapitre. Nous discuterons plus tard de leur pertinence.

Hypothèse 1. L'ordre r de a modulo N est pair.

Nous verrons dans le chapitre suivant « Compléments d'arithmétique » que c'est le cas pour plus de la moitié des entiers a choisis au départ. Si r n'est pas pair, alors on arrête l'algorithme et on choisit une nouvelle valeur de a . Si r est pair, à l'aide de l'identité $a^2 - b^2 = (a - b)(a + b)$ et sachant que $a^r - 1 \equiv 0 \pmod{N}$, on obtient alors

$$(a^{r/2} - 1)(a^{r/2} + 1) \equiv 0 \pmod{N}$$

Cette décomposition est la clé pour une factorisation de N .

Hypothèse 2. $a^{r/2} + 1$ n'est pas divisible par N .

Encore une fois nous verrons dans le chapitre suivant que c'est le cas pour une majorité des entiers a choisis au départ.

Proposition 3.

Avec les hypothèses 1 et 2, les entiers

$$d = \text{pgcd}(a^{r/2} - 1, N) \quad \text{et} \quad d' = \text{pgcd}(a^{r/2} + 1, N)$$

sont des facteurs non triviaux de N .

Lemme 1.

Si $ab \equiv 0 \pmod{N}$ avec $a \not\equiv 0 \pmod{N}$ et $b \not\equiv 0 \pmod{N}$ alors $\text{pgcd}(a, N)$ et $\text{pgcd}(b, N)$ sont des diviseurs non triviaux de N .

Remarque.

L'anneau des entiers \mathbb{Z} est intègre, cela signifie que si un produit ab est nul alors l'un des facteurs a ou b est nul. Ici ce n'est pas le cas, car si N n'est pas un nombre premier l'anneau $\mathbb{Z}/N\mathbb{Z}$ n'est pas intègre. Par exemple avec $N = 6$ on a $2 \times 3 \equiv 0 \pmod{6}$.

Preuve du lemme. Supposons $ab \equiv 0 \pmod{N}$, c'est-à-dire N divise ab .

- Si on avait $\text{pgcd}(a, N) = 1$, comme $N|ab$, par le lemme de Gauss on a $N|b$, donc $b \equiv 0 \pmod{N}$, ce qui donnerait une contradiction.
 - Bien sûr $d = \text{pgcd}(a, N)$ est plus petit que N , et d'autre part $d \neq N$ car sinon $a \equiv 0 \pmod{N}$.
- Ainsi $d = \text{pgcd}(a, N)$ est un diviseur de N avec $1 < d < N$, de même pour $d' = \text{pgcd}(b, N)$. \square

Preuve de la proposition.

- Tout d'abord $a^{r/2} - 1 \not\equiv 0 \pmod{N}$, car sinon $a^{r/2} \equiv 1 \pmod{N}$ ce qui contredirait que r est le plus petit entier tel que $a^r \equiv 1 \pmod{N}$.
 - L'hypothèse 2 dit exactement que $a^{r/2} + 1 \not\equiv 0 \pmod{N}$.
 - Par le lemme $d = \text{pgcd}(a^{r/2} - 1, N)$ et $d' = \text{pgcd}(a^{r/2} + 1, N)$ sont des facteurs non triviaux de N .
- \square

1.5. Exemple de $N = 15$

Prenons $N = 15$.

- Si l'entier a est choisi parmi $\{3, 5, 6, 9, 10, 12\}$, alors a n'est pas premier avec N . Dans ce cas $d = \text{pgcd}(a, N)$ donne un diviseur strict de N et c'est terminé. Par exemple si $a = 9$, alors $d = \text{pgcd}(9, 15) = 3$ est un facteur de $N = 15$.
- Si l'entier $a \in \{2, 4, 7, 8, 11, 13, 14\}$ alors $\text{pgcd}(a, N) = 1$. Il faut maintenant calculer l'ordre de a .
- Prenons l'exemple de $a = 2$. Alors l'ordre de 2 modulo 15 est $r = 4$, car $2^4 = 16 \equiv 1 \pmod{15}$.
 - $a^{r/2} - 1 = 2^2 - 1 = 3$ ainsi $d = 3 = \text{pgcd}(3, 15)$ est un facteur de N .
 - $a^{r/2} + 1 = 2^2 + 1 = 5$ ainsi $d' = 5 = \text{pgcd}(5, 15)$ est aussi un facteur de N .
 - Dans ce cas nous avons factorisé $15 = 3 \times 5$.
- Prenons l'exemple de $a = 7$. Alors l'ordre de 7 modulo 15 est encore $r = 4$, car $7^4 = 2401 \equiv 1 \pmod{15}$.
 - $a^{r/2} - 1 = 7^2 - 1 = 48$ ainsi $d = \text{pgcd}(48, 15) = 3$ est un facteur de N .
 - $a^{r/2} + 1 = 7^2 + 1 = 50$ ainsi $d' = \text{pgcd}(50, 15) = 5$ est aussi un facteur de N .
 - Dans ce cas nous avons factorisé $15 = 3 \times 5$.

Voici une table qui résume les différents cas pour $N = 15$:

a	a premier avec N ? (et ordre)	facteurs
2	oui $r = 4$	$d = \text{pgcd}(2^{4/2} - 1, 15) = 3$ et $d' = \text{pgcd}(2^{4/2} + 1, 15) = 5$
3	non	$d = \text{pgcd}(3, 15) = 3$
4	oui $r = 2$	$d = \text{pgcd}(4^{2/2} - 1, 15) = 3$ et $d' = \text{pgcd}(4^{2/2} + 1, 15) = 5$
5	non	$d = \text{pgcd}(5, 15) = 5$
6	non	$d = \text{pgcd}(6, 15) = 3$
7	oui $r = 4$	$d = \text{pgcd}(7^{4/2} - 1, 15) = 3$ et $d' = \text{pgcd}(7^{4/2} + 1, 15) = 5$
8	oui $r = 4$	$d = \text{pgcd}(8^{4/2} - 1, 15) = 3$ et $d' = \text{pgcd}(8^{4/2} + 1, 15) = 5$
9	non	$d = \text{pgcd}(9, 15) = 3$
10	non	$d = \text{pgcd}(10, 15) = 5$
11	oui $r = 2$	$d = \text{pgcd}(11^{2/2} - 1, 15) = 5$ et $d' = \text{pgcd}(11^{2/2} + 1, 15) = 3$
12	non	$d = \text{pgcd}(12, 15) = 3$
13	oui $r = 4$	$d = \text{pgcd}(13^{4/2} - 1, 15) = 3$ et $d' = \text{pgcd}(13^{4/2} + 1, 15) = 5$
14	oui $r = 2$	l'hypothèse 2 n'est pas vérifiée, échec

1.6. Exemple de $N = 21$

Fixons $N = 21$.

- Les $a \in \{3, 6, 7, 9, 12, 14, 15, 18\}$ ne sont pas premiers avec N . Dans ce cas $d = \text{pgcd}(a, N)$ donne un diviseur strict de N et c'est terminé.

- Les éléments a de $\{8, 13\}$ sont d'ordre $r = 2$; ceux de $\{2, 10, 11, 19\}$ sont d'ordre $r = 6$. Dans ces deux situations on obtient les facteurs d et d' égaux à 3 et 7.
- Les éléments $a = 4$ et $a = 16$ sont d'ordre $r = 3$ impair. L'hypothèse 1 n'est pas vérifiée et l'algorithme échoue.
- Pour $a = 5$, $a = 17$ ou $a = 20$, l'entier $N = 21$ divise $a^{r/2} + 1$. L'hypothèse 2 n'est pas vérifiée et l'algorithme échoue.

Exemple. Prenons $a = 2$. Son ordre est $r = 6$, car $2^6 = 64 \equiv 1 \pmod{21}$, l'ordre est pair; $d = \text{pgcd}(2^{6/2} - 1, 21) = \text{pgcd}(7, 21) = 7$ et $d' = \text{pgcd}(2^{6/2} + 1, 21) = \text{pgcd}(9, 21) = 3$ sont les facteurs de $N = 21$.

Exercice. Faire un tableau qui détaille tous les cas pour $N = 21$ (comme ci-dessus pour $N = 15$).

1.7. Calcul de l'ordre sur un ordinateur classique

Comme mentionné précédemment, il n'y a pas de formule pour calculer directement r ou $\varphi(N)$ si on ne connaît pas déjà les facteurs de N . Ainsi un algorithme d'informatique classique pour calculer l'ordre d'un élément a modulo N nécessiterait de calculer successivement a^1, a^2, a^3, \dots modulo N , jusqu'à trouver l'ordre r caractérisé par $a^r \equiv 1 \pmod{N}$. Il y a donc au total environ $O(N)$ calculs du type $a^k \pmod{N}$.

C'est là qu'intervient la magie de l'informatique quantique qui permet d'évaluer tous les a^k en même temps.

2. Début de l'algorithme de Shor

Pour un entier a fixé, le but est de calculer tous les a^k modulo N pour k variant de 0 à $N - 1$ afin de trouver l'ordre r pour lequel $a^r \equiv 1 \pmod{N}$. On rappelle que cet ordre r est aussi la plus petite période de la fonction $k \mapsto a^k \pmod{N}$.

2.1. Ordre

Fixons un entier a . Considérons la fonction f définie par

$$\begin{aligned} f : \mathbb{Z} &\longrightarrow \mathbb{Z}/N\mathbb{Z} \\ k &\longmapsto a^k \pmod{N}. \end{aligned}$$

Ainsi $f(1) = a \pmod{N}$, $f(2) = a^2 \pmod{N}$, $f(3) = a^3 \pmod{N}, \dots$

On rappelle qu'à une fonction $f : x \mapsto y$ on associe l'oracle $F : (x, y) \mapsto (x, y \oplus f(x))$. Donc l'oracle associé à notre fonction $f : k \mapsto a^k \pmod{N}$ est $F : (k, y) \mapsto (k, y \oplus a^k \pmod{N})$, mais notre circuit sera toujours initialisé avec $y = 0$, donc dans notre situation nous considérerons $F : (k, 0) \mapsto (k, a^k \pmod{N})$.

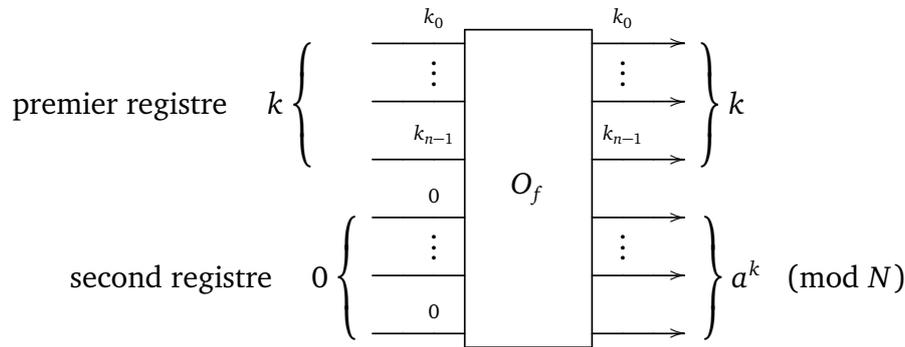
Exemple.

Pour $N = 15$ et $a = 2$:

$$\begin{array}{lll} (0, 0) & \xrightarrow{F} & (0, 1) \\ (1, 0) & \xrightarrow{F} & (1, 2) \\ (2, 0) & \xrightarrow{F} & (2, 4) \\ (3, 0) & \xrightarrow{F} & (3, 8) \\ (4, 0) & \xrightarrow{F} & (4, 1) \quad \text{car } 16 \equiv 1 \pmod{15} \\ (5, 0) & \xrightarrow{F} & (5, 2) \quad \text{car } 32 \equiv 2 \pmod{15} \\ (6, 0) & \xrightarrow{F} & (6, 4) \quad \text{car } 64 \equiv 4 \pmod{15} \\ & & \vdots \end{array}$$

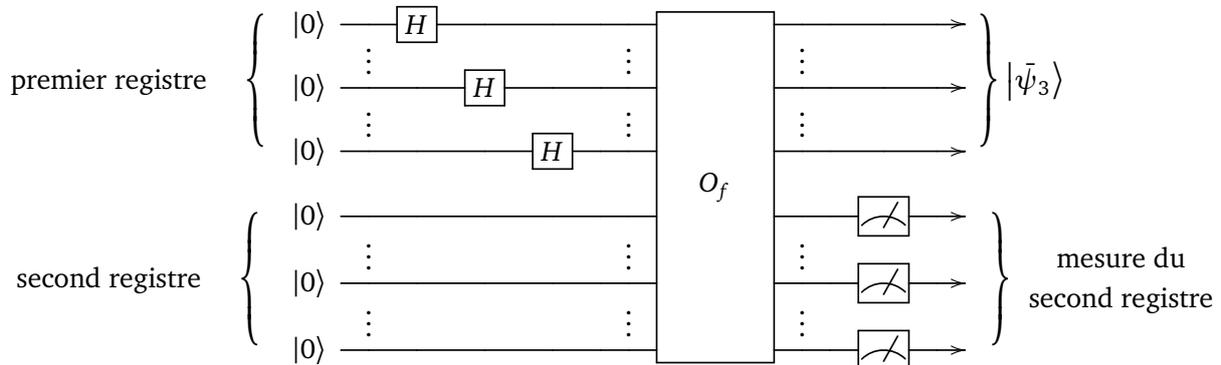
Choisissons un entier n tel que $2^n \geq N$. (Ce sera suffisant pour le cas étudié dans ce chapitre, mais dans le cas général il faut en fait avoir $2^n \geq N^2$, voir le chapitre « Compléments d'arithmétique ».)

On peut alors coder n'importe quel entier plus petit que 2^n à l'aide d'un n -bit : pour $0 \leq x < 2^n$, on note $\underline{x} = x_{n-1} \dots x_1 x_0$ son écriture binaire sur n bits.



Le circuit de l'oracle est composé de deux **registres**, en entrée le premier registre reçoit l'entier k , codé sur n bits, donc à l'aide de n lignes quantiques, même chose pour le second registre qui correspond à 0. Nous avons également deux registres en sortie, le premier renvoie k et le second $a^k \pmod N$. L'oracle a bien pour action $(k, 0) \mapsto (k, a^k \pmod N)$. En termes de qubits, si l'entrée de l'oracle est $|\underline{k}\rangle \otimes |\underline{0}\rangle$ alors la sortie $|\underline{k}\rangle \otimes |\underline{a^k \pmod N}\rangle$.

2.2. Début du circuit



- **Initialisation.** Le circuit est initialisé par des qubits tous égaux à $|0\rangle$.

$$|\psi_0\rangle = |0 \dots 0\rangle \otimes |0 \dots 0\rangle = |0\rangle^{\otimes n} \otimes |0\rangle^{\otimes n} = |\underline{0}\rangle \otimes |\underline{0}\rangle.$$

- **Transformation de Hadamard.** On applique une transformation de Hadamard, mais seulement sur le premier registre (donc sur les n premières lignes).

$$|\psi_1\rangle = H^{\otimes n} (|\underline{0}\rangle) \otimes |\underline{0}\rangle = \left(\frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} |\underline{k}\rangle \right) \otimes |\underline{0}\rangle = |\psi_H\rangle \otimes |\underline{0}\rangle.$$

- **Oracle.** On a vu que l'oracle envoie $|\underline{k}\rangle \otimes |\underline{0}\rangle$ sur $|\underline{k}\rangle \otimes |\underline{a^k}\rangle$, donc

$$|\psi_2\rangle = \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} |\underline{k}\rangle \otimes |\underline{a^k}\rangle.$$

Remarque. Les calculs dans le second registre se font modulo N . On raccourcit l'écriture $|\underline{a^k \pmod N}\rangle$ en $|\underline{a^k}\rangle$.

Nous allons maintenant récrire $|\psi_2\rangle$ en utilisant le fait que la fonction $k \mapsto a^k \pmod N$ est périodique de période r .

Hypothèse 3. L'ordre r divise 2^n .

C'est une hypothèse qui sert à simplifier la suite des calculs. Contrairement aux hypothèses 1 et 2, ce n'est pas une hypothèse vraie en général. Quand cette hypothèse est fautive les calculs qui suivent doivent être adaptés et sont un petit peu plus compliqués, mais le principe reste le même (voir le chapitre suivant).
 Sous l'hypothèse 3, pour $0 \leq k < 2^n$, écrivons la division euclidienne de k par r :

$$k = ar + \beta \quad \text{avec } 0 \leq \alpha < \frac{2^n}{r} \text{ et } 0 \leq \beta < r.$$

Ainsi le qubit $|k\rangle \otimes |a^k\rangle$ s'écrit aussi $|\alpha r + \beta\rangle \otimes |a^\beta\rangle$ car on rappelle que modulo N :

$$a^k = a^{ar+\beta} = a^{ar} \cdot a^\beta = (a^r)^\alpha \cdot a^\beta \equiv 1^\alpha \cdot a^\beta \equiv a^\beta \pmod N.$$

Ainsi :

$$|\psi_2\rangle = \frac{1}{\sqrt{2^n}} \sum_{\beta=0}^{r-1} \sum_{\alpha=0}^{2^n/r-1} |\alpha r + \beta\rangle \otimes |a^{\alpha r + \beta}\rangle = \frac{1}{\sqrt{2^n}} \sum_{\beta=0}^{r-1} \left(\sum_{\alpha=0}^{2^n/r-1} |\alpha r + \beta\rangle \right) \otimes |a^\beta\rangle.$$

2.3. Mesure du second registre

Après l'oracle, on effectue une mesure du second registre, c'est-à-dire des n dernières lignes du circuit. On obtient pour ce second registre la mesure d'un $|a^{\beta_0}\rangle$ pour un certain entier $0 \leq \beta_0 < r$. Après cette mesure le qubit du premier registre est :

$$|\bar{\psi}_3\rangle = \frac{\sqrt{r}}{\sqrt{2^n}} \sum_{\alpha=0}^{2^n/r-1} |\alpha r + \beta_0\rangle.$$

Le qubit complet en sortie de circuit est donc

$$|\psi_3\rangle = |\bar{\psi}_3\rangle \otimes |a^{\beta_0}\rangle$$

(en supposant ici que la mesure du second registre $|a^{\beta_0}\rangle$, fige le second registre en le n -qubit $|a^{\beta_0}\rangle$).

2.4. Que donnerait la mesure du premier registre ?

Le qubit $|\bar{\psi}_3\rangle$ est une somme de $|\alpha r + \beta_0\rangle$, $\alpha = 0, \dots, \frac{2^n}{r} - 1$. Si on mesurait ensuite le premier registre, c'est-à-dire le qubit $|\bar{\psi}_3\rangle$ alors on obtiendrait l'un des états

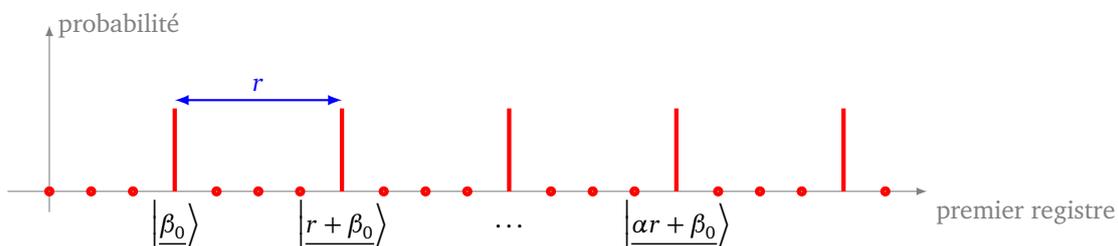
$$|\alpha r + \beta_0\rangle \quad \text{pour un certain } 0 \leq \alpha < 2^n/r,$$

ou plus exactement, un n -bit $\alpha r + \beta_0$, c'est-à-dire l'un des entiers

$$\alpha r + \beta_0 \quad \text{pour un certain } 0 \leq \alpha < 2^n/r.$$

En plus, ces entiers $\alpha r + \beta_0$ sont tous équiprobables.

Voici schématiquement ce que donne la mesure du premier registre : parmi toutes les possibilités $|0\rangle$ à $|2^n - 1\rangle$, la mesure donne l'un des $\alpha r + \beta_0$ où β_0 est un entier fixé (donné par la mesure du second registre) et r est la période.



Le but est de trouver r , mais on ne connaît ni α , ni β_0 , donc connaître l'un des $\alpha r + \beta_0$, ne permet pas de retrouver r .

Par exemple, pour $N = 15$, $\beta_0 = 1$, la mesure donne l'un des entiers 1, 5, 9 ou 13. L'écart entre ces entiers donne la période cherchée $r = 4$, mais la connaissance d'un seul de ces entiers ne permet pas de retrouver r . Malheureusement on ne peut pas refaire l'expérience pour obtenir un autre entier de la liste, car lors de la nouvelle expérience on n'obtiendra pas nécessairement le même β_0 .

Il va falloir compléter le circuit pour obtenir r . Cela va nous demander pas mal d'efforts et tout le reste de ce chapitre.

2.5. Exemple de $N = 15$

Essayons de factoriser $N = 15$. On prend alors $n = 4$, car $2^4 = 16 \geq N$. Le circuit est donc composé de deux registres de 4-bits (donc 8 lignes en tout).

On fixe un entier a premier avec N . Pour un exemple concret on prendra $a = 2$.

- **Initialisation.**

$$|\psi_0\rangle = |0.0.0.0\rangle \otimes |0.0.0.0\rangle = |\underline{0}\rangle \otimes |\underline{0}\rangle.$$

- **Transformation de Hadamard.**

$$|\psi_1\rangle = H^{\otimes 4} |\underline{0}\rangle \otimes |\underline{0}\rangle = \frac{1}{4} (|\underline{0}\rangle + |\underline{1}\rangle + |\underline{2}\rangle + \dots + |\underline{15}\rangle) \otimes |\underline{0}\rangle.$$

- **Oracle.**

$$|\psi_2\rangle = \frac{1}{4} (|\underline{0}\rangle \cdot |\underline{a^0}\rangle + |\underline{1}\rangle \cdot |\underline{a^1}\rangle + |\underline{2}\rangle \cdot |\underline{a^2}\rangle + \dots + |\underline{15}\rangle \cdot |\underline{a^{15}}\rangle)$$

Souvenons-nous que les termes des seconds facteurs sont calculés modulo N . Considérons le choix de $a = 2$, alors l'ordre que l'on veut déterminer est $r = 4$:

$$\begin{aligned} 2^0 &\equiv 1 \pmod{15} & 2^1 &\equiv 2 \pmod{15} & 2^2 &\equiv 4 \pmod{15} & 2^3 &\equiv 8 \pmod{15} \\ 2^4 &\equiv 16 \equiv 1 \pmod{15} & 2^5 &\equiv 32 \equiv 2 \pmod{15} & \dots & & & \end{aligned}$$

Donc

$$\begin{aligned} |\psi_2\rangle &= \frac{1}{4} (|\underline{0}\rangle |\underline{1}\rangle + |\underline{1}\rangle |\underline{2}\rangle + |\underline{2}\rangle |\underline{4}\rangle + |\underline{3}\rangle |\underline{8}\rangle \\ &\quad + |\underline{4}\rangle |\underline{1}\rangle + |\underline{5}\rangle |\underline{2}\rangle + |\underline{6}\rangle |\underline{4}\rangle + |\underline{7}\rangle |\underline{8}\rangle \\ &\quad + |\underline{8}\rangle |\underline{1}\rangle + |\underline{9}\rangle |\underline{2}\rangle + |\underline{10}\rangle |\underline{4}\rangle + |\underline{11}\rangle |\underline{8}\rangle \\ &\quad + |\underline{12}\rangle |\underline{1}\rangle + |\underline{13}\rangle |\underline{2}\rangle + |\underline{14}\rangle |\underline{4}\rangle + |\underline{15}\rangle |\underline{8}\rangle) \end{aligned}$$

On peut regrouper les termes selon le second facteur :

$$\begin{aligned} |\psi_2\rangle &= \frac{1}{4} (|\underline{0}\rangle + |\underline{4}\rangle + |\underline{8}\rangle + |\underline{12}\rangle) |\underline{1}\rangle \\ &\quad + \frac{1}{4} (|\underline{1}\rangle + |\underline{5}\rangle + |\underline{9}\rangle + |\underline{13}\rangle) |\underline{2}\rangle \\ &\quad + \frac{1}{4} (|\underline{2}\rangle + |\underline{6}\rangle + |\underline{10}\rangle + |\underline{14}\rangle) |\underline{4}\rangle \\ &\quad + \frac{1}{4} (|\underline{3}\rangle + |\underline{7}\rangle + |\underline{11}\rangle + |\underline{15}\rangle) |\underline{8}\rangle \end{aligned}$$

- **Mesure du second registre.**

Une mesure sur le second registre renvoie de façon équiprobable :

$$\underline{1} \quad \text{ou} \quad \underline{2} \quad \text{ou} \quad \underline{4} \quad \text{ou} \quad \underline{8}.$$

Le qubit $|\bar{\psi}_3\rangle$ du premier registre dépend alors de cette mesure :

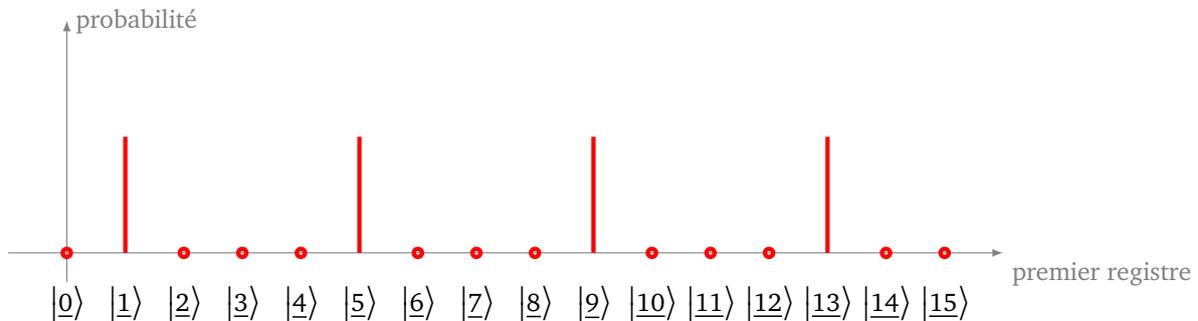
- si la mesure du second registre est $\underline{1}$ alors $|\bar{\psi}_3\rangle = \frac{1}{2} (|\underline{0}\rangle + |\underline{4}\rangle + |\underline{8}\rangle + |\underline{12}\rangle)$,
- si la mesure du second registre est $\underline{2}$ alors $|\bar{\psi}_3\rangle = \frac{1}{2} (|\underline{1}\rangle + |\underline{5}\rangle + |\underline{9}\rangle + |\underline{13}\rangle)$,

- si la mesure du second registre est 4 alors $|\bar{\psi}_3\rangle = \frac{1}{2}(|\underline{2}\rangle + |\underline{6}\rangle + |\underline{10}\rangle + |\underline{14}\rangle)$,
- si la mesure du second registre est 8 alors $|\bar{\psi}_3\rangle = \frac{1}{2}(|\underline{3}\rangle + |\underline{7}\rangle + |\underline{11}\rangle + |\underline{15}\rangle)$.

• **Mesure du premier registre.**

On effectue ensuite une mesure sur le premier registre. Par exemple, plaçons-nous dans le cas où le second registre a donné la mesure 2, alors le qubit $|\bar{\psi}_3\rangle = \frac{1}{2}(|\underline{1}\rangle + |\underline{5}\rangle + |\underline{9}\rangle + |\underline{13}\rangle)$ se mesure de façon équiprobable en :

$$\underline{1} \text{ ou } \underline{5} \text{ ou } \underline{9} \text{ ou } \underline{13}.$$



La mesure donne donc un entier parmi 1, 5, 9, 13, qui ont un écart entre eux de $r = 4$ (la période que l'on veut trouver), mais comme on n'obtient qu'un seul de ces entiers cela ne permet pas de retrouver ce r .

Autre exemple : si le second registre a donné la mesure 8, alors la mesure du premier registre donne l'un des entiers 3, 7, 11 ou bien 15, mais ne permet pas de retrouver r .

3. Transformée de Fourier discrète

3.1. Préambule sur les nombres complexes

Rappelons quelques résultats sur l'écriture trigonométrique des nombres complexes :

- tout nombre complexe $z \in \mathbb{C}$ s'écrit $z = r e^{i\theta}$ où $r \geq 0$ et $\theta \in \mathbb{R}$,
- un nombre complexe de module 1 s'écrit $z = e^{i\theta}$ où $\theta \in \mathbb{R}$,
- $e^{2i\pi} = 1$,
- $(e^{i\theta})^* = e^{-i\theta}$ où z^* désigne le conjugué de z .

Lemme 2 (Somme d'une suite géométrique).

Soit $z \in \mathbb{C}$. Soit $n \geq 0$. Alors

$$1 + z + z^2 + \dots + z^{n-1} = \begin{cases} n & \text{si } z = 1, \\ \frac{1-z^n}{1-z} & \text{sinon.} \end{cases}$$

Démonstration. Notons $S_n = \sum_{k=0}^{n-1} z^k$ la somme à calculer. Si $z = 1$ alors S_n est la somme de n termes égaux à 1. Sinon en développant $(1-z) \cdot S_n = S_n - z \cdot S_n = 1 - z^n$ car presque tous les termes se télescopent. \square

Le lemme suivant est le point-clé de la transformée de Fourier discrète que l'on étudiera plus loin.

Lemme 3 (Lemme crucial).

Soient $n \in \mathbb{N}^*$ et $j \in \mathbb{Z}$.

$$\frac{1}{n} \sum_{k=0}^{n-1} e^{2i\pi \frac{kj}{n}} = \begin{cases} 1 & \text{si } \frac{j}{n} \text{ est un entier,} \\ 0 & \text{sinon.} \end{cases}$$

Noter que si on pose $\omega = e^{\frac{2i\pi}{n}}$ et $z = \omega^j = e^{\frac{2i\pi j}{n}}$ alors la somme à calculer est simplement $\Sigma_n = \frac{1}{n} \sum_{k=0}^{n-1} z^k$.

Démonstration. Par la remarque précédente il s'agit de calculer la somme d'une suite géométrique (au facteur $\frac{1}{n}$ près) : $\Sigma_n = \frac{1}{n} \sum_{k=0}^{n-1} z^k$.

Si j/n est un entier alors $z = e^{\frac{2i\pi j}{n}} = e^{2i\pi} = 1$ et par le premier cas du lemme 2, alors $\Sigma_n = 1$.

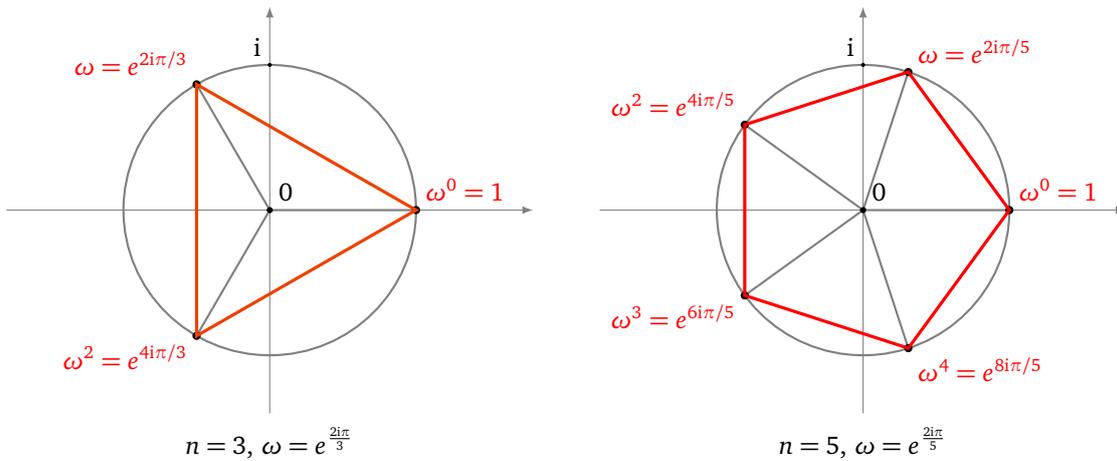
Sinon, $z \neq 1$ et à l'aide du second cas du lemme 2 :

$$\Sigma_n = \frac{1}{n} \sum_{k=0}^{n-1} z^k = \frac{1}{n} \cdot \frac{1 - z^n}{1 - z}.$$

Mais $z^n = \left(e^{\frac{2i\pi j}{n}}\right)^n = e^{2i\pi j} = 1$ et ainsi $\Sigma_n = 0$.

□

Voici l'interprétation géométrique de ce lemme. Notons de nouveau $\omega = e^{\frac{2i\pi}{n}}$, c'est une racine n -ième de l'unité. Alors les ω^k forment les sommets d'un polygone régulier à n côtés. Le barycentre de ces points a pour coordonnées $\frac{1}{n} \sum_{k=0}^{n-1} \omega^k = 0$ par le lemme, c'est donc bien l'origine ! D'un point de vue physique on parle d'interférence destructive.



3.2. Transformée de Fourier

Fixons un entier $n \geq 1$. La **transformée de Fourier discrète** \hat{F} transforme un n -qubit de base en une somme de n -qubits de base selon la formule :

$$\hat{F} |\underline{k}\rangle = \frac{1}{\sqrt{2^n}} \sum_{j=0}^{2^n-1} e^{2i\pi \frac{kj}{2^n}} |j\rangle$$

Si on note $\omega = e^{\frac{2i\pi}{2^n}}$ alors la formule s'écrit aussi :

$$\hat{F} |\underline{k}\rangle = \frac{1}{\sqrt{2^n}} \sum_{j=0}^{2^n-1} (\omega^k)^j |j\rangle$$

Ensuite \hat{F} est étendue par linéarité à n'importe quel n -qubit $|\psi\rangle$. Si $|\psi\rangle = \sum_{k=0}^{2^n-1} \alpha_k |\underline{k}\rangle$ alors

$$\hat{F} |\psi\rangle = \sum_{k=0}^{2^n-1} \alpha_k \hat{F} |\underline{k}\rangle.$$

Commençons par des exemples avec de petites valeurs de n .

Exemple.

Fixons $n = 1$. Les deux 1-qubits de base sont $|0\rangle$ et $|1\rangle$. On a alors $2^n = 2$ et $\omega = e^{\frac{2i\pi}{2}} = e^{i\pi} = -1$.

Pour $k = 0$, les coefficients seront $(\omega^0)^0 = 1$ et $(\omega^0)^1 = 1$, donc :

$$\hat{F} |0\rangle = \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle).$$

Pour $k = 1$, les coefficients seront $(\omega^1)^0 = 1$ et $(\omega^1)^1 = -1$, donc :

$$\hat{F} |1\rangle = \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle).$$

Résumons les coefficients par le tableau des valeurs $(\omega^k)^j$, avec ici $\omega = -1$.

	$j = 0$	$j = 1$
$k = 0$	$(\omega^0)^0$	$(\omega^0)^1$
$k = 1$	$(\omega^1)^0$	$(\omega^1)^1$

	$j = 0$	$j = 1$
$k = 0$	1	1
$k = 1$	1	-1

Pour un qubit $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$ alors

$$\begin{aligned} \hat{F} |\psi\rangle &= \hat{F}(\alpha |0\rangle + \beta |1\rangle) = \alpha \hat{F} |0\rangle + \beta \hat{F} |1\rangle \\ &= \frac{1}{\sqrt{2}} \alpha (|0\rangle + |1\rangle) + \frac{1}{\sqrt{2}} \beta (|0\rangle - |1\rangle) \\ &= \frac{1}{\sqrt{2}} (\alpha + \beta) |0\rangle + \frac{1}{\sqrt{2}} (\alpha - \beta) |1\rangle \end{aligned}$$

Noter qu'au final pour $n = 1$, \hat{F} est égale à la porte de Hadamard H .

Exemple.

Fixons $n = 2$. Les 2-qubits de base sont $|\underline{0}\rangle = |0.0\rangle$, $|\underline{1}\rangle = |0.1\rangle$, $|\underline{2}\rangle = |1.0\rangle$ et $|\underline{3}\rangle = |1.1\rangle$. On a alors $2^n = 4$ et $\omega = e^{\frac{2i\pi}{4}} = e^{i\frac{\pi}{2}} = i$.

Les coefficients de $\hat{F} |\underline{k}\rangle$ sont les $(\omega^k)^j$.

- Pour $k = 0$, les coefficients sont tous 1,
- Pour $k = 1$, les coefficients sont les i^j ,
- Pour $k = 2$, les coefficients sont les $(-1)^j$,
- Pour $k = 3$, les coefficients sont les $(-i)^j$.

Voici le tableau des coefficients $(\omega^k)^j$ avec $\omega = i$.

	$j = 0$	$j = 1$	$j = 2$	$j = 3$
$k = 0$	$(\omega^0)^0$	$(\omega^0)^1$	$(\omega^0)^2$	$(\omega^0)^3$
$k = 1$	$(\omega^1)^0$	$(\omega^1)^1$	$(\omega^1)^2$	$(\omega^1)^3$
$k = 2$	$(\omega^2)^0$	$(\omega^2)^1$	$(\omega^2)^2$	$(\omega^2)^3$
$k = 3$	$(\omega^3)^0$	$(\omega^3)^1$	$(\omega^3)^2$	$(\omega^3)^3$

	$j = 0$	$j = 1$	$j = 2$	$j = 3$
$k = 0$	1	1	1	1
$k = 1$	1	i	-1	-i
$k = 2$	1	-1	1	-1
$k = 3$	1	-i	-1	i

Ainsi :

$$\begin{aligned} \hat{F} |\underline{0}\rangle &= \frac{1}{2} (|\underline{0}\rangle + |\underline{1}\rangle + |\underline{2}\rangle + |\underline{3}\rangle) \\ \hat{F} |\underline{1}\rangle &= \frac{1}{2} (|\underline{0}\rangle + i|\underline{1}\rangle - |\underline{2}\rangle - i|\underline{3}\rangle) \\ \hat{F} |\underline{2}\rangle &= \frac{1}{2} (|\underline{0}\rangle - |\underline{1}\rangle + |\underline{2}\rangle - |\underline{3}\rangle) \\ \hat{F} |\underline{3}\rangle &= \frac{1}{2} (|\underline{0}\rangle - i|\underline{1}\rangle - |\underline{2}\rangle + i|\underline{3}\rangle) \end{aligned}$$

De façon générale pour n quelconque et $\omega = e^{\frac{2i\pi}{2^n}}$, alors voici comment s'organise le tableau des $(\omega^k)^j$.

	$j = 0$	$j = 1$	$j = 2$	$j = 3$	\dots
$k = 0$	1	1	1	1	\dots
$k = 1$	1	ω	ω^2	ω^3	\dots
$k = 2$	1	ω^2	$(\omega^2)^2$	$(\omega^2)^3$	\dots
$k = 3$	1	ω^3	$(\omega^3)^2$	$(\omega^3)^3$	\dots
$k = 4$	1	ω^4	\dots		
\dots	\dots	\dots			

3.3. La transformée de Fourier discrète est unitaire

Proposition 4.

La transformation de Fourier discrète est une application unitaire.

Commençons par le vérifier sur des exemples.

- Pour $n = 1$, notons $|\psi_0\rangle = \hat{F}|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ et $|\psi_1\rangle = \hat{F}|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. Alors la base $(|0\rangle, |1\rangle)$ est envoyée sur la base $(|\psi_0\rangle, |\psi_1\rangle)$. Comme une base orthonormale est envoyée sur une base orthonormale, \hat{F} est une transformation unitaire.
- Pour $n = 2$, notons de même $|\psi_k\rangle = \hat{F}|k\rangle$ pour $k = 0, 1, 2, 3$ que l'on a calculé auparavant. On vérifie que les $|\psi_k\rangle$ forment une base orthonormale. Par exemple, montrons que $|\psi_1\rangle$ et $|\psi_2\rangle$ sont orthogonaux. Tout d'abord

$$\langle \psi_1 | \psi_1 \rangle = |\psi_1\rangle^* = \frac{1}{2} (\langle 0 | - i \langle 1 | - \langle 2 | + i \langle 3 |)$$

Donc :

$$\langle \psi_1 | \psi_2 \rangle = \frac{1}{4} (\langle 0 | - i \langle 1 | - \langle 2 | + i \langle 3 |) \cdot (|0\rangle - |1\rangle + |2\rangle - |3\rangle).$$

On développe tout, et on utilise que $\langle p | q \rangle = 0$ si $p \neq q$ et $\langle p | p \rangle = 1$:

$$\langle \psi_1 | \psi_2 \rangle = \frac{1}{4} (\langle 0 | 0 \rangle + i \langle 1 | 1 \rangle - \langle 2 | 2 \rangle - i \langle 3 | 3 \rangle) = \frac{1}{4} (1 + i - 1 - i) = 0.$$

Démonstration. Nous allons montrer que la base canonique des $|k\rangle$ s'envoie sur une base orthonormale $|\psi_k\rangle$. Ainsi \hat{F} envoie une base orthonormale sur une base orthonormale et est donc une transformation unitaire (voir le chapitre « Portes quantiques »).

Notons $|\psi_k\rangle = \hat{F}|k\rangle$, pour $0 \leq k \leq 2^n - 1$.

$$\begin{aligned} \langle \psi_k | \psi_\ell \rangle &= (\hat{F}|k\rangle)^* \cdot \hat{F}|l\rangle \\ &= \left(\frac{1}{\sqrt{2^n}} \sum_{p=0}^{2^n-1} e^{-2i\pi \frac{kp}{2^n}} \langle p | \right) \cdot \left(\frac{1}{\sqrt{2^n}} \sum_{q=0}^{2^n-1} e^{2i\pi \frac{\ell q}{2^n}} |q\rangle \right) \\ &= \frac{1}{2^n} \sum_{p=0}^{2^n-1} \sum_{q=0}^{2^n-1} e^{-2i\pi \frac{kp}{2^n}} e^{2i\pi \frac{\ell q}{2^n}} \langle p | q \rangle \\ &= \frac{1}{2^n} \sum_{p=0}^{2^n-1} e^{-2i\pi \frac{kp}{2^n} + 2i\pi \frac{\ell p}{2^n}} \langle p | p \rangle \quad \text{car } \langle p | q \rangle = 0 \text{ si } p \neq q \\ &= \frac{1}{2^n} \sum_{p=0}^{2^n-1} e^{2i\pi \frac{(\ell-k)p}{2^n}} \quad \text{car } \langle p | p \rangle = 1 \end{aligned}$$

Si $k = \ell$ alors $e^{2i\pi \frac{(\ell-k)p}{2^n}} = e^0 = 1$, et ainsi $\langle \psi_k | \psi_k \rangle = 1$. Si $k \neq \ell$, alors $\frac{\ell-k}{2^n}$ n'est pas un entier, et par le lemme crucial 3, $\langle \psi_k | \psi_\ell \rangle = 0$. Ainsi les $|\psi_k\rangle$ forment une base orthonormale et \hat{F} est une transformation unitaire. \square

3.4. Transformée de Fourier inverse

Comme \hat{F} est unitaire alors \hat{F} est inversible et $\hat{F}^{-1} = \hat{F}^*$. Ainsi la formule de $\hat{F}^{-1} |k\rangle$ est celle de $\hat{F} |k\rangle$ mais avec un signe moins dans l'exponentielle :

$$\hat{F}^{-1} |k\rangle = \frac{1}{\sqrt{2^n}} \sum_{j=0}^{2^n-1} e^{-2i\pi \frac{kj}{2^n}} |j\rangle$$

Si on note $\omega = e^{\frac{2i\pi}{2^n}}$, alors $\omega^* = e^{-\frac{2i\pi}{2^n}}$ et ainsi

$$\hat{F}^{-1} |k\rangle = \frac{1}{\sqrt{2^n}} \sum_{j=0}^{2^n-1} (\omega^{*k})^j |j\rangle.$$

Exemple.

Fixons $n = 1$, $\omega^* = -1$.

$$\hat{F}^{-1} |0\rangle = \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle) \quad \text{et} \quad \hat{F}^{-1} |1\rangle = \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle)$$

Dans ce cas $\hat{F}^{-1} = \hat{F}$.

Exemple.

Fixons $n = 2$, $\omega^* = -i$.

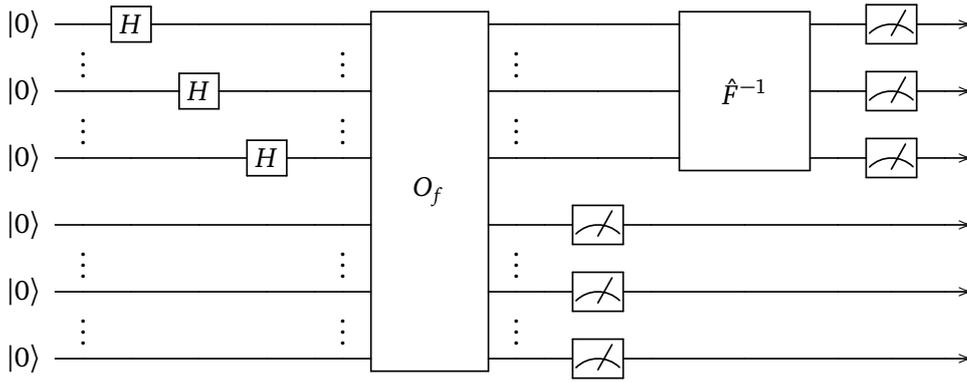
$$\begin{aligned} \hat{F}^{-1} |0\rangle &= \frac{1}{2} (|0\rangle + |1\rangle + |2\rangle + |3\rangle) \\ \hat{F}^{-1} |1\rangle &= \frac{1}{2} (|0\rangle - i|1\rangle - |2\rangle + i|3\rangle) \\ \hat{F}^{-1} |2\rangle &= \frac{1}{2} (|0\rangle - |1\rangle + |2\rangle - |3\rangle) \\ \hat{F}^{-1} |3\rangle &= \frac{1}{2} (|0\rangle + i|1\rangle - |2\rangle - i|3\rangle) \end{aligned}$$

Exercice. Vérifier à la main que $\hat{F}^{-1}(\hat{F} |1\rangle) = |1\rangle$.

4. Fin de l'algorithme de Shor

4.1. Fin du circuit

Après l'oracle, nous en étions restés à une mesure du second registre, et nous avons vu que la mesure du premier registre ne permettait pas de conclure. Après la mesure du second registre, nous allons faire agir sur le premier registre la transformée de Fourier discrète inverse \hat{F}^{-1} .



4.2. Calculs

On se souvient que la mesure du second registre a donné a^{β_0} (on peut aussi considérer que l'état quantique du second registre s'est effondré à $|a^{\beta_0}\rangle$). Alors le qubit du premier registre est :

$$|\bar{\psi}_3\rangle = \frac{\sqrt{r}}{\sqrt{2^n}} \sum_{\alpha=0}^{2^n/r-1} |\alpha r + \beta_0\rangle.$$

Calculons le qubit $|\bar{\psi}_4\rangle$ obtenu après l'action de la transformée de Fourier inverse :

$$\begin{aligned} |\bar{\psi}_4\rangle &= \hat{F}^{-1} |\bar{\psi}_3\rangle \\ &= \hat{F}^{-1} \left(\frac{\sqrt{r}}{\sqrt{2^n}} \sum_{\alpha=0}^{2^n/r-1} |\alpha r + \beta_0\rangle \right) \\ &= \frac{\sqrt{r}}{\sqrt{2^n}} \sum_{\alpha=0}^{2^n/r-1} \hat{F}^{-1} |\alpha r + \beta_0\rangle \\ &= \frac{\sqrt{r}}{\sqrt{2^n}} \sum_{\alpha=0}^{2^n/r-1} \frac{1}{\sqrt{2^n}} \sum_{j=0}^{2^n-1} e^{-2i\pi \frac{(\alpha r + \beta_0)j}{2^n}} |j\rangle \\ &= \frac{\sqrt{r}}{2^n} \sum_{j=0}^{2^n-1} \left(\sum_{\alpha=0}^{2^n/r-1} e^{-2i\pi \frac{\alpha j}{2^n/r}} \right) e^{-2i\pi \frac{\beta_0 j}{2^n}} |j\rangle \end{aligned}$$

Pour la dernière égalité nous avons interverti les deux sommes et utilisé que r divise 2^n . Calculons le coefficient qui intervient dans cette somme à l'aide du lemme crucial 3 :

$$\frac{1}{2^n/r} \sum_{\alpha=0}^{2^n/r-1} e^{-2i\pi \frac{\alpha j}{2^n/r}} = \begin{cases} 1 & \text{si } \frac{j}{2^n/r} \text{ est un entier,} \\ 0 & \text{sinon.} \end{cases}$$

Ainsi

$$|\bar{\psi}_4\rangle = \frac{1}{\sqrt{r}} \sum_{\substack{j=0, \dots, 2^n-1 \\ \text{avec } \frac{j}{2^n/r} \text{ entier}}} e^{-2i\pi \frac{\beta_0 j}{2^n}} |j\rangle = \frac{1}{\sqrt{r}} \sum_{\ell=0}^{r-1} e^{-2i\pi \beta_0 \frac{\ell}{r}} \left| \frac{2^n \ell}{r} \right\rangle.$$

Pour la dernière égalité, nous avons juste changé la notation des indices en posant $j = \frac{2^n \ell}{r}$.

Qu'avons-nous gagné avec l'action de la transformée de Fourier ? Tout d'abord la constante β_0 n'apparaît que dans les coefficients des qubits et n'intervient plus après mesure. Ensuite la période r que l'on veut déterminer est au dénominateur dans l'expression de l'entier $\frac{2^n \ell}{r}$.

4.3. Mesure du premier registre

La mesure du premier registre donne un entier $\frac{2^n \ell}{r}$, correspondant à l'un des états $\left| \frac{2^n \ell}{r} \right\rangle$ du qubit $|\bar{\psi}_4\rangle$.

Nous obtenons donc un entier $m = \frac{2^n \ell}{r}$ et nous voulons en déduire la période r . L'entier n est connu, par contre ℓ n'est pas connu (on sait $0 \leq \ell \leq r - 1$).

Commençons par diviser par 2^n (valeur connue) pour obtenir le rationnel $x = \frac{m}{2^n} = \frac{\ell}{r}$.

- Si x est entier alors on n'obtient aucune information sur r . C'est le cas si $\ell = 0$, ou bien si r divise ℓ . Dans ce cas notre méthode échoue. Il faut recommencer l'exécution du circuit quantique. Noter que ce cas se produit assez rarement.
- Si $\text{pgcd}(\ell, r) = 1$ alors d'une part $x = \frac{m}{2^n}$ est connu et son écriture sous la forme de fraction irréductible est $\frac{\ell}{r}$. Donc en réduisant la fraction $x = \frac{m}{2^n}$ en une fraction irréductible, on obtient r (et ℓ). Par exemple si $x = \frac{26}{8}$ alors l'écriture irréductible est $\frac{13}{4}$ donc $r = 4$ et $\ell = 13$.
- Si $\text{pgcd}(\ell, r) \neq 1$, alors l'écriture irréductible de $x = \frac{m}{2^n}$ est $\frac{\ell'}{r'}$. On a

$$x = \frac{m}{2^n} = \frac{\ell'}{r'} = \frac{\ell}{r}.$$

Ainsi $r'\ell = r\ell'$, donc r' divise $r\ell'$, mais comme $\text{pgcd}(r', \ell') = 1$ alors par le lemme de Gauss r' divise r . Nous n'avons pas trouvé la période r mais un facteur r' de r . C'est un progrès ! On recommence notre algorithme avec le choix de $a^{r'}$ au lieu de a . En effet, comme la fonction $a \mapsto a^k$ est de période r , alors la fonction $k \mapsto (a^{r'})^k$ est de période r/r' . Nous sommes certains que ce processus se termine car r n'a qu'un nombre fini de facteurs.

4.4. Exemple

Voyons la fin de l'algorithme sur l'exemple $N = 15$.

- Si $a \in \{3, 5, 6, 9, 10, 12\}$, alors a n'est pas premier avec N et $d = \text{pgcd}(a, N) > 1$ est un diviseur strict de N et c'est terminé.
- Les entiers $a \in \{4, 11, 14\}$ sont d'ordre $r = 2$, mais par contre $a = 14$ ne vérifie pas l'hypothèse 2. Étudions les cas $a = 4$ et $a = 11$. La mesure finale du circuit fournit $m = \frac{2^n \ell}{r}$, on calcule $x = \frac{m}{2^n} = \frac{\ell}{r}$ avec $0 \leq \ell \leq r - 1$, donc ici avec $r = 2$, $x = \frac{\ell}{2}$ avec $\ell = 0$ ou $\ell = 1$. On rappelle que l'on connaît la valeur de x , mais qu'il s'agit d'obtenir r et ℓ .
 - Dans le cas $\ell = 0$ alors on a la connaissance de $x = 0$, mais on n'obtient aucune information sur r . Il faut recommencer l'algorithme.
 - Dans le cas $\ell = 1$ alors on a la connaissance de $x = \frac{1}{2}$. Comme $x = \frac{\ell}{r}$ est une fraction irréductible, la connaissance de x permet de retrouver $\ell = 1$ et $r = 2$. Nous avons obtenu la période $r = 2$.
- Les entiers $a \in \{2, 7, 8, 13\}$ sont d'ordre $r = 4$. La mesure finale du circuit fournit $m = \frac{2^n \ell}{r}$, on calcule $x = \frac{m}{2^n} = \frac{\ell}{r}$ avec $0 \leq \ell \leq r - 1$, donc ici avec $r = 4$, $x = \frac{\ell}{4}$ avec $\ell \in \{0, 1, 2, 3\}$.
 - Dans le cas $\ell = 0$, la connaissance de $x = 0$ ne permet pas de trouver r . Il faut recommencer l'algorithme.
 - Dans le cas $\ell = 1$, on a $x = \frac{1}{4}$. Comme $x = \frac{\ell}{r}$ est une fraction irréductible, la connaissance de x permet de retrouver $\ell = 1$ et $r = 4$. Nous avons obtenu la période $r = 4$.
 - Dans le cas $\ell = 2$ on a $x = \frac{1}{2}$. Nous obtenons la fraction irréductible $x = \frac{\ell'}{r'}$ avec $\ell' = 1$ et $r' = 2$. L'entier $r' = 2$ n'est pas la période (c'est facile à vérifier) mais on sait que $r' = 2$ divise r . On avait choisi un entier a au début de l'algorithme, on recommence maintenant l'algorithme avec le choix de a^2 qui a pour période $r/2$. En un nombre fini d'itérations de l'algorithme on obtiendra r .
 - Dans le cas $\ell = 3$, la connaissance de $x = \frac{3}{4}$ permet de retrouver $\ell = 3$ et $r = 4$.

Nous verrons dans le chapitre suivant un exemple dans lequel l'ordre r n'est pas une puissance de 2.

Compléments d'arithmétique

Vidéo ■ partie 14.1. Fractions continues

Vidéo ■ partie 14.2. Algorithme de Shor pour n'importe quel ordre

Vidéo ■ partie 14.3. L'algorithme de Shor fonctionne au moins une fois sur deux

Nous apportons des compléments à l'algorithme de Shor vu lors du chapitre précédent en étudiant chacune des hypothèses.

Dans ce chapitre il n'y a pas d'informatique quantique mais beaucoup de mathématiques ! Certaines parties sont assez techniques et d'un niveau un peu plus élevé que les chapitres précédents.

1. Fractions continues

1.1. Motivation

Dans le chapitre précédent nous avons fait une hypothèse simplificatrice : l'ordre r divise 2^n . Ce n'est pas vrai en général, mais l'algorithme de Shor reste valide moyennant quelques adaptations.

Reprenons la fin du circuit de l'algorithme de Shor qui permet de calculer l'ordre r d'un élément.

- Si r divise 2^n alors la mesure du premier registre conduit à un nombre rationnel $x = \frac{m}{2^n}$ qui est aussi égal à $\frac{\ell}{r}$. Ainsi x est un multiple de $\frac{1}{r}$ et permet de retrouver r (ou au moins un facteur de r).
- Si r ne divise pas 2^n alors la mesure du premier registre conduit à un nombre rationnel $x = \frac{m}{2^n}$ qui est proche d'un multiple de $\frac{1}{r}$ (mais n'est pas exactement un multiple). Comment retrouver r à partir de x ?

Voici l'exemple que l'on étudiera en détails dans la section suivante afin de factoriser $N = 21$ à l'aide du choix $a = 2$. Imaginons qu'une mesure conduise à $x = \frac{427}{512}$. Comment retrouver l'ordre r ? On pourrait aussi obtenir $x = \frac{426}{512}$ ou bien $x = \frac{428}{512}$. On voit que x est proche de $\frac{4}{5}$. Mais est-ce que 5 est vraiment la période ?

1.2. Fractions continues

Une *fraction continue* est une fraction

$$a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{\dots + \frac{1}{a_n}}}}$$

où $a_0 \geq 0$ et $a_i > 0$ (pour $i > 0$). On note cette fraction par la liste $[a_0, a_1, \dots, a_n]$.

On note $\frac{p_n}{q_n}$ l'écriture irréductible du rationnel $[a_0, a_1, \dots, a_n]$.

Exemple.

L'écriture en fraction continue $[5, 2, 1, 4]$ représente le nombre rationnel :

$$x = 5 + \frac{1}{2 + \frac{1}{1 + \frac{1}{4}}} = \frac{75}{14} = 5.3571428\dots$$

Prendre les sous-listes de la fraction continue permet d'obtenir des approximations de x de plus en plus précises :

- Sous liste [5], alors $\frac{p_0}{q_0} = 5$.
- Sous liste [5, 2] alors $\frac{p_1}{q_1} = \frac{11}{2} = 5.5$.
- Sous liste [5, 2, 1] alors $\frac{p_2}{q_2} = \frac{16}{3} = 5.33\dots$
- Liste complète [5, 2, 1, 4] alors $\frac{p_3}{q_3} = \frac{75}{14} = x$.

1.3. Approximations par les fractions continues

Les fractions continues prennent tout leur intérêt pour approcher des réels (ou des rationnels) par des fractions simples. Prenons l'exemple de π . Comment approcher π par une fraction avec un dénominateur pas trop grand (disons avec moins de trois chiffres) ? L'idée la plus simple est d'utiliser l'écriture décimale $\pi = 3.1415\dots \simeq \frac{314}{100}$. Mais peut-on faire mieux ?

Calculons pour commencer la fraction continue de $x = \frac{314}{100}$. Cela se fait par des divisions euclidiennes successives : $314 = 3 \times 100 + 14$ donc

$$\frac{314}{100} = 3 + \frac{14}{100} = 3 + \frac{1}{\frac{100}{14}}$$

puis $100 = 7 \times 14 + 2$, donc

$$\frac{314}{100} = 3 + \frac{1}{7 + \frac{2}{14}} = 3 + \frac{1}{7 + \frac{1}{7}}$$

Ainsi

$$\frac{314}{100} \simeq 3 + \frac{1}{7} = \frac{22}{7}$$

Nous avons donc approché π par $\frac{22}{7} = 3.1428\dots$ ce qui est aussi bien que $\frac{314}{100}$ mais avec un dénominateur beaucoup plus petit.

Bien évidemment on peut pousser les calculs plus loin : $\pi \simeq \frac{314159}{100000}$. On calcule la fraction continue de π (ou de $\frac{314159}{100000}$) et on obtient $[3, 7, 15, 1, \dots]$. Cela fournit les approximations successives :

- Sous liste [3, 7], alors $\frac{p_1}{q_1} = \frac{22}{7} = 3.1428\dots$
- Sous liste [3, 7, 15], alors $\frac{p_2}{q_2} = \frac{333}{106} = 3.141509\dots$
- Sous liste [3, 7, 15, 1], alors $\frac{p_3}{q_3} = \frac{355}{113} = 3.14159292\dots$

Ainsi avec des fractions dont les dénominateurs restent petits, on trouve de très bonnes approximations de π . En un sens les fractions continues donnent les meilleures approximations possibles d'un réel x par des rationnels.

1.4. Exemple

Reprenons l'exemple de la factorisation de $N = 21$ à l'aide du choix $a = 2$. Supposons que le circuit de Shor nous donne la valeur $x = \frac{427}{512}$, comment obtenir l'ordre r ?

La mauvaise idée est d'utiliser l'écriture décimale pour dire $x = \frac{427}{512} \simeq \frac{400}{500} = \frac{4}{5}$ donc le dénominateur naturel (qui donne l'ordre r) serait 5. Ce n'est pas vrai.

La bonne méthode est de calculer le développement en fraction continue de x :

$$x = \frac{427}{512} = [0, 1, 5, 42, 2] = 0 + \frac{1}{1 + \frac{1}{5 + \frac{1}{42 + \frac{1}{2}}}}$$

Ce qui fournit les approximations successives :

- Sous liste [0, 1] = $0 + \frac{1}{1}$, alors $\frac{p_1}{q_1} = \frac{1}{1} = 1$.
- Sous liste [0, 1, 5] = $0 + \frac{1}{1 + \frac{1}{5}} = \frac{5}{6}$.

- Sous liste $[0, 1, 5, 42] = 0 + \frac{1}{1 + \frac{1}{5 + \frac{1}{42}}} = \frac{211}{253}$.
- Sous liste $[0, 1, 5, 42, 2] = \frac{427}{512}$.

Les dénominateurs sont les candidats pour l'ordre r , mais on sait que l'ordre r cherché est inférieur à l'entier $N = 21$. Donc ici, la meilleure fraction ayant un dénominateur inférieur à N est $\frac{5}{6}$, on trouve ainsi $r = 6$. Il est facile de vérifier que l'ordre de $a = 2$ modulo $N = 21$ est bien $r = 6$.

Lors de la mesure on peut aussi obtenir des valeurs légèrement différentes par exemple $x' = \frac{426}{512}$ ou bien $x'' = \frac{428}{512}$. Que se passe-t-il alors ?

- Si $x' = \frac{426}{512} = [0, 1, 4, 1, 20, 2]$, les fractions successives sont $\frac{1}{1}, \frac{4}{5}, \frac{5}{6}, \frac{104}{125}, \frac{213}{256}$. La meilleure fraction ayant un dénominateur inférieur à N est $\frac{5}{6}$, on retrouve ainsi $r = 6$.
- Si $x'' = \frac{428}{512} = [0, 1, 5, 10, 2]$, les fractions successives sont $\frac{1}{1}, \frac{5}{6}, \frac{51}{61}, \frac{107}{128}$. La meilleure fraction ayant un dénominateur inférieur à N est encore $\frac{5}{6}$ et on retrouve $r = 6$.

Conclusion : la méthode des fractions continues permet de retrouver l'ordre r .

2. Algorithme de Shor pour n'importe quel ordre pair

2.1. Fin du circuit

- **Cas du chapitre précédent.** Si r divise 2^n alors la mesure du premier registre conduit à la mesure d'un état $\left| \frac{2^n \ell}{r} \right\rangle$ et donne donc un entier $m = \frac{2^n \ell}{r}$. On définit alors le rationnel $x = \frac{m}{2^n}$ qui est aussi égal à $\frac{\ell}{r}$ et qui permet de retrouver r (ou au moins un facteur de r). Noter que comme r divise 2^n , m est un multiple de $\frac{2^n}{r}$. Autrement dit, x est un multiple (avec un facteur entier) de $\frac{1}{r}$.
- **Cas considéré maintenant.** Si r ne divise pas 2^n alors la mesure du premier registre conduit à un entier m . Cet entier m est proche de $\frac{2^n \ell}{r}$ (pour un certain entier ℓ) mais la fraction $\frac{2^n \ell}{r}$ n'est plus un entier. Autrement dit on obtient un nombre rationnel $x = \frac{m}{2^n}$ qui est proche d'un multiple de $\frac{1}{r}$ (mais n'est pas exactement un multiple).

2.2. L'exemple $N = 21$: début

Soit N l'entier à factoriser. Dans toute la suite on considérera l'exemple de $N = 21$. Dans le cas où r divise 2^n , il suffisait de choisir l'entier n tel que $N \leq 2^n$. Dans le cas général on choisit n de sorte à avoir les inégalités $N^2 \leq 2^n < 2N^2$. Pour $N = 21$, on a $N^2 = 441$, donc avec $n = 9$ on a bien $N^2 \leq 2^n = 512 < 2N^2$.

Reprenons les calculs du circuit de Shor :

- **Initialisation.**

$$|\psi_0\rangle = |0\rangle^{\otimes n} \otimes |0\rangle^{\otimes n}.$$

- **Transformation de Hadamard.**

$$|\psi_1\rangle = \left(\frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} |k\rangle \right) \otimes |0\rangle.$$

- **Oracle.**

$$|\psi_2\rangle = \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} |k\rangle \otimes |a^k\rangle.$$

2.3. L'exemple $N = 21$: milieu

Choisissons ensuite un entier a inversible modulo N . Prenons simplement $a = 2$ qui est bien premier avec $N = 21$. Nous devons retrouver l'ordre de a modulo N qui est ici $r = 6$.

Réordonnons les éléments de $|\psi_2\rangle$ en regroupant les termes selon le second facteur qui est l'un des $|\underline{a^k}\rangle$ pour k variant de 0 à $r - 1 = 5$.

$$\begin{aligned}
 |\psi_2\rangle &= \frac{1}{\sqrt{512}} \left(|0\rangle + |6\rangle + \dots + |504\rangle + |510\rangle \right) |1\rangle \\
 &\quad \frac{1}{\sqrt{512}} \left(|1\rangle + |7\rangle + \dots + |505\rangle + |511\rangle \right) |2\rangle \\
 &\quad \frac{1}{\sqrt{512}} \left(|2\rangle + |8\rangle + \dots + |506\rangle \right) |4\rangle \\
 &\quad \frac{1}{\sqrt{512}} \left(|3\rangle + |9\rangle + \dots + |507\rangle \right) |8\rangle \\
 &\quad \frac{1}{\sqrt{512}} \left(|4\rangle + |10\rangle + \dots + |508\rangle \right) |16\rangle \\
 &\quad \frac{1}{\sqrt{512}} \left(|5\rangle + |11\rangle + \dots + |509\rangle \right) |11\rangle
 \end{aligned}$$

Bien noter la différence avec le cas où r était une puissance de 2. Ici on n'obtient pas un tableau rectangulaire. Les deux premières lignes contiennent une somme de 86 termes alors que les suivantes en ont seulement 85.

On effectue ensuite une mesure du second registre et on obtient l'un des $|\underline{\alpha}^k\rangle$. Dans la suite on suppose par exemple qu'on obtient $|\underline{2}\rangle$, alors le premier registre, une fois normalisé, contient le qubit :

$$|\bar{\psi}_3\rangle = \frac{1}{\sqrt{86}} \left(|1\rangle + |7\rangle + |13\rangle + \dots + |505\rangle + |511\rangle \right).$$

2.4. L'exemple $N = 21$: fin

La dernière étape est d'appliquer la transformée de Fourier inverse et d'effectuer une mesure sur le premier registre.

$$\begin{aligned}
 |\bar{\psi}_4\rangle &= \hat{F}^{-1} |\bar{\psi}_3\rangle \\
 &= \hat{F}^{-1} \left(\frac{1}{\sqrt{86}} \sum_{\alpha=0}^{85} |6\alpha + 1\rangle \right) \\
 &= \frac{1}{\sqrt{86}} \sum_{\alpha=0}^{85} \frac{1}{\sqrt{512}} \sum_{j=0}^{511} e^{-2i\pi \frac{(6\alpha+1)j}{512}} |j\rangle \\
 &= \frac{1}{\sqrt{512}} \sum_{j=0}^{511} \left(\frac{1}{\sqrt{86}} \sum_{\alpha=0}^{85} e^{-2i\pi \frac{6\alpha j}{512}} \right) e^{-2i\pi \frac{j}{512}} |j\rangle
 \end{aligned}$$

Cette fois la somme

$$\Sigma(j) = \frac{1}{\sqrt{86}} \sum_{\alpha=0}^{85} e^{-2i\pi \frac{6\alpha j}{512}}$$

est un nombre complexe qui peut prendre des valeurs autres que 0 et 1.

La mesure du premier registre conduit à la valeur j avec la probabilité :

$$p_j = \frac{1}{512} |\Sigma(j)|^2.$$

Ces probabilités sont presque nulles sauf pour les valeurs de j proches des réels $\frac{2^n \ell}{r}$ (qui ne sont pas des entiers) avec $\ell = 0, 1, \dots, r - 1$. Nous avons ici

$$\frac{2^n}{r} = \frac{512}{6} = 85.33 \dots$$

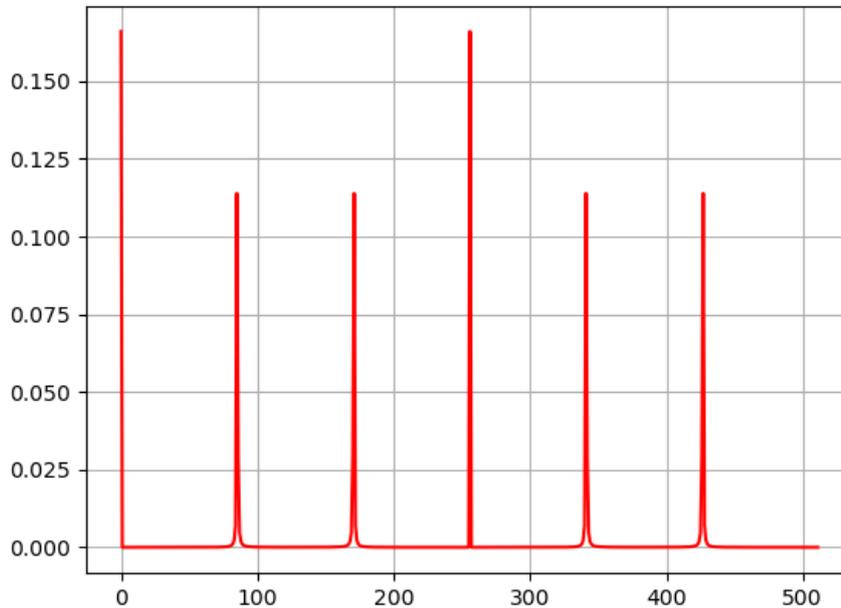
Les valeurs $\frac{2^n \ell}{r}$ pour $\ell = 0, \dots, 5$ sont les réels :

$$0 \quad 85.33 \dots \quad 170.66 \dots \quad 256 \quad 341.33 \dots \quad 426.66 \dots$$

Donc ici les probabilités sont presque nulles, sauf autour de entiers :

$$j = 0, \quad j = 85, \quad j = 171, \quad j = 256, \quad j = 341, \quad j = 427.$$

Voici le diagramme des probabilités p_j , pour $0 \leq j < 512$. Les 6 pics sont nettement visibles.



Voici le tableau des valeurs autour du pic à $j = 427$.

j	p_j
422	0.00062...
423	0.00099...
424	0.00186...
425	0.00469...
426	0.02888...
427	0.11389...
428	0.00702...
429	0.00226...
430	0.00109...
431	0.00063...

On note la probabilité élevée en $j = 427$, une probabilité plus faible en $j = 426$ (qui s'explique car pour $\ell = 5$, $\frac{2^{n\ell}}{r} = \frac{512 \times 5}{6} = 426.66\dots$), pour les valeurs plus éloignées les probabilités sont presque nulles.

2.5. Ordre

On obtient l'ordre r , ou l'un de ses facteurs, à partir du développement en fractions continues comme expliqué précédemment. À part cela, les conclusions sont similaires aux cas du chapitre « Algorithme de Shor » :

- Si la mesure donne un entier j proche de 0, alors on n'obtient aucune information sur l'ordre r , il faut recommencer.
- Si la mesure donne un entier j proche de 85 ou proche de 427, alors le développement en fraction continue de $\frac{j}{512}$ donne l'ordre $r = 6$.

- Si la mesure donne un entier proche j proche de 171 ou 341 alors on n'obtient pas r mais le facteur $r' = 3$; si la mesure donne un entier j proche de 256 alors on n'obtient pas r mais le facteur $r'' = 2$. Dans ces cas on relance l'algorithme pour obtenir la factorisation complète.

2.6. Conclusion

Il nous reste à justifier que l'approximation du pic conduit au bon résultat.

Théorème 1 (Hardy – Wright).

Soit $x \in \mathbb{R}$. Soit une fraction $\frac{p}{q}$ telle que :

$$\left| x - \frac{p}{q} \right| < \frac{1}{2q^2}.$$

Alors $\frac{p}{q}$ est obtenu comme l'une des fractions du développement en fractions continues de x .

Dans notre situation nous considérons l'entier m le plus proche de $\frac{2^n \ell}{r}$. Donc $\left| m - \frac{2^n \ell}{r} \right| \leq \frac{1}{2}$. En posant $x = \frac{m}{2^n}$ on obtient $\left| x - \frac{\ell}{r} \right| \leq \frac{1}{2^{n+1}}$. Par notre choix de n on a $2^n \geq N^2 > r^2$, donc $\left| x - \frac{\ell}{r} \right| \leq \frac{1}{2r^2}$. Par le théorème, $\frac{\ell}{r}$ s'obtient comme l'une des fractions du développement en fractions continues de x , comme on l'avait expliqué dans la première section.

Le reste du chapitre est consacré à la théorie des groupes afin de justifier la pertinence des hypothèses 1 et 2 de l'algorithme de Shor.

3. Ordre d'un élément

3.1. Définition

Soit (G, \times) un groupe commutatif ayant pour élément neutre e . L'**ordre** de $x \in G$, noté $\text{ord}(x)$, est le plus petit entier $r > 0$, tel que $x^r = e$.

Voici quelques propriétés de l'ordre :

- si k est un entier tel que $x^k = e$ alors $\text{ord}(x)$ divise k ;
- $\text{ord}(x^k)$ divise $\text{ord}(x)$.

Le théorème de Lagrange pour un groupe fini G de cardinal n affirme que $x^n = e$ quel que soit x . Ainsi $\text{ord}(x)$ divise n , quel que soit l'élément x . En particulier, tout élément admet un ordre fini.

3.2. Plus grand ordre

Proposition 1.

Soit G un groupe fini et m le plus grand ordre parmi tous les $x \in G$, alors pour tout $x \in G$, $\text{ord}(x)$ divise m .

Une formulation équivalente est la suivante : soit ℓ le plus petit entier tel que pour tout $x \in G$ on ait $x^\ell = e$, alors il existe $x_0 \in G$ tel que $\text{ord}(x_0) = \ell$.

Pour la preuve nous aurons besoin du résultat suivant :

Lemme 1.

Soient deux éléments x et y d'ordres $m = \text{ord}(x)$ et $n = \text{ord}(y)$ premiers entre eux, alors $\text{ord}(x \cdot y) = mn$.

Démonstration. Notons $r = \text{ord}(xy)$. Il s'agit de montrer $r = mn$ en prouvant que $r|mn$ puis que $mn|r$. Tout d'abord $(xy)^{mn} = x^{mn} \cdot y^{mn} = (x^m)^n \cdot (y^n)^m = e$, donc $r|mn$. Réciproquement, on sait que $(xy)^r = e$ donc $x^r \cdot y^r = e$, autrement dit $z = x^r = y^{-r}$. D'une part $z^m = (x^r)^m = x^{rm} = (x^m)^r = e$, donc $\text{ord}(z)|m$, de même $z^n = (y^r)^n = e$, donc $\text{ord}(z)|n$. Comme m et n sont premiers entre eux, alors $\text{ord}(z) = 1$, c'est-à-dire $z = e$. Ainsi $x^r = e$, donc $m = \text{ord}(x)|r$ et $y^r = e$ donc $n = \text{ord}(y)|r$, ainsi $mn|r$. \square

Preuve de la proposition. Soit m le plus grand ordre parmi les éléments de G , il existe donc y d'ordre m . Fixons x un élément quelconque de G et notons n son ordre. Il s'agit de montrer que $n|m$. Par l'absurde on suppose que n ne divise pas m . On va obtenir une contradiction en construisant un élément z avec $\text{ord}(z) > m$. Par exemple si m et n sont premiers entre eux, alors $z = xy$ est d'ordre $mn > m$, ce qui donne la contradiction. Si m et n ne sont pas premiers entre eux, soit p un facteur premier commun à m et n tel que $p^e | n$, $p^f | m$ avec $e > f$ les plus grands possibles (un tel p existe car n ne divise pas m). Soient $y' = y^{p^f}$ et $x' = x^{n/p^e}$. Alors y' a pour ordre $m' = m/p^f$ et x' a pour ordre $n' = p^e$. Les entiers m' et n' sont premiers entre eux (car m' n'est pas divisible par p). Ainsi $z = x'y'$ a pour ordre $m'n' = \frac{m}{p^f} p^e = mp^{e-f} > m$. On obtient bien la contradiction cherchée. \square

4. Le groupe $(\mathbb{Z}/p\mathbb{Z})^*$

Dans toute la suite nous allons étudier en détails le groupe $(\mathbb{Z}/n\mathbb{Z})^*$ qui est l'ensemble des éléments inversibles modulo n . Nous commençons par le cas d'un nombre premier p . Nous savons déjà que

$$\text{Card}(\mathbb{Z}/p\mathbb{Z})^* = \varphi(p) = p - 1$$

mais nous souhaitons aller plus loin en étudiant la structure de $(\mathbb{Z}/p\mathbb{Z})^*$.

4.1. Isomorphisme

Théorème 2.

Le groupe $((\mathbb{Z}/p\mathbb{Z})^*, \times)$ est isomorphe au groupe $(\mathbb{Z}/(p-1)\mathbb{Z}, +)$.

Pour la preuve nous aurons besoin du résultat suivant.

Proposition 2.

Un polynôme $P \in \mathbb{Z}/p\mathbb{Z}[X]$ de degré d possède au plus d racines, c'est-à-dire des éléments $x \in \mathbb{Z}/p\mathbb{Z}$ tels que $P(x) \equiv 0 \pmod{p}$.

Idee de la preuve de la proposition. C'est un fait général : sur un corps k un polynôme $P \in k[X]$ de degré d a au plus d racines. En effet, $a \in k$ est une racine si et seulement si $X - a$ est un facteur de $P(X)$. Si $\{a_1, \dots, a_k\}$ est l'ensemble des racines de $P(X)$ alors $(X - a_1)(X - a_2) \cdots (X - a_k)$ divise $P(X)$ et donc en comparant les degrés : $\text{deg } P \geq k$. \square

Preuve du théorème. L'ensemble $(\mathbb{Z}/p\mathbb{Z})^* = \{1, 2, \dots, p-1\}$ est en bijection avec $\mathbb{Z}/(p-1)\mathbb{Z} = \{0, 1, \dots, p-2\}$. Mais on veut plus : on veut que les structures de groupes, avec la loi « \times » pour $(\mathbb{Z}/p\mathbb{Z})^*$ et « $+$ » pour $\mathbb{Z}/(p-1)\mathbb{Z}$, soient préservées. Nous allons trouver un élément a d'ordre $p-1$ dans $(\mathbb{Z}/p\mathbb{Z})^*$ ce qui va nous permettre de construire l'isomorphisme :

$$\begin{aligned} \phi : \mathbb{Z}/(p-1)\mathbb{Z} &\longrightarrow (\mathbb{Z}/p\mathbb{Z})^* \\ k &\longmapsto a^k \pmod{p}. \end{aligned}$$

Cette application ϕ est bien définie car $\phi(k + \ell(p-1)) = a^{k+\ell(p-1)} = a^k = \phi(k)$ et est un morphisme car $\phi(k + k') = \phi(k) \times \phi(k')$. De plus ϕ est bijective, car elle est surjective (puisque les $\{a^k\}$ sont $p-1$ éléments distincts, ils forment l'ensemble d'arrivée) et les ensembles de départ et d'arrivée ont le même nombre d'éléments.

Pour montrer qu'il existe un élément d'ordre $p-1$, remarquons d'abord que pour tout élément $x \in (\mathbb{Z}/p\mathbb{Z})^*$ on a $\text{ord}(x) | p-1$. En effet, par le petit théorème de Fermat, $x^{p-1} \equiv 1 \pmod{p}$. Soit m le plus grand des ordres des éléments de $(\mathbb{Z}/p\mathbb{Z})^*$. On vient de voir que $m | p-1$, donc $m \leq p-1$. Par la proposition 1, on sait que pour tout $x \in (\mathbb{Z}/p\mathbb{Z})^*$, $\text{ord}(x)$ divise m , c'est-à-dire $x^m \equiv 1 \pmod{p}$. Considérons le polynôme défini par $P(X) = X^m - 1$. Alors pour tout $x \in (\mathbb{Z}/p\mathbb{Z})^*$, $P(x) = x^m - 1 \equiv 0 \pmod{p}$. Nous avons donc trouvé $p-1$ racines au polynôme P de degré m , donc $p-1 \leq m$. Conclusion : $m = p-1$, donc par définition de m il existe un élément a d'ordre $p-1$. \square

Remarque : la preuve n'est pas constructive, pour trouver a d'ordre $p - 1$ il n'y a pas d'autres moyens que de tester différentes valeurs de a et de calculer à chaque fois a, a^2, a^3, \dots

4.2. Éléments d'ordre pair

Proposition 3.

Dans $(\mathbb{Z}/p\mathbb{Z})^*$, avec $p \geq 3$, la moitié au moins des éléments sont d'ordre pair.

Démonstration. Notons $\psi : (\mathbb{Z}/p\mathbb{Z})^* \rightarrow \mathbb{Z}/(p-1)\mathbb{Z}$ l'isomorphisme de groupes. Alors l'ordre d'un élément x de $(\mathbb{Z}/p\mathbb{Z})^*$ est égal à l'ordre de l'élément $\psi(x)$ dans $\mathbb{Z}/(p-1)\mathbb{Z}$.

L'ordre d'un élément y dans le groupe additif $\mathbb{Z}/(p-1)\mathbb{Z}$ est le plus petit entier $r > 0$ tel que $r \cdot y \equiv 0 \pmod{p-1}$. Considérons les entiers impairs $y = 2k + 1, k = 0, 1, \dots, \frac{p-1}{2}$. Ces y ont des ordres pairs : en effet si $r \cdot (2k + 1) \equiv 0 \pmod{p-1}$ alors $r(2k + 1) = \ell(p-1)$. Comme $\ell(p-1)$ est pair (car p est premier et supérieur à 3) et que $2k + 1$ est impair, r est nécessairement pair. Ainsi la moitié au moins des éléments de $\mathbb{Z}/(p-1)\mathbb{Z}$ sont d'ordre pair. Par isomorphie, il en est de même pour $(\mathbb{Z}/p\mathbb{Z})^*$. □

4.3. Racines carrées de 1

Le point-clé initial de l'algorithme de Shor est la factorisation $x^2 - 1 = (x - 1)(x + 1)$. Dans $(\mathbb{Z}/n\mathbb{Z})^*$ trouver un élément tel que $x^2 - 1 = 0$ peut permettre une factorisation de n à l'aide de $(x - 1)(x + 1)$.

Définition.

On appelle *racine carrée de 1 modulo n* tout élément x de $\mathbb{Z}/n\mathbb{Z}$ tel que

$$x^2 \equiv 1 \pmod{n}$$

Une telle racine carrée est en fait nécessairement un élément de $(\mathbb{Z}/n\mathbb{Z})^*$. Attention ! L'équation $X^2 - 1 = 0$ est une équation polynomiale de degré 2. Elle peut avoir plus de deux solutions dans $(\mathbb{Z}/n\mathbb{Z})^*$ qui n'est pas toujours un corps, nous y reviendrons. Revenons au cas où $n = p$ est un nombre premier, pour lequel $\mathbb{Z}/p\mathbb{Z}$ est un corps. Il y a dans ce cas effectivement deux solutions.

Proposition 4.

Il y a exactement deux racines carrées modulo p (où $p \geq 3$ est un nombre premier) : $+1$ et -1 .

Encore une fois, ceci n'est valable que modulo un nombre premier.

Après l'application de l'isomorphisme $\psi : (\mathbb{Z}/p\mathbb{Z})^* \rightarrow \mathbb{Z}/(p-1)\mathbb{Z}$, les deux racines carrées sont $\psi(1) = 0$ et $\psi(-1) = \frac{p-1}{2}$ (en effet l'identité $(-1)^2 \equiv 1 \pmod{p}$ devient $2 \times \frac{p-1}{2} \equiv 0 \pmod{p-1}$).

Démonstration. Pour $x = +1$ on a bien sûr $x^2 = 1$. L'écriture $x = -1$ est une autre façon d'écrire $x = p - 1$ (car $x = p - 1 \equiv -1 \pmod{p}$) et bien sûr $x^2 = (-1)^2 = 1$.

Pour justifier qu'il n'y a pas d'autres racines : si x est une racine carrée de 1 alors $x^2 - 1 \equiv 0 \pmod{p}$ donc $(x - 1)(x + 1) \equiv 0 \pmod{p}$. Comme $\mathbb{Z}/p\mathbb{Z}$ est un corps, un produit est nul si et seulement si un des facteur est nul, donc $x - 1 \equiv 0 \pmod{p}$ ou $x + 1 \equiv 0 \pmod{p}$, c'est-à-dire $x = +1$ ou $x = -1$ (modulo p).

Un autre argument serait de dire que $+1$ et -1 sont racines du polynôme $P(X) = X^2 - 1$, et comme $\deg P(X) = 2$, il n'y a pas d'autres solutions par la proposition 2. □

Cependant le point clé de l'algorithme de Shor est un peu plus délicat, il s'agit de trouver un entier r pair tel que $x^r \equiv 1 \pmod{n}$, ce qui donne la factorisation $(x^{r/2} - 1)(x^{r/2} + 1) \equiv 0 \pmod{n}$ et peut conduire à une factorisation de n à partir de la factorisation $(x^{r/2} - 1)(x^{r/2} + 1)$. Il faut supposer que $x^{r/2} + 1 \not\equiv 0 \pmod{n}$ pour que la procédure fonctionne, voir l'hypothèse 2 du chapitre « Algorithme de Shor ».

Faisons le point : pour l'algorithme de Shor, on cherche un entier pair r tel que $x^{r/2}$ soit une racine carrée de 1, en excluant le cas où $x^{r/2} \equiv 1 \pmod{n}$ (pour lequel l'ordre serait $r/2$ et pas r) et $x^{r/2} \equiv -1 \pmod{n}$ (qui ne permet pas toujours d'obtenir une factorisation).

Dans le cas d'un nombre premier : une telle racine carrée n'existe pas, car on a vu que les deux seules racines carrées de 1 sont +1 et -1 qui sont justement les deux cas à éviter.

Ainsi :

Proposition 5.

Lorsque p est un nombre premier, l'hypothèse 1 ou l'hypothèse 2 de l'algorithme de Shor n'est pas vérifiée.

Noter que ce résultat négatif n'a pas d'incidence pour l'algorithme de Shor pour lequel il s'agit de factoriser un entier qui n'est pas premier. Nous avons déjà expliqué pourquoi cette proposition est vraie, nous le justifions de nouveau de manière plus condensée.

Démonstration. Soit $x \in (\mathbb{Z}/p\mathbb{Z})^*$. Supposons que l'hypothèse 1 soit vraie, c'est-à-dire que l'ordre r de x est pair. Comme $x^r - 1 \equiv 0 \pmod{p}$ alors $(x^{r/2} - 1)(x^{r/2} + 1) \equiv 0 \pmod{p}$. Mais $x^{r/2} - 1 \not\equiv 0 \pmod{p}$ car sinon l'ordre serait $\leq r/2$. Comme p est un nombre premier alors $x^{r/2} - 1 \not\equiv 0$ est inversible. Si y désigne son inverse, alors en multipliant par cet inverse on obtient $y(x^{r/2} - 1)(x^{r/2} + 1) \equiv 0 \pmod{p}$, donc $x^{r/2} + 1 \equiv 0 \pmod{p}$ et ainsi $x^{r/2} \equiv -1 \pmod{p}$ ce qui empêche l'hypothèse 2 d'être valide. \square

5. Le groupe $(\mathbb{Z}/p^\alpha\mathbb{Z})^*$

L'étape suivante est d'étudier le groupe des éléments inversibles modulo une puissance d'un nombre premier.

5.1. Isomorphisme

On sait déjà que $\text{Card}(\mathbb{Z}/p^\alpha\mathbb{Z})^* = \varphi(p^\alpha) = p^\alpha - p^{\alpha-1} = p^\alpha \left(1 - \frac{1}{p}\right)$, mais nous allons aller plus loin en montrant que $(\mathbb{Z}/p^\alpha\mathbb{Z})^*$ est un groupe cyclique (pour $p \geq 3$), c'est-à-dire qu'il peut être engendré par un seul élément.

5.2. Isomorphisme

Théorème 3.

Si $p \geq 3$ est un nombre premier alors le groupe $(\mathbb{Z}/p^\alpha\mathbb{Z})^*$ est isomorphe au groupe $\mathbb{Z}/(p-1)\mathbb{Z} \times \mathbb{Z}/p^{\alpha-1}\mathbb{Z}$ et c'est un groupe cyclique. Pour $p = 2$, $(\mathbb{Z}/2^\alpha\mathbb{Z})^*$ est isomorphe au groupe $\mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2^{\alpha-2}\mathbb{Z}$.

Démonstration. Nous nous limitons à $p \geq 3$, situation de l'algorithme de Shor. Nous allons construire dans $(\mathbb{Z}/p^\alpha\mathbb{Z})^*$ un élément a d'ordre $p-1$ et un élément b d'ordre $p^{\alpha-1}$ ce qui conduira à l'isomorphisme :

$$\begin{aligned} \phi : \mathbb{Z}/(p-1)\mathbb{Z} \times \mathbb{Z}/p^{\alpha-1}\mathbb{Z} &\longrightarrow (\mathbb{Z}/p^\alpha\mathbb{Z})^* \\ (k, \ell) &\longmapsto a^k b^\ell \pmod{p^\alpha}. \end{aligned}$$

Tout d'abord soit a' un élément d'ordre $p-1$ dans $(\mathbb{Z}/p\mathbb{Z})^*$ (un tel élément existe par le théorème 2), donc $a'^{p-1} \equiv 1 \pmod{p}$. Considérons a' comme élément de $(\mathbb{Z}/p^\alpha\mathbb{Z})^*$, et notons $r = \text{ord}(a')$ de sorte que $a'^r \equiv 1 \pmod{p^\alpha}$. Ainsi $a'^r - 1$ est divisible par p^α , donc a fortiori par p , donc $a'^r \equiv 1 \pmod{p}$. Ainsi l'ordre de a' dans $(\mathbb{Z}/p\mathbb{Z})^*$ divise r : c'est-à-dire $p-1|r$. Notons $a = a'^{\frac{r}{p-1}}$. C'est un élément d'ordre $p-1$ dans $(\mathbb{Z}/p^\alpha\mathbb{Z})^*$: en effet $a^{p-1} = a'^r \equiv 1 \pmod{p^\alpha}$ et par définition de l'ordre r , il ne peut exister d'entier plus petit.

Notons $b = 1 + p$ alors par le lemme ci-dessous $b^{p^{\alpha-1}} = (1+p)^{p^{\alpha-1}} \equiv 1 \pmod{p^\alpha}$, donc en particulier, l'ordre de b divise $p^{\alpha-1}$, mais toujours par ce lemme, pour $k < \alpha-1$, $b^{p^{k-1}} \not\equiv 1 \pmod{p^\alpha}$. Ainsi $\text{ord}(b) = p^{\alpha-1}$.

Nous avons donc trouvé a avec $\text{ord}(a) = p-1$ et b avec $\text{ord}(b) = p^{\alpha-1}$. Ces deux ordres sont premiers entre eux (car $p-1$ et p le sont) donc par le lemme 1, l'élément ab est d'ordre $(p-1)p^{\alpha-1}$.

On a donc montré en plus que $(\mathbb{Z}/p^\alpha\mathbb{Z})^*$ est engendré par le seul élément ab , c'est donc un groupe cyclique. \square

Voici l'énoncé du lemme utilisé dans la preuve.

Lemme 2.

Soient $k \geq 0$ et $p \geq 3$ un nombre premier, alors :

$$(1 + p)^{p^k} \equiv 1 + p^{k+1} \pmod{p^{k+2}}.$$

La preuve de ce lemme peut être omise en première lecture. Elle se fait par récurrence sur k et reste assez technique. C'est une version améliorée de l'exercice classique suivant : $(x + y)^p \equiv x^p + y^p \pmod{p}$. Les ingrédients sont les mêmes dans la preuve qui nous concerne : il faut utiliser la formule du binôme de Newton et utiliser que le coefficient $\binom{p}{i}$ est divisible par p , lorsque $0 < i < p$.

Nous aurons besoin pour la preuve de la variante suivante : si $x \equiv y \pmod{p^k}$ avec $k \geq 1$, alors $x^p \equiv y^p \pmod{p^{k+1}}$. Il suffit d'écrire $x = y + \lambda p^k$ puis d'utiliser la formule du binôme de Newton, $x^p = (y + \lambda p^k)^p = y^p + \dots$ où les termes de la somme omis sont tous divisibles par p^{k+1} (car de nouveau $\binom{p}{i}$ est divisible par p).

Preuve du lemme. La démonstration se fait par récurrence. Pour $k = 0$, l'assertion est vraie : $(1 + p)^1 \equiv 1 + p \pmod{p^2}$. Supposons l'assertion vraie au rang $k \geq 0$ et prouvons-la au rang $k + 1$. Par hypothèse de récurrence

$$(1 + p)^{p^k} \equiv 1 + p^{k+1} \pmod{p^{k+2}},$$

donc par la variante rappelée ci-dessus :

$$(1 + p)^{p^{k+1}} = \left((1 + p)^{p^k} \right)^p \equiv (1 + p^{k+1})^p \pmod{p^{k+3}}.$$

Développons $(1 + p^{k+1})^p$ selon la formule du binôme de Newton :

$$(1 + p^{k+1})^p = 1 + p \cdot p^{k+1} + \dots$$

Les termes omis dans les points de suspension sont tous divisibles par p^{k+3} donc $(1 + p^{k+1})^p \equiv 1 + p^{k+2} \pmod{p^{k+3}}$, ce qui conduit au résultat souhaité. □

5.3. Éléments d'ordre pair

Proposition 6.

Dans $(\mathbb{Z}/p^\alpha\mathbb{Z})^*$, avec $p \geq 3$, la moitié au moins des éléments sont d'ordre pair.

Démonstration. La preuve est similaire à celle de la proposition 3. L'ordre de $(2k + 1, \ell)$ est pair dans $\mathbb{Z}/(p - 1)\mathbb{Z} \times \mathbb{Z}/p^{\alpha-1}\mathbb{Z}$, quel que soit l'entier impair $2k + 1$ et quel que soit ℓ . Donc la moitié au moins des éléments de $\mathbb{Z}/(p - 1)\mathbb{Z} \times \mathbb{Z}/p^{\alpha-1}\mathbb{Z}$ sont d'ordre pair. Par l'isomorphisme du théorème 3, il en est de même pour $(\mathbb{Z}/p^\alpha\mathbb{Z})^*$. □

5.4. Racines carrées de 1

On rappelle qu'une racine carrée de 1 modulo p^α est un élément x tel que $x^2 \equiv 1 \pmod{p^\alpha}$.

Proposition 7.

Il y a exactement deux racines carrées modulo p^α (où $p \geq 3$ est un nombre premier) : $+1$ et -1 .

Attention cette fois $\mathbb{Z}/p^\alpha\mathbb{Z}$ n'est pas un corps, il pourrait donc y avoir a priori plus de deux racines carrées de 1. Par l'isomorphisme, dans $\mathbb{Z}/(p - 1)\mathbb{Z} \times \mathbb{Z}/p^{\alpha-1}\mathbb{Z}$ ces deux racines carrées sont $(1, 0)$ et $(\frac{p-1}{2}, 0)$.

Démonstration. Notons $\psi : (\mathbb{Z}/p^\alpha\mathbb{Z})^* \rightarrow \mathbb{Z}/(p - 1)\mathbb{Z} \times \mathbb{Z}/p^{\alpha-1}\mathbb{Z}$ l'isomorphisme de groupes du théorème 3. Si $\psi(x) = (a, b)$ alors $\psi(x^k) = (ka, kb)$ et $\psi(1) = (0, 0)$ (l'élément neutre du groupe multiplicatif s'envoie sur l'élément neutre du groupe additif). Donc $x^2 \equiv 1 \pmod{p^\alpha}$ équivaut à $(2a, 2b) \equiv (0, 0)$, où plus précisément $2a \equiv 0 \pmod{p - 1}$ et $2b \equiv 0 \pmod{p^{\alpha-1}}$. Comme $\text{pgcd}(2, p) = 1$ alors 2 est inversible modulo $p^{\alpha-1}$ la seconde équation donne donc $b \equiv 0 \pmod{p^{\alpha-1}}$. En revanche, comme $p - 1$ est pair, l'équation $2a \equiv 0 \pmod{p - 1}$ admet deux solutions $a = 0$ et $a = \frac{p-1}{2}$.

Bilan : nous avons obtenu deux solutions $(0, 0)$ et $(\frac{p-1}{2}, 0)$ qui par l'isomorphisme donnent les deux seules racines carrées 1 et -1 . □

Nous n'allons pas étudier l'équation $x^{r/2} + 1 \equiv 0 \pmod{p^\alpha}$, d'une part le cas $n = p^\alpha$ est étudié spécifiquement dans l'algorithme de Shor, d'autre part on étudiera plus tard cette équation dans le cas plus général d'un n quelconque.

6. Le théorème des restes chinois

6.1. Cas simple

Théorème 4.

Soient p et q deux nombres premiers entre eux alors $\mathbb{Z}/pq\mathbb{Z}$ est isomorphe à $\mathbb{Z}/p\mathbb{Z} \times \mathbb{Z}/q\mathbb{Z}$.

Par exemple $\mathbb{Z}/6\mathbb{Z}$ est isomorphe à $\mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/3\mathbb{Z}$. Attention ! $\mathbb{Z}/4\mathbb{Z}$ n'est pas isomorphe à $\mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z}$.

Démonstration. Notons $\phi : \mathbb{Z}/pq\mathbb{Z} \rightarrow \mathbb{Z}/p\mathbb{Z} \times \mathbb{Z}/q\mathbb{Z}$, l'application définie par

$$\phi(x) = (\bar{x}, \tilde{x})$$

où \bar{x} est la réduction de x modulo p , et \tilde{x} est la réduction de x modulo q .

Cette application ϕ est bien définie et c'est un morphisme de groupes. De plus elle est injective : si $\phi(x) = (0, 0)$ alors $x \equiv 0 \pmod{p}$ et $x \equiv 0 \pmod{q}$, donc p divise x et q divise x ; ainsi p et q étant premiers entre eux, le produit pq divise x , donc $x \equiv 0 \pmod{pq}$. Comme les ensembles de départ et d'arrivée ont le même cardinal pq alors ϕ est bijective. \square

Corollaire 1.

Soient p et q deux nombres premiers entre eux. Soient $a, b \in \mathbb{Z}$. Il existe $x \in \mathbb{Z}$ tel que :

$$\begin{cases} x \equiv a \pmod{p} \\ x \equiv b \pmod{q} \end{cases} .$$

6.2. Version générale

Théorème 5.

Soit $p_1^{\alpha_1} \times p_2^{\alpha_2} \times \dots \times p_\ell^{\alpha_\ell}$ la décomposition d'un entier n en produit de facteurs premiers. Alors $\mathbb{Z}/n\mathbb{Z}$ est isomorphe au groupe $\mathbb{Z}/p_1^{\alpha_1}\mathbb{Z} \times \dots \times \mathbb{Z}/p_\ell^{\alpha_\ell}\mathbb{Z}$.

La preuve est une récurrence à partir du cas pq .

7. Le groupe $(\mathbb{Z}/n\mathbb{Z})^*$

7.1. Groupe produit

Si A et B sont deux groupes, alors le **groupe produit** $A \times B$ est défini par la loi $(a, b) \times (a', b') = (aa', bb')$ et l'élément neutre est (e_A, e_B) formé à partir des éléments neutres de chaque groupe. En particulier $(a, b)^k = (a^k, b^k)$ et $(a, b)^{-1} = (a^{-1}, b^{-1})$. L'ordre de (a, b) est le plus petit multiple commun des ordres de a et b :

$$\text{ord}(a, b) = \text{ppcm}(\text{ord}(a), \text{ord}(b)).$$

7.2. Isomorphisme

Proposition 8.

Soient p et q deux nombres premiers entre eux alors $(\mathbb{Z}/pq\mathbb{Z})^*$ est isomorphe à $(\mathbb{Z}/p\mathbb{Z})^* \times (\mathbb{Z}/q\mathbb{Z})^*$.

Démonstration. C'est le théorème des restes chinois (cas pq) avec le fait que x est premier avec pq si et seulement si x est premier avec p et avec q . \square

La version générale est la suivante :

Théorème 6.

Soit $p_1^{\alpha_1} \times p_2^{\alpha_2} \times \dots \times p_\ell^{\alpha_\ell}$ la décomposition d'un entier n en produit de facteurs premiers. Alors $(\mathbb{Z}/n\mathbb{Z})^*$ est isomorphe au groupe $(\mathbb{Z}/p_1^{\alpha_1}\mathbb{Z})^* \times \dots \times (\mathbb{Z}/p_\ell^{\alpha_\ell}\mathbb{Z})^*$.

7.3. Éléments d'ordre pair**Proposition 9.**

Soit $n = \prod_{i=1}^{\ell} p_i^{\alpha_i}$ la décomposition de n en produits de ℓ facteurs, avec $p_i \geq 3$. La proportion d'éléments d'ordre pair de $(\mathbb{Z}/n\mathbb{Z})^*$ est supérieure à $1 - \frac{1}{2^\ell}$.

Démonstration. Par le théorème des restes chinois pour les éléments inversibles (théorème 6), chaque élément $x \in (\mathbb{Z}/n\mathbb{Z})^*$ est en correspondance avec un élément (x_1, \dots, x_ℓ) où $x_i \in (\mathbb{Z}/p_i^{\alpha_i}\mathbb{Z})^*$. L'ordre de x est le ppcm des ordres des x_i , donc $\text{ord}(x)$ est pair si et seulement si l'un au moins des $\text{ord}(x_i)$ est pair. Autrement dit $\text{ord}(x)$ est impair si et seulement si tous les $\text{ord}(x_i)$ sont impairs. Par la proposition 6, pour chaque i , la proportion de x_i d'ordre impair est strictement inférieure à $\frac{1}{2}$, donc la proportion de ℓ -uplets (x_1, \dots, x_ℓ) dont tous les éléments sont d'ordre impair est strictement inférieure à $(\frac{1}{2})^\ell$. Par complément la proportion d'éléments d'ordre pair dans $(\mathbb{Z}/n\mathbb{Z})^*$ est supérieure à $1 - (\frac{1}{2})^\ell$. \square

7.4. Racines carrées de 1**Proposition 10.**

Soit $n = \prod_{i=1}^{\ell} p_i^{\alpha_i}$ la décomposition de n en produit de ℓ facteurs, avec $p_i \geq 3$. Il y a exactement 2^ℓ racines carrées de 1 modulo n .

Démonstration. Par le théorème des restes chinois pour les éléments inversibles (théorème 6), un élément $x \in (\mathbb{Z}/n\mathbb{Z})^*$ est en correspondance avec un élément (x_1, \dots, x_ℓ) où $x_i \in (\mathbb{Z}/p_i^{\alpha_i}\mathbb{Z})^*$. Alors x vérifie $x^2 \equiv 1 \pmod{n}$ si et seulement si $x_i^2 \equiv 1 \pmod{p_i^{\alpha_i}}$, pour tout $i = 1, \dots, \ell$. Par la proposition 7, il existe deux racines carrées pour chaque i : $x_i = +1$ ou $x_i = -1$, ce qui donne au total 2^ℓ solutions $(x_1, \dots, x_\ell) = (\pm 1, \pm 1, \dots, \pm 1)$, donc 2^ℓ racines carrées dans $(\mathbb{Z}/n\mathbb{Z})^*$. \square

8. Les hypothèses de l'algorithme de Shor**8.1. Préalable de l'algorithme de Shor**

L'algorithme de Shor a pour but de factoriser un entier n . Plus précisément on souhaite trouver un facteur k de n (autre que 1 et n). On pourra ensuite relancer l'algorithme avec $\frac{n}{k}$ et avec k .

Au préalable on exclut certaines situations :

- n n'est pas un entier pair. Il est très facile de tester si n est pair en vérifiant si $n \equiv 0 \pmod{2}$. Si n est pair, alors il est divisible par 2 et c'est terminé.
- n n'est pas un nombre premier p . Il existe des tests performants pour savoir si un entier n est premier ou pas sans calculer sa factorisation (voir le chapitre « Arithmétique »).
- n n'est pas une puissance p^α d'un nombre premier. Un test simple repose sur le fait que si $n = p^\alpha$ alors $\alpha = \log_p(n) \leq \log_2(n)$. Il suffit donc de tester si $n^{\frac{1}{k}}$ est un entier pour un k parmi $2, 3, \dots$ jusqu'à $\log_2(n)$.

Dans la suite on considère donc un entier n impair, ayant au moins deux facteurs premiers distincts. On écrit sa décomposition $n = \prod_{i=1}^{\ell} p_i^{\alpha_i}$ en produit de ℓ facteurs, avec $p_i \geq 3$ premiers.

8.2. L'algorithme de Shor fonctionne au moins une fois sur deux

Hypothèse 1. L'ordre r de a modulo n est pair.

Hypothèse 2. $a^{r/2} + 1$ n'est pas divisible par n .

Théorème 7.

Soit $n = \prod_{i=1}^{\ell} p_i^{\alpha_i}$ la décomposition de n en produit de ℓ facteurs, avec p_i premier et $p_i \geq 3$. Alors la probabilité qu'un entier $a \in (\mathbb{Z}/n\mathbb{Z})^*$ vérifie l'hypothèse 1 et l'hypothèse 2 est supérieure à $1 - \frac{1}{2^{\ell-1}}$.

Remarques.

- Le pire cas se produit lorsqu'il y a seulement $\ell = 2$ facteurs premiers, comme dans le protocole RSA où $n = pq$. Dans ce cas au moins 50% des a conviennent.
- On peut énoncer un résultat combinatoire : le nombre d'éléments a satisfaisant les hypothèses 1 et 2 est supérieur à $(1 - \frac{1}{2^{\ell-1}})\varphi(n)$ parmi tous les $\varphi(n)$ éléments de $(\mathbb{Z}/n\mathbb{Z})^*$.
- La preuve n'est pas constructive, il n'existe pas de moyen simple de calculer l'ordre de a (c'est d'ailleurs le but de l'algorithme de Shor).

8.3. Préliminaires

On connaît déjà la proportion d'éléments a qui vérifient l'hypothèse 1. Par la proposition 9, la proportion d'éléments d'ordre pair est supérieure à $1 - \frac{1}{2^{\ell}}$.

On rappelle que l'objectif de l'algorithme de Shor est de trouver la période r d'un élément a et si r est pair d'écrire l'égalité $a^r \equiv 1 \pmod{n}$ sous la forme d'une factorisation :

$$(a^{r/2} - 1)(a^{r/2} + 1) \equiv 0 \pmod{n}.$$

Cela permet de trouver un facteur de n à condition que les termes de la factorisation ci-dessus soient non nuls.

Nous allons étudier en détails les racines carrées non-triviales de 1. Pourquoi ?

- Tout d'abord, par définition de l'ordre r , on ne peut pas avoir $a^{r/2} \equiv 1 \pmod{n}$. Ainsi $a^{r/2} - 1 \not\equiv 0 \pmod{n}$ et le premier terme de la factorisation est non nul.
- Supposons que l'hypothèse 2 soit vraie, c'est-à-dire $a^{r/2} + 1$ n'est pas divisible par n . Alors $a^{r/2} + 1 \not\equiv 0 \pmod{n}$ et ainsi le second terme est non nul. Dans ce cas on a $a^{r/2} \not\equiv -1 \pmod{n}$.
- Enfin, comme r est l'ordre de a , alors $(a^{r/2})^2 \equiv +1 \pmod{n}$ et avec les hypothèses 1 et 2, $a^{r/2}$ est une racine carrée non-triviale de 1.

Notons \mathcal{R} l'ensemble des racines carrées de 1 modulo n :

$$\mathcal{R} = \{x \in \mathbb{Z}/n\mathbb{Z} \mid x^2 \equiv 1 \pmod{n}\}.$$

On rappelle que :

- Une racine carrée est nécessairement inversible (son inverse est elle-même), donc $\mathcal{R} \subset (\mathbb{Z}/n\mathbb{Z})^*$.
- 1 et -1 sont les deux racines carrées évidentes.
- Il y a exactement 2^{ℓ} racines carrées : $\text{Card } \mathcal{R} = 2^{\ell}$, où ℓ est le nombre de facteurs premiers distincts de n . Et il y a donc $2^{\ell} - 2$ racines carrées non triviales.

8.4. Deux lemmes

Lemme 3.

Soit $s \geq 1$ et soit $y \in (\mathbb{Z}/n\mathbb{Z})^*$. L'équation $x^s \equiv y^s \pmod{n}$, d'inconnue x , possède toujours le même nombre de solutions quel que soit y .

Démonstration. Notons $\mathcal{S} = \{a_1, \dots, a_d\}$ l'ensemble des solutions de $x^s \equiv 1 \pmod{n}$. Alors

$$x^s \equiv y^s \pmod{n} \iff \left(\frac{x}{y}\right)^s \equiv 1 \pmod{n} \iff \frac{x}{y} \in \mathcal{S} \iff x = a_i y \text{ pour un } i \in \{1, \dots, d\}.$$

Les solutions de $x^s \equiv y^s$ sont donc les d éléments $\{a_1 y, \dots, a_d y\}$. □

Lemme 4.

S'il existe $x_0 \in (\mathbb{Z}/n\mathbb{Z})^$ tel que $x_0^s \equiv -1 \pmod{n}$ alors toute racine carrée z dans \mathcal{R} peut s'écrire sous la forme $z = y^s$, pour un certain $y \in (\mathbb{Z}/n\mathbb{Z})^*$.*

Démonstration. Le théorème des restes chinois fournit un isomorphisme entre $(\mathbb{Z}/n\mathbb{Z})^*$ et $(\mathbb{Z}/p_1^{\alpha_1}\mathbb{Z})^* \times \dots \times (\mathbb{Z}/p_\ell^{\alpha_\ell}\mathbb{Z})^*$. De plus, la proposition 10 donne la correspondance entre une racine carrée $z \in \mathcal{R} \subset (\mathbb{Z}/n\mathbb{Z})^*$ et un élément $(z_1, \dots, z_\ell) = (\pm 1, \pm 1, \dots, \pm 1)$ dans le produit des $(\mathbb{Z}/p_i^{\alpha_i}\mathbb{Z})^*$.

Le théorème des restes chinois fait correspondre x_0 à un élément (x_1, \dots, x_ℓ) et -1 à $(-1, \dots, -1)$. L'hypothèse $x_0^s \equiv -1 \pmod{n}$ se traduit donc en $x_i^s \equiv -1 \pmod{p_i^{\alpha_i}}$.

Si $z_i = +1$, alors on pose $y_i = +1$, si $z_i = -1$ alors on pose $y_i \equiv x_i$. Dans les deux cas on a $y_i^s \equiv z_i \pmod{p_i^{\alpha_i}}$ et par isomorphisme l'élément (y_1, \dots, y_ℓ) du groupe produit correspond à $y \in (\mathbb{Z}/n\mathbb{Z})^*$ tel que $y^s = z$. □

8.5. Puissance qui est une racine carrée

On ne veut pas compter les racines carrées mais le nombre d'éléments x tel que x^s soit une racine carrée. On rappelle que dans tous les cas on exclut la racine carrée 1 (qui n'a pas le bon ordre). Mais par contre parmi les x tel que x^s soit une racine carrée on veut distinguer la racine carrée -1 (qui est celle à éviter pour avoir l'hypothèse 2).

Lemme 5.

Soit $s \geq 1$. Notons

$$\mathcal{S}_s = \{x \in (\mathbb{Z}/n\mathbb{Z})^* \mid x^s \in \mathcal{R} \setminus \{1\}\}$$

et

$$\mathcal{S}'_s = \{x \in (\mathbb{Z}/n\mathbb{Z})^* \mid x^s \in \mathcal{R} \setminus \{1, -1\}\}.$$

Alors

$$\frac{\text{Card } \mathcal{S}'_s}{\text{Card } \mathcal{S}_s} \geq 1 - \frac{1}{2^\ell - 1}.$$

Démonstration. Dans le cas où l'équation $x^s \equiv -1 \pmod{n}$ n'a pas de solution, les deux ensembles \mathcal{S}_s et \mathcal{S}'_s sont égaux, donc $\frac{\text{Card } \mathcal{S}'_s}{\text{Card } \mathcal{S}_s} = 1$ et l'assertion est vraie.

Supposons qu'il existe x_0 tel que $x_0^s \equiv -1 \pmod{n}$. Soit $z \in \mathcal{R}$, alors par le lemme 4, l'équation $x^s \equiv z \pmod{n}$ est équivalente à l'équation $x^s \equiv y^s \pmod{n}$ (pour un certain y). Par le lemme 3, cette équation possède toujours le même nombre d de solutions (quel que soit y et donc aussi quel que soit $z \in \mathcal{R}$). Sachant que $\text{Card } \mathcal{R} = 2^\ell$ alors

$$\text{Card } \mathcal{S}'_s = d \times \text{Card}(\mathcal{R} \setminus \{1, -1\}) = d \times (2^\ell - 2).$$

De même

$$\text{Card } \mathcal{S}_s = d \times \text{Card}(\mathcal{R} \setminus \{1\}) = d \times (2^\ell - 1).$$

Donc $\frac{\text{Card } \mathcal{S}'_s}{\text{Card } \mathcal{S}_s} = \frac{2^\ell - 2}{2^\ell - 1} = 1 - \frac{1}{2^\ell - 1}$. □

8.6. Cas favorables

Nous terminons la preuve du théorème 7.

Preuve du théorème 7. Nous avons déjà estimé dans la proposition 9, le nombre d'éléments d'ordre pair, que l'on note $P(\mathbb{Z}/n\mathbb{Z})^*$:

$$\frac{\text{Card } P(\mathbb{Z}/n\mathbb{Z})^*}{\text{Card}(\mathbb{Z}/n\mathbb{Z})^*} \geq 1 - \frac{1}{2^\ell} = \frac{2^\ell - 1}{2^\ell}.$$

Nous allons maintenant déterminer la proportion d'éléments vérifiant l'hypothèse 2 parmi les éléments d'ordre pair. Pour tout ordre pair r , on note $s = \frac{r}{2}$. Ainsi si un élément x est d'ordre pair r , on a $(x^s)^2 \equiv 1 \pmod{n}$, avec $x^s \not\equiv 1 \pmod{n}$. Autrement dit $x^s \in \mathcal{R} \setminus \{1\}$, c'est-à-dire $x \in \mathcal{S}_s$. Ainsi l'ensemble des éléments d'ordre pair $P(\mathbb{Z}/n\mathbb{Z})^*$ (c'est-à-dire satisfaisant l'hypothèse 1) est l'union des \mathcal{S}_s (pour $1 \leq s \leq \frac{\varphi(n)}{2}$). De plus, l'ordre étant unique, ces ensembles sont disjoints.

Notons l'ensemble des cas favorables $F(\mathbb{Z}/n\mathbb{Z})^*$, c'est-à-dire les éléments satisfaisant l'hypothèse 1 et l'hypothèse 2. $F(\mathbb{Z}/n\mathbb{Z})^*$ est simplement l'union disjointe des \mathcal{S}'_s .

Par le lemme 5, on a pour chaque s :

$$\frac{\text{Card } \mathcal{S}'_s}{\text{Card } \mathcal{S}_s} \geq 1 - \frac{1}{2^\ell - 1} = \frac{2^\ell - 2}{2^\ell - 1}.$$

Comme cette inégalité est vraie pour les ensembles indexés par s , on obtient également pour l'union :

$$\frac{\text{Card } F(\mathbb{Z}/n\mathbb{Z})^*}{\text{Card } P(\mathbb{Z}/n\mathbb{Z})^*} \geq \frac{2^\ell - 2}{2^\ell - 1}.$$

Conclusion :

$$\frac{\text{Card } F(\mathbb{Z}/n\mathbb{Z})^*}{\text{Card}(\mathbb{Z}/n\mathbb{Z})^*} = \frac{\text{Card } F(\mathbb{Z}/n\mathbb{Z})^*}{\text{Card } P(\mathbb{Z}/n\mathbb{Z})^*} \times \frac{\text{Card } P(\mathbb{Z}/n\mathbb{Z})^*}{\text{Card}(\mathbb{Z}/n\mathbb{Z})^*} \geq \frac{2^\ell - 2}{2^\ell - 1} \times \frac{2^\ell - 1}{2^\ell} = \frac{2^\ell - 2}{2^\ell} = 1 - \frac{1}{2^{\ell-1}}.$$

□

Notes. L'explication du cas dans lequel l'ordre n'est pas une puissance de 2 est basée sur l'article *Shor's algorithm for factoring large integers* par C. Lavor, L.R.U. Manssur, R. Portugal. Il n'est pas facile de trouver une référence exacte et complète pour le comptage des cas favorables (théorème 7). La preuve donnée ici est reprise de « Introduction à l'informatique quantique » par Y. Leroyer et G. Sénizergues à l'Enseirb-Matmecca.

Transformée de Fourier discrète

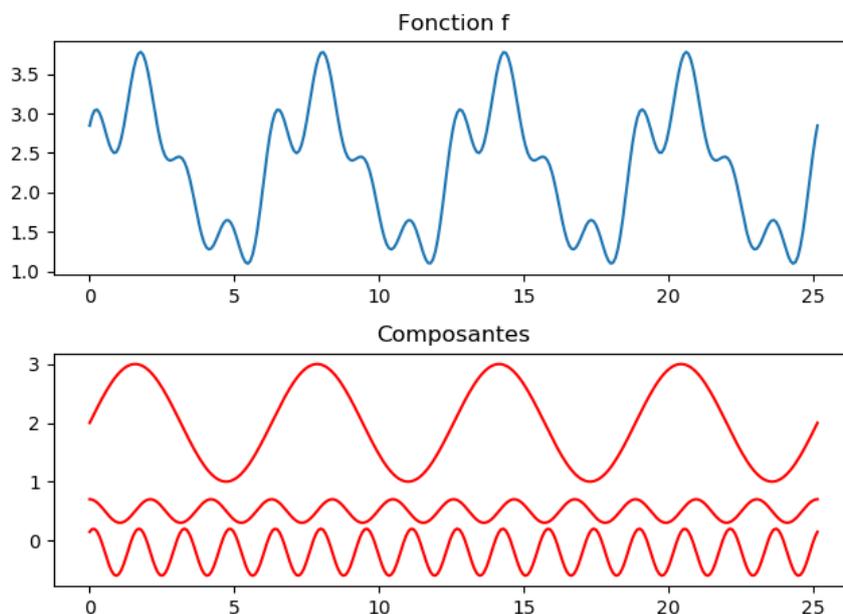
Vidéo ■ partie 15. Transformée de Fourier discrète (par des dessins)

Nous revenons sur l'outil principal de l'algorithme de Shor : la transformée de Fourier. Nous expliquons comment elle est construite, comment la réaliser par un circuit quantique et quelles sont ses autres applications.

1. Comprendre la transformée de Fourier

La transformée de Fourier, c'est la magie de pouvoir récupérer chacune des couleurs qu'on a mélangées dans un pot de peinture !

1.1. Les transformées de Fourier



Voici un exemple : on prend trois fonctions sinusoïdales f_1, f_2, f_3 (figure du bas) que l'on additionne pour obtenir une fonction compliquée $f(x) = f_1(x) + f_2(x) + f_3(x)$ (figure du haut). Alors la transformée de Fourier permet de retrouver chacune des composantes f_1, f_2, f_3 à partir de f .

Le monde de Fourier est assez vaste, voici un petit lexique :

- la transformée de Fourier concerne une fonction quelconque, elle se calcule à l'aide d'une intégrale,
- les séries de Fourier s'appliquent à des fonctions périodiques (comme ci-dessus),

- la transformée de Fourier discrète s'applique à une liste de nombres réels ou complexes,
- la transformée de Fourier discrète quantique est une variante de la précédente et transforme un qubit en un autre qubit.

1.2. La transformée de Fourier discrète classique

Nous allons expliquer le principe de la transformée de Fourier discrète (non quantique) et justifier comment elle permet de retrouver les périodes.

Voici la définition de la transformée de Fourier discrète. Soit (x_0, \dots, x_{n-1}) une suite de n nombres (ils peuvent être complexes mais pour nos exemples ce seront des réels). La transformée de Fourier discrète de (x_0, \dots, x_{n-1}) est la liste de nombres complexes (X_0, \dots, X_{n-1}) où chaque X_k est défini par :

$$X_k = \frac{1}{\sqrt{n}} \sum_{j=0}^{n-1} x_j e^{-2i\pi \frac{kj}{n}}.$$

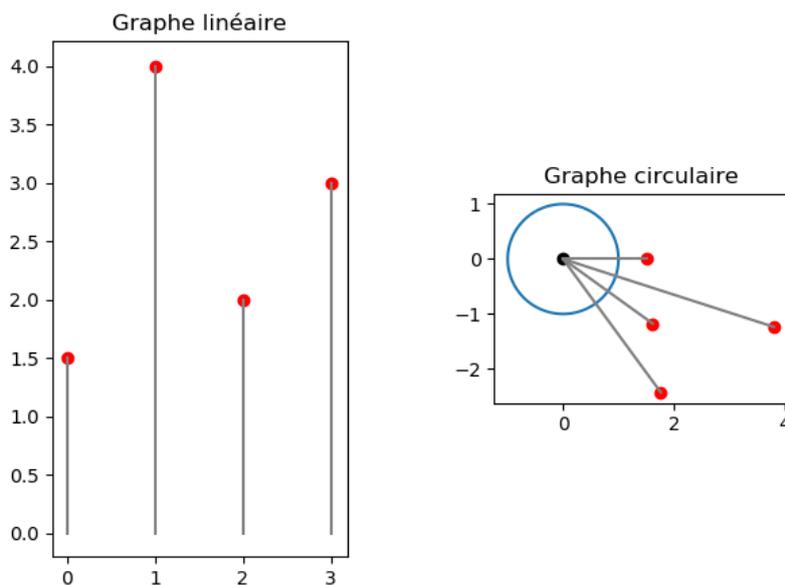
Si on note $\omega = e^{\frac{2i\pi}{n}}$ et $\omega^* = e^{-\frac{2i\pi}{n}}$, alors

$$X_k = \frac{1}{\sqrt{n}} (x_0(\omega^*)^{k \cdot 0} + x_1(\omega^*)^{k \cdot 1} + \dots + x_{n-1}(\omega^*)^{k \cdot (n-1)}).$$

La formule n'est donc pas si difficile à comprendre. Mais pourquoi vouloir transformer une suite de nombres par une opération aussi compliquée ?

1.3. Construction de la transformée de Fourier discrète

L'idée de la construction de la transformée de Fourier est très simple ! Nous avons des données x_j (pour nous $x_j \in \mathbb{R}$). Nous représentons traditionnellement x_j sous la forme d'un point (où l'ordonnée du point est la valeur x_j , les points étant placés de gauche à droite). Mais on peut aussi utiliser une représentation circulaire : chaque point est à une distance x_j de l'origine, et les points sont répartis dans le sens des aiguilles d'une montre.



Les données sont $(x_0, x_1, x_2, x_3) = (1.5, 4, 2, 3)$.

Figure de gauche : les données sont présentées sous la forme de points (j, x_j) .

Figure de droite les mêmes données sous forme d'écriture polaire : l'angle est proportionnel à j et le rayon est x_j .

Le point clé de la représentation circulaire est qu'on peut répartir les données sur un angle plus ou moins grand. Notons $2\pi t$ l'angle total d'étalement, alors plus précisément l'angle entre deux données est $\frac{2\pi t}{n}$.

Comment est calculée cette représentation circulaire? La donnée x_j correspond au point du plan situé à distance x_j de l'origine et faisant un angle $-\frac{2\pi t}{n} \times j$ avec l'horizontale : c'est donc le nombre complexe $z_j = x_j e^{-2i\pi \frac{t}{n} j}$. On voit apparaître le terme de la définition de la transformée de Fourier.

Voici différents étalements possibles, ils correspondent à différentes valeurs du paramètre t . Pour $t = \frac{1}{4}$ les données sont réparties sur un quart de cercle, pour $t = \frac{1}{2}$ un demi-cercle, pour $t = 1$ le cercle entier et pour $t > 1$ les données s'enroulent plusieurs fois autour du cercle.

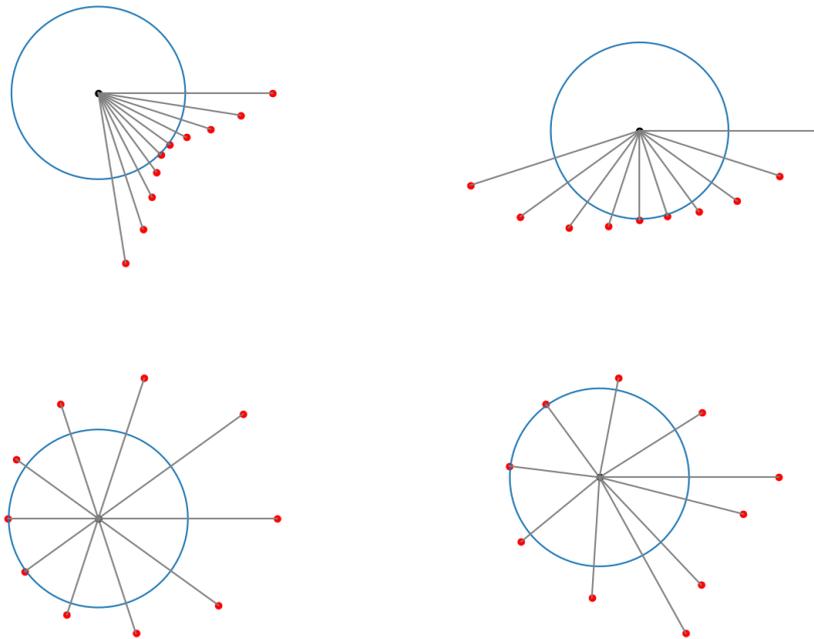
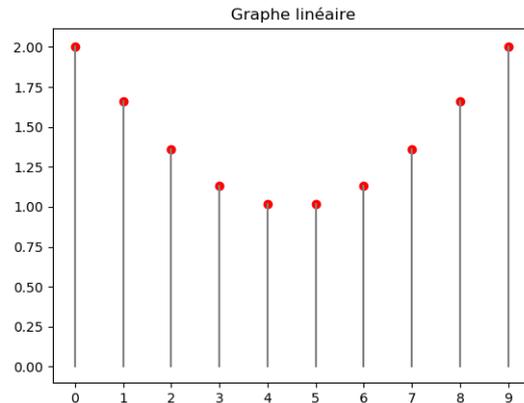
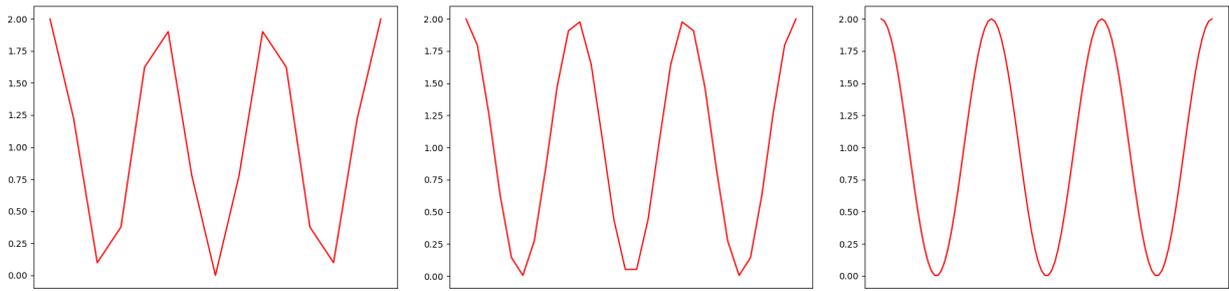


Figure de dessus : les données sont sous la forme de points (j, x_j) .
Figures du dessous : les mêmes données sous forme circulaire avec $t = \frac{1}{4}$, $t = \frac{1}{2}$,
 $t = 1$ et $t > 1$.

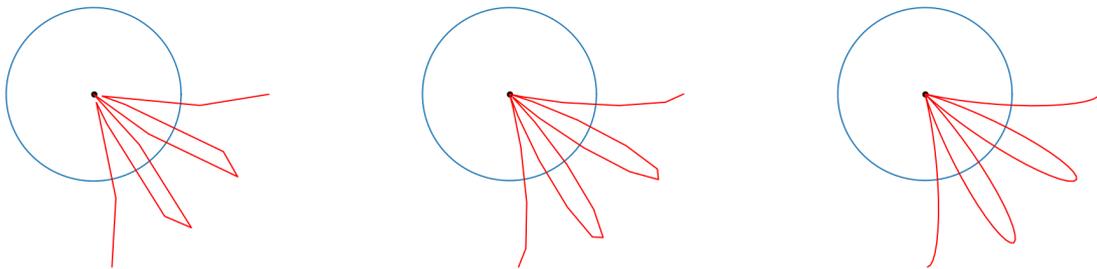
1.4. Enroulements particuliers

On remplace l'étude d'une fonction $f : [a, b] \rightarrow \mathbb{R}$ par l'étude de valeurs $x_j = f(a_j)$ pour une subdivision (a_j) de l'intervalle $[a, b]$. Pour avoir une meilleure précision, il suffit d'augmenter le nombre n de points dans la subdivision. On relie les points de ces données pour approcher le graphe de f .



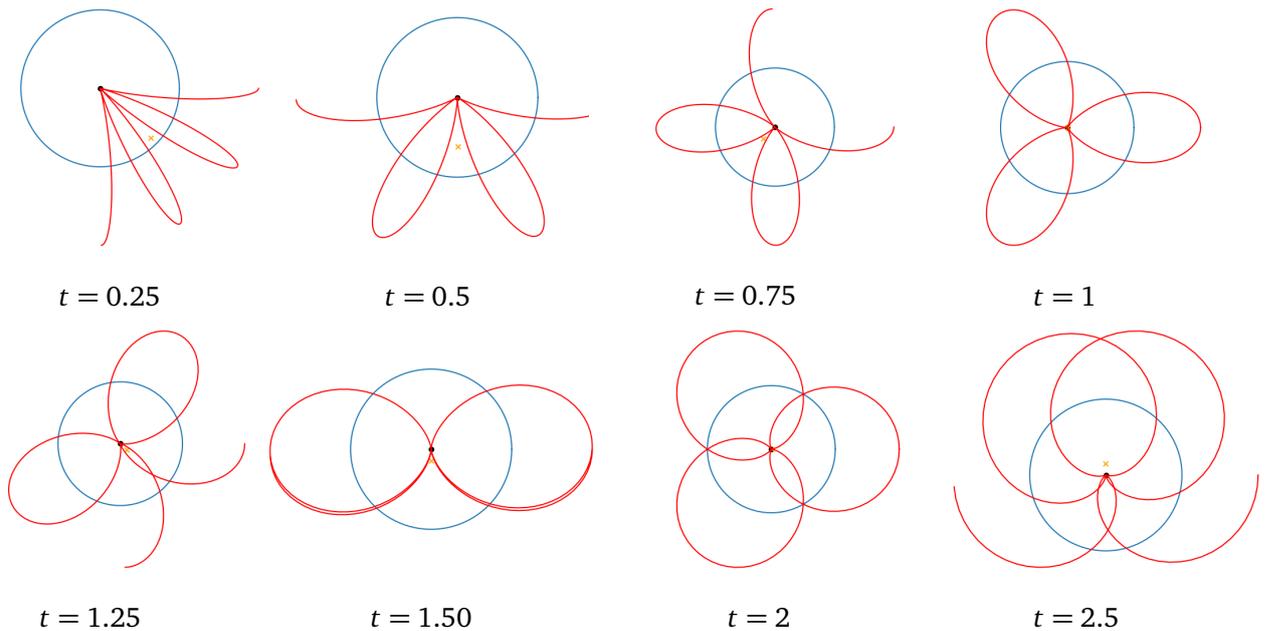
Fonction $f(x) = 1 + \cos(3x)$ sur l'intervalle $[0, 2\pi]$.
 Les données sont les $x_j = f(a_j)$ avec une subdivision de $n = 15$ points à gauche,
 $n = 30$ au centre, $n = 100$ à droite.

On peut aussi relier les points du graphe circulaire. Géométriquement on a ainsi enroulé le graphe de la fonction f , sur le cercle.



Les mêmes données sur le graphe circulaire avec $n = 15$ points à gauche,
 $n = 30$ au centre, $n = 100$ à droite.

Regardons maintenant les différents enroulements selon le paramètre d'étalement t . Tout d'abord on exclut les t avec $t < 1$ car le graphe ne s'enroule pas totalement autour du cercle. Pour la plupart des paramètres on obtient une jolie figure symétrique autour de l'origine, mais il y a des exceptions. Pour $t = 3$ on obtient une figure complètement décalée à droite. Que se passe-t-il en $t = 3$? Le graphe de f est périodique et pour cette valeur de t la courbe enroulée vient se superposer à elle-même lors des enroulements successifs.



$t = 0.25$

$t = 0.5$

$t = 0.75$

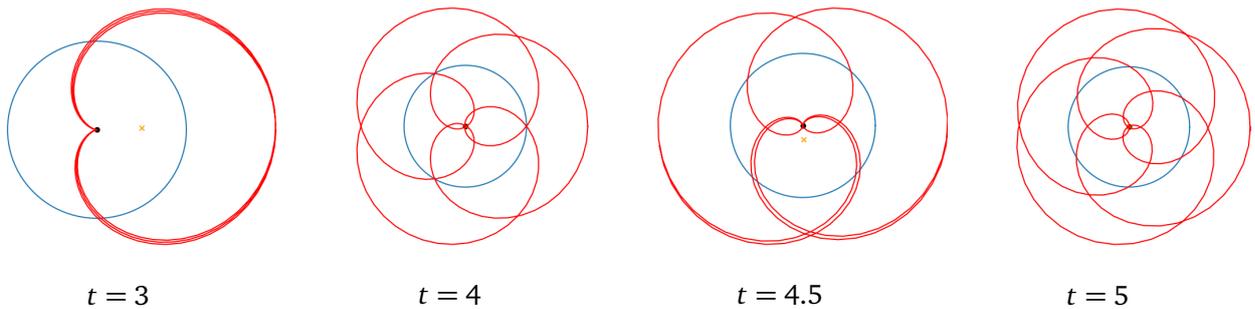
$t = 1$

$t = 1.25$

$t = 1.50$

$t = 2$

$t = 2.5$



Graphes de $f(x) = 1 + \cos(3x)$ sur l'intervalle $[0, 2\pi]$
enroulé sur le cercle pour différentes valeurs de t .

Pour $t = 3$ la figure n'est plus symétrique par rapport à l'origine.

La petite croix orange désigne le centre de gravité.

Ce cas particulier $t = 3$ est le cas qui nous intéresse ! Le paramètre $t = 3$ correspond à la période de notre fonction. En effet, la fonction $f(x) = 1 + \cos(3x)$ est $\frac{2\pi}{3}$ périodique et le paramètre spécial est $t = 3$. Pour une fonction $f(x) = 1 + \cos(\alpha x)$, la période est $\frac{2\pi}{\alpha}$ et le paramètre spécial est $t = \alpha$ (en fait t correspond à la fréquence qui est l'inverse de la période).

1.5. Centre de gravité

Est-ce que cette technique permet d'obtenir les différentes périodes d'un signal obtenu par superposition (comme dans l'exemple du tout début de ce chapitre) ? La réponse est oui ! Voyons comment repérer ces paramètres particuliers qui correspondent à des périodes. Pour la plupart des paramètres les figures obtenues sont symétriques par rapport à l'origine, donc le centre de gravité est proche de l'origine (en physique on parle d'interférence destructive). Par contre, pour certains paramètres particuliers, le centre de gravité s'éloigne de l'origine. Nous obtenons donc un critère numérique simple.

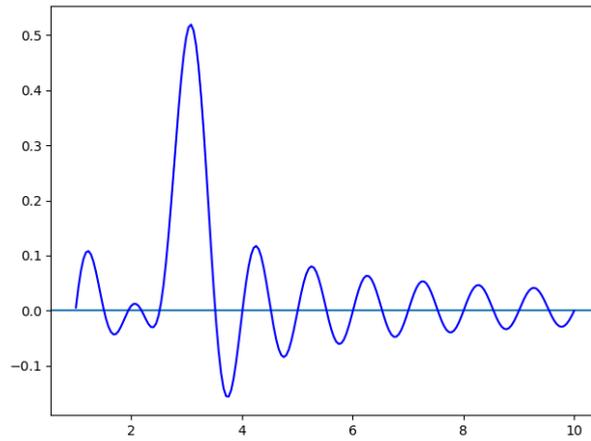
Le centre de gravité des points se calcule comme une moyenne des points :

$$X_t = \frac{1}{\sqrt{n}} \sum_{j=0}^{n-1} x_j e^{-2i\pi \frac{tj}{n}}.$$

Quelques remarques :

- Sur les dessins précédents ce centre de gravité était représenté par une petite croix orange.
- Nous préférons le choix du coefficient $\frac{1}{\sqrt{n}}$ (au lieu du coefficient $\frac{1}{n}$ du vrai centre de gravité).
- Pour $t = k$ la formule est exactement celle de la transformée de Fourier discrète des (x_j) .

Comment évolue X_t en fonction de t et comment repérer les paramètres particuliers ? Le nombre X_t est un nombre complexe, on se contente de regarder sa partie réelle $\text{Re}(X_t)$. La fonction $t \mapsto \text{Re}(X_t)$ mesure donc l'abscisse du centre de gravité.

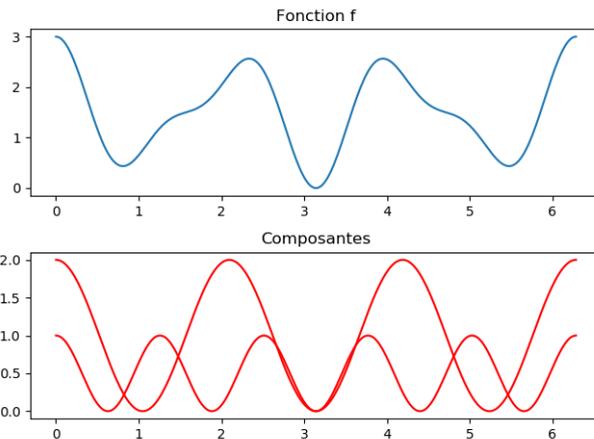


Graphe de la fonction $t \mapsto \text{Re}(X_t)$ qui mesure l'abscisse du centre de gravité de l'enroulement de la fonction $f(x) = 1 + \cos(3x)$ sur l'intervalle $[0, 2\pi]$ pour différentes valeurs de t . Le pic à $t = 3$ marque la rupture de symétrie.

Pour la plupart des valeurs de t , $\text{Re}(X_t)$ est proche de 0, les pics correspondent à la rupture de symétrie centrale et déterminent les périodes des composantes de la fonction.

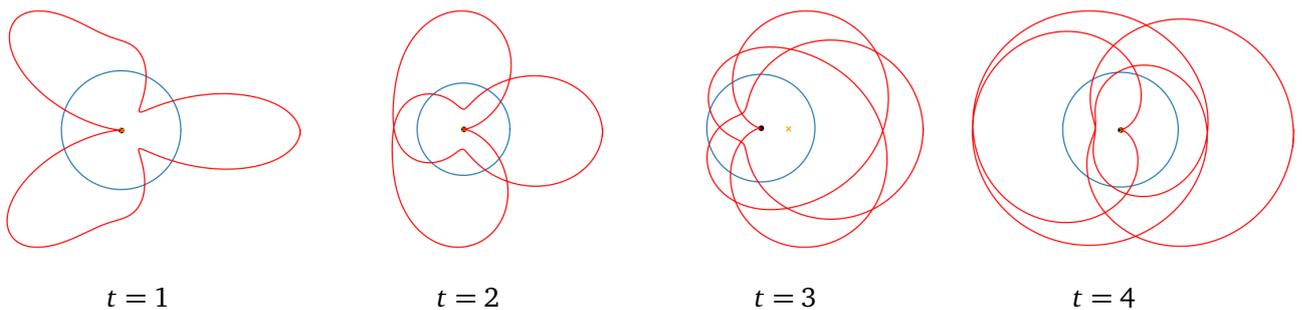
1.6. Autre exemple

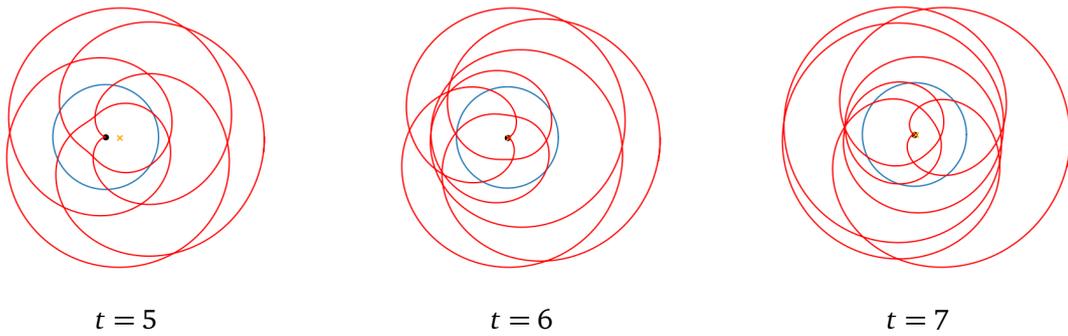
Soient $f_1(x) = 1 + \cos(3x)$ et $f_2(x) = \frac{1}{2}(1 + \cos(5x))$ et leur somme $f(x) = f_1(x) + f_2(x)$ définie sur l'intervalle $[0, 2\pi]$. À partir de f , nous souhaitons retrouver les périodes $\frac{2\pi}{3}$ et $\frac{2\pi}{5}$ de ses deux composantes.



Le graphe de la fonction $f(x) = f_1(x) + f_2(x)$ sur $[0, 2\pi]$ avec $f_1(x) = 1 + \cos(3x)$ et $f_2(x) = \frac{1}{2}(1 + \cos(5x))$.

On enroule le graphe de f sur un cercle, selon différentes valeurs de t .



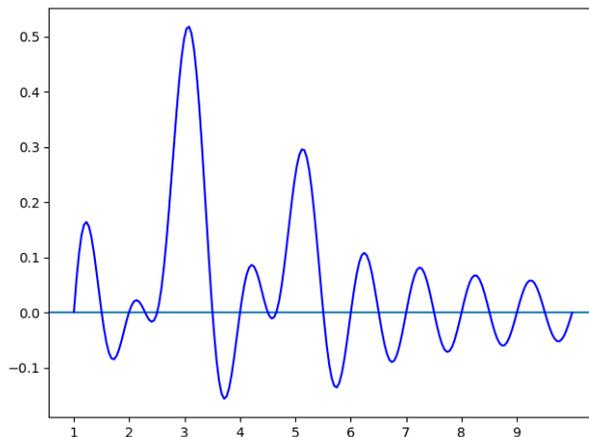


$t = 5$ $t = 6$ $t = 7$
 Enroulement du graphe de f pour différentes valeurs de t .
 Noter le décalage du centre de gravité (la croix orange) en $t = 3$ et $t = 5$.

Le centre de gravité se calcule selon la formule de la transformée de Fourier discrète, pour nos paramètres $t = 1, t = 2, \dots$

$$X_t = \frac{1}{\sqrt{n}} \sum_{j=0}^{n-1} x_j e^{-2i\pi \frac{tj}{n}}$$

où $x_j = f\left(\frac{j}{n}\right)$



Graphe de la fonction $t \mapsto \text{Re}(X_t)$ qui mesure l'abscisse du centre de gravité de l'enroulement de la fonction $f(x)$ sur l'intervalle $[0, 2\pi]$ pour différentes valeurs de t .
 On remarque des pics en $t = 3$ et $t = 5$.

1.7. L'inverse de la transformée de Fourier discrète

On a donc vu l'intérêt de la transformée de Fourier : elle permet de retrouver les caractéristiques d'une fonction (ou d'une série de données). Mais de plus cette transformation ne perd pas d'information. D'un point de vue mathématique la transformation est bijective. On peut retrouver les (x_k) connaissant les (X_j) par une formule similaire (seul le signe de l'exponentielle change) :

$$x_k = \frac{1}{\sqrt{n}} \sum_{j=0}^{n-1} X_j e^{+2i\pi \frac{kj}{n}}$$

Conclusion : la transformée de Fourier discrète transforme une liste de n nombres en une autre liste de n nombres. Cette transformation est bijective et permet en particulier de déterminer la période d'un signal périodique.

1.8. La transformée de Fourier discrète quantique

La transformée de Fourier discrète quantique est par définition cette variante de la transformée de Fourier discrète :

$$\hat{F} |\underline{k}\rangle = \frac{1}{\sqrt{2^n}} \sum_{j=0}^{2^n-1} e^{2i\pi \frac{k \cdot j}{2^n}} |j\rangle$$

Les différences sont les suivantes :

- Par convention on choisit un signe « + » pour l'exposant de la transformée quantique (comme pour la transformée de Fourier discrète *inverse* classique).
- Les données sont remplacées par les qubits de bases $|0\rangle, |1\rangle, \dots$
- En conséquence le nombre total de données est une puissance de 2. Pour les n -qubits, il y a 2^n qubits de base $|0\rangle, \dots, |2^n-1\rangle$.
- La fonction \hat{F} s'étend par linéarité à n'importe quel n -qubit. Si $|\psi\rangle = \alpha_0 |0\rangle + \alpha_1 |1\rangle + \dots$ alors $\hat{F} |\psi\rangle = \alpha_0 \hat{F} |0\rangle + \alpha_1 \hat{F} |1\rangle + \dots$

2. Écritures de la transformée de Fourier

2.1. Définition de la transformée de Fourier

On rappelle la définition de la transformée de Fourier discrète quantique pour un n -qubit de base $|\underline{k}\rangle$:

$$\hat{F} |\underline{k}\rangle = \frac{1}{\sqrt{2^n}} \sum_{j=0}^{2^n-1} e^{2i\pi \frac{k \cdot j}{2^n}} |j\rangle. \quad (1)$$

Dans toute la suite du chapitre la notion de « transformée de Fourier » désigne la transformée de Fourier discrète quantique. On renvoie au chapitre « Algorithme de Shor » pour les détails et les premières propriétés.

2.2. Factorisation de la transformée de Fourier

Voici le résultat fondamental de ce chapitre qui permettra de réaliser le circuit quantique de la transformée de Fourier.

Théorème 1.

$$\hat{F} |\underline{k}\rangle = \frac{1}{\sqrt{2^n}} \prod_{\ell=1}^n \left(|0\rangle + e^{2i\pi \frac{k}{2^\ell}} |1\rangle \right)$$

Notation : on note le produit sous la forme $\prod_{\ell=1}^n$ afin de ne pas effrayer le lecteur, alors qu'en toute rigueur il s'agit d'un produit tensoriel :

$$\hat{F} |\underline{k}\rangle = \frac{1}{\sqrt{2^n}} \bigotimes_{\ell=1}^n \left(|0\rangle + e^{2i\pi \frac{k}{2^\ell}} |1\rangle \right).$$

De façon développée la factorisation s'écrit :

$$\hat{F} |\underline{k}\rangle = \frac{1}{\sqrt{2^n}} \left(|0\rangle + e^{2i\pi \frac{k}{2}} |1\rangle \right) \otimes \left(|0\rangle + e^{2i\pi \frac{k}{2^2}} |1\rangle \right) \otimes \dots \otimes \left(|0\rangle + e^{2i\pi \frac{k}{2^n}} |1\rangle \right).$$

Exemple.

- Pour $n = 1$ le produit est réduit à un seul élément, $\hat{F} |\underline{k}\rangle = \frac{1}{\sqrt{2}} \left(|0\rangle + e^{2i\pi \frac{k}{2}} |1\rangle \right)$. Ainsi pour $k = 0$, $\hat{F} |0\rangle = \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle)$ et pour $k = 1$, $e^{i\pi} = -1$ donc $\hat{F} |1\rangle = \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle)$. On retrouve bien que pour $n = 1$, la transformée de Fourier correspond à la porte de Hadamard H .

- Pour $n = 2$, on écrit pour $\ell = 1$, $e^{2i\pi \frac{k}{2}} = (-1)^k$ et pour $\ell = 2$, $e^{2i\pi \frac{k}{2^2}} = i^k$, la factorisation s'écrit donc

$$\hat{F} |k\rangle = \frac{1}{2} (|0\rangle + (-1)^k |1\rangle) (|0\rangle + i^k |1\rangle).$$

- Pour $n = 3$, notons $\omega = e^{\frac{2i\pi}{2^3}} = e^{\frac{i\pi}{4}}$, la factorisation s'écrit :

$$\hat{F} |k\rangle = \frac{1}{\sqrt{8}} (|0\rangle + (-1)^k |1\rangle) (|0\rangle + i^k |1\rangle) (|0\rangle + \omega^k |1\rangle).$$

Mais comme $\omega^2 = i$ et $\omega^4 = -1$, on peut aussi l'écrire :

$$\hat{F} |k\rangle = \frac{1}{\sqrt{8}} (|0\rangle + \omega^{2 \cdot k} |1\rangle) (|0\rangle + \omega^{2 \cdot k} |1\rangle) (|0\rangle + \omega^k |1\rangle).$$

La preuve du théorème repose sur l'écriture binaire des entiers. Prenons j un entier (avec $0 \leq j < 2^n$) et écrivons sa décomposition suivant les puissances de 2 :

$$j = \sum_{\ell=0}^{n-1} j_\ell 2^\ell = j_{n-1} \cdot 2^{n-1} + \dots + j_2 \cdot 2^2 + j_1 \cdot 2 + j_0$$

avec $j_\ell = 0$ ou $j_\ell = 1$, pour $\ell = 0, \dots, n-1$ et notons comme d'habitude $\underline{j} = j_{n-1} \dots j_2 \cdot j_1 \cdot j_0$ l'écriture binaire de j .

Par définition :

$$|j\rangle = |j_{n-1} \dots j_2 \cdot j_1 \cdot j_0\rangle = |j_{n-1}\rangle \dots |j_2\rangle \cdot |j_1\rangle \cdot |j_0\rangle.$$

Démonstration. Nous partons du produit

$$\prod_{\ell=1}^n (|0\rangle + e^{2i\pi \frac{k}{2^\ell}} |1\rangle) \tag{2}$$

que nous allons développer. Nous allons montrer que le coefficient devant le terme $|j\rangle$ est le même que celui de la définition (1) de la transformée de Fourier.

Récrivons le produit (2) sous une forme plus explicite :

$$\begin{aligned} & (|0\rangle + e^{2i\pi \frac{k}{2^1}} |1\rangle) \\ & \times (|0\rangle + e^{2i\pi \frac{k}{2^2}} |1\rangle) \\ & \times (|0\rangle + e^{2i\pi \frac{k}{2^3}} |1\rangle) \\ & \times \dots \end{aligned} \tag{3}$$

Lorsque l'on développe cette expression, on obtient des termes qui résultent du choix pour chaque ligne de (3) d'un des deux éléments $|0\rangle$ ou $e^{2i\pi \frac{k}{2^\ell}} |1\rangle$.

Par exemple, si on choisit $|0\rangle$ à chaque ligne de (3), alors on obtient le terme $|0\rangle \cdot |0\rangle \dots |0\rangle = |0.0 \dots 0\rangle = |\underline{0}\rangle$ avec comme coefficient 1, exactement comme le coefficient $|\underline{0}\rangle$ de l'expression (1).

Revenons au cas général. Pour la première ligne de (3), soit on choisit le facteur $|0\rangle$ et alors on va obtenir un terme qui commence par $|0\rangle : |j\rangle = |0 \dots\rangle$, soit on choisit le facteur $e^{2i\pi \frac{k}{2^1}} |1\rangle$ et on va obtenir un terme qui commence par $|1\rangle : |j\rangle = |1 \dots\rangle$. On peut regrouper ces deux cas en une seule formule : notons j_{n-1} un bit (qui vaut 0 ou 1) alors le facteur de la première ligne s'écrit $e^{2i\pi \frac{k \cdot j_{n-1}}{2^1}} |j_{n-1}\rangle$. En effet si $j_{n-1} = 0$ alors ce facteur vaut $e^{2i\pi \cdot 0} |0\rangle$, c'est donc $|0\rangle$, et si $j_{n-1} = 1$ c'est $e^{2i\pi \frac{k \cdot 1}{2^1}} |1\rangle$. Ce facteur va produire un terme qui commence par le bit $j_{n-1} : |j\rangle = |j_{n-1} \dots\rangle$. Ainsi le choix du facteur de la première ligne correspond au premier bit de \underline{j} (celui le plus à gauche).

Plus généralement, le facteur de la ligne ℓ de (3) s'écrit $e^{2i\pi \frac{k \cdot j_{n-\ell}}{2^\ell}} |j_{n-\ell}\rangle$. En effet, si $j_{n-\ell} = 0$ alors c'est $|0\rangle$ et si $j_{n-\ell} = 1$ alors c'est bien $e^{2i\pi \frac{k}{2^\ell}} |1\rangle$. Ce facteur va produire un terme avec le bit $j_{n-\ell} : |j\rangle = |\dots j_{n-\ell} \dots\rangle$.

Ainsi le terme qui correspond au qubit $|j\rangle$ dans le développement de (3) est le produit des facteurs $e^{2i\pi \frac{k \cdot j_{n-\ell}}{2^\ell}} |j_{n-\ell}\rangle$ pour chacune des lignes. Calculons ce terme :

$$\begin{aligned} \prod_{\ell=1}^n \left(e^{2i\pi \frac{k \cdot j_{n-\ell}}{2^\ell}} |j_{n-\ell}\rangle \right) &= \left(\prod_{\ell=1}^n e^{2i\pi \frac{k \cdot j_{n-\ell}}{2^\ell}} \right) |j_{n-1} \dots j_2 \cdot j_1 \cdot j_0\rangle \\ &= e^{2i\pi k \cdot \sum_{\ell=1}^n \frac{j_{n-\ell}}{2^\ell}} |j\rangle \\ &= e^{2i\pi \frac{k}{2^n} \cdot \sum_{\ell=1}^n j_{n-\ell} 2^{n-\ell}} |j\rangle \\ &= e^{2i\pi \frac{k}{2^n} \cdot \sum_{\ell'=0}^{n-1} j_{\ell'} 2^{\ell'}} |j\rangle \\ &= e^{2i\pi \frac{k}{2^n} \cdot j} |j\rangle \end{aligned}$$

Ainsi le coefficient de $|j\rangle$ du développement de la formule (2) est $e^{2i\pi \frac{k}{2^n} \cdot j}$ qui est exactement celui du coefficient de $|j\rangle$ dans la définition de la transformée de Fourier (1).

Ceci étant vrai quel que soit le qubit $|j\rangle$, on a donc bien :

$$\hat{F} |k\rangle = \frac{1}{\sqrt{2^n}} \sum_{j=0}^{2^n-1} e^{2i\pi \frac{k \cdot j}{2^n}} |j\rangle = \frac{1}{\sqrt{2^n}} \prod_{\ell=1}^n (|0\rangle + e^{2i\pi \frac{k}{2^\ell}} |1\rangle).$$

□

2.3. Variante

Commençons par introduire l'écriture binaire pour un nombre $0 \leq x < 1$.

$$0..j_1 \cdot j_2 \dots j_n = \frac{j_1}{2} + \frac{j_2}{2^2} + \dots + \frac{j_n}{2^n} = \sum_{\ell=1}^n \frac{j_\ell}{2^\ell}.$$

La notation est $0..j_1 \cdot j_2 \dots j_n$: les points séparent les bits, le double point symbolise la virgule car en écriture décimale le nombre s'écrit $0.abc \dots$.

Par exemple $x = 0.625$ (en écriture décimale) s'écrit en écriture binaire $x = 0..1.0.1$ car $0.625 = \frac{1}{2} + \frac{0}{4} + \frac{1}{8}$. Nous reformulons le théorème 1 de factorisation en jouant sur le passage de l'écriture binaire de l'entier k à l'écriture binaire d'un nombre à virgule.

Corollaire 1.

Si $|k\rangle = |k_{n-1} \dots k_1 \cdot k_0\rangle$

$$\hat{F} |k\rangle = \frac{1}{\sqrt{2^n}} \prod_{\ell=1}^n (|0\rangle + e^{2i\pi 0..k_{\ell-1} \dots k_0} |1\rangle).$$

Autrement dit

$$\hat{F} |k\rangle = \frac{1}{\sqrt{2^n}} (|0\rangle + e^{2i\pi 0..k_0} |1\rangle) \otimes (|0\rangle + e^{2i\pi 0..k_1 \cdot k_0} |1\rangle) \otimes \dots \otimes (|0\rangle + e^{2i\pi 0..k_{n-1} \dots k_1 \cdot k_0} |1\rangle).$$

Démonstration. Notons tout d'abord que pour n'importe quel entier p , $e^{2i\pi p} = 1$.

Alors

$$\begin{aligned} \frac{k}{2^\ell} &= \frac{k_{n-1}2^{n-1} + \dots + k_22^2 + k_12 + k_0}{2^\ell} \\ &= \underbrace{k_{n-1}2^{n-1-\ell} + \dots + k_\ell}_{\text{partie entière}} + \underbrace{\frac{k_{\ell-1}}{2} + \dots + \frac{k_0}{2^\ell}}_{\text{partie décimale}} \\ &= p + 0.k_{\ell-1} \dots k_0 \end{aligned}$$

Ainsi

$$e^{2i\pi \frac{k}{2^\ell}} = e^{2i\pi(p+0.k_{\ell-1} \dots k_0)} = e^{2i\pi 0.k_{\ell-1} \dots k_0}.$$

Par exemple :

- pour $\ell = 1$, $e^{2i\pi \frac{k}{2}} = e^{2i\pi 0.k_0}$;
- pour $\ell = 2$, $e^{2i\pi \frac{k}{4}} = e^{2i\pi 0.k_1.k_0}$;
- et pour $\ell = n$, $e^{2i\pi \frac{k}{2^n}} = e^{2i\pi 0.k_{n-1} \dots k_1.k_0}$.

Le théorème 1 donne alors la formule voulue. □

3. Circuit de la transformation Fourier

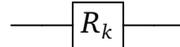
Nous allons construire un circuit quantique qui réalise la transformée de Fourier.

3.1. Porte R_k

Soit $R_k \in M_2(\mathbb{C})$ la matrice unitaire suivante :

$$R_k = \begin{pmatrix} 1 & 0 \\ 0 & e^{\frac{2i\pi}{2^k}} \end{pmatrix}.$$

Notons aussi R_k la porte quantique correspondante :

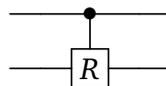


- Pour $n = 0$, $e^{\frac{2i\pi}{1}} = 1$ donc $R_0 = I$: la transformation est l'identité.
- Pour $n = 1$, $e^{\frac{2i\pi}{2}} = -1$ donc la transformation est $R_1 = Z$ ($|0\rangle \mapsto |0\rangle$, $|1\rangle \mapsto -|1\rangle$).
- Pour $n = 2$, $e^{\frac{2i\pi}{4}} = i$, la porte R_2 est aussi appelée porte S ($|0\rangle \mapsto |0\rangle$, $|1\rangle \mapsto i|1\rangle$).
- Pour $n = 3$, $e^{\frac{2i\pi}{8}} = e^{\frac{i\pi}{4}}$, la porte R_3 est aussi appelée porte T ($|0\rangle \mapsto |0\rangle$, $|1\rangle \mapsto e^{\frac{i\pi}{4}}|1\rangle$).

$$R_0 = I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad R_1 = Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad R_2 = S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \quad R_3 = T = \begin{pmatrix} 1 & 0 \\ 0 & e^{\frac{i\pi}{4}} \end{pmatrix}$$

3.2. Contrôle des portes

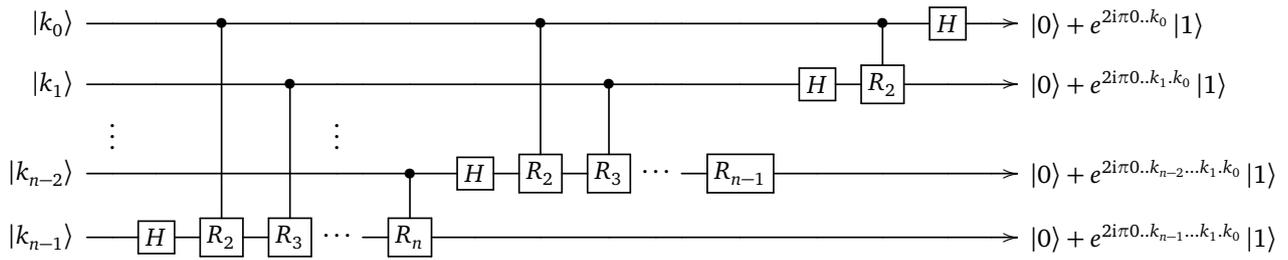
Chaque porte R va être « contrôlée » par un autre qubit qui déterminera si on applique ou non la porte R .



Si le premier qubit est $|0\rangle$, on ne change pas le second qubit, si le premier qubit est $|1\rangle$, on applique la porte R au second qubit restant.



3.3. Circuit



Justifions que ce circuit convient en regardant les exemples avec peu de lignes.

Cas $n = 1$. Le circuit est simplement réduit à une seule ligne contenant la seule porte H :

$$|k_0\rangle \xrightarrow{H} |\psi_0\rangle$$

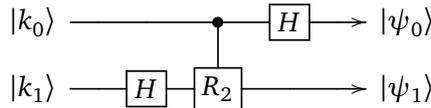
Donc $|\psi_0\rangle = H |k_0\rangle$. Ainsi $|\psi_0\rangle = |0\rangle + |1\rangle$ si $k_0 = 0$, et $|\psi_0\rangle = |0\rangle - |1\rangle$ si $k_0 = 1$ (aux coefficients $\sqrt{2}$ près).

On résume cela en une seule formule pour les deux cas :

$$|\psi_0\rangle = |0\rangle + (-1)^{k_0} |1\rangle = |0\rangle + e^{2i\pi \frac{k_0}{2}} |1\rangle = |0\rangle + e^{2i\pi 0..k_0} |1\rangle,$$

car on rappelle que $e^{2i\pi 0..k_0} = e^{2i\pi \frac{k_0}{2}} = e^{i\pi k_0} = (-1)^{k_0}$ qui vaut $+1$ si $k_0 = 0$ et -1 si $k_0 = 1$.

Cas $n = 2$.



Il est clair que $|\psi_0\rangle$ est le même qubit que dans le cas $n = 1$ ci-dessus.

Calculons le qubit $|\psi_1\rangle$. Si $|k_0\rangle = |0\rangle$ alors la seconde ligne est juste une porte H car la porte R_2 n'est pas activée. Donc si $k_0 = 0$, on a comme ci-dessus pour le cas $n = 1$:

$$|\psi_1\rangle = H |k_1\rangle = |0\rangle + e^{2i\pi \frac{k_1}{2}} |1\rangle \tag{4}$$

Par contre si $k_0 = 1$ la porte R_2 est activée et la seconde ligne devient :

$$|k_1\rangle \xrightarrow{H} \xrightarrow{R_2} |\psi_1\rangle$$

Ainsi :

$$\begin{aligned} |\psi_1\rangle &= R_2(H |k_1\rangle) \\ &= R_2\left(|0\rangle + e^{2i\pi \frac{k_1}{2}} |1\rangle\right) \\ &= R_2 |0\rangle + e^{2i\pi \frac{k_1}{2}} R_2 |1\rangle \\ &= |0\rangle + e^{2i\pi \frac{k_1}{2}} \cdot e^{2i\pi \frac{1}{4}} |1\rangle \end{aligned} \tag{5}$$

On peut regrouper les cas $k_0 = 0$ et $k_0 = 1$ des équations (4) et (5) en une seule équation :

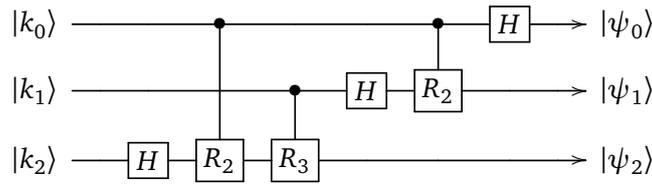
$$|\psi_1\rangle = |0\rangle + e^{2i\pi \frac{k_1}{2}} \cdot e^{2i\pi \frac{k_0}{4}} |1\rangle \tag{6}$$

où l'on a utilisé que $e^{2i\pi \frac{k_0}{4}}$ vaut 1 si $k_0 = 0$ et $e^{2i\pi \frac{1}{4}}$ si $k_0 = 1$.

Mais comme $e^{2i\pi \frac{k_1}{2}} \cdot e^{2i\pi \frac{k_0}{4}} = e^{2i\pi 0..k_1.k_0}$, on obtient bien :

$$|\psi_1\rangle = |0\rangle + e^{2i\pi 0..k_1.k_0} |1\rangle.$$

Cas $n = 3$.



Les calculs s'effectuent sur le même principe : $|\psi_0\rangle$ et $|\psi_1\rangle$ sont les mêmes que précédemment et

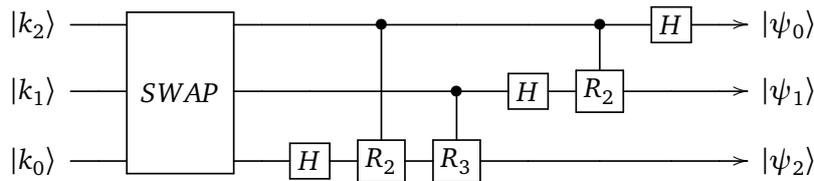
$$|\psi_2\rangle = |0\rangle + e^{2i\pi \frac{k_2}{2}} \cdot e^{2i\pi \frac{k_1}{4}} \cdot e^{2i\pi \frac{k_0}{8}} |1\rangle = |0\rangle + e^{2i\pi 0..k_2.k_1.k_0} |1\rangle.$$

Le calcul pour un n quelconque se fait par récurrence et prouve que le circuit calcule la transformée de Fourier.

3.4. Porte SWAP

Il faut faire une petite modification au circuit de la transformée de Fourier qui, en effet, ne respecte pas notre convention habituelle sur l'ordre d'écriture des qubits. Dans notre circuit le qubit en entrée est $|k_0\rangle \otimes |k_1\rangle \otimes \dots \otimes |k_{n-1}\rangle$. Mais si l'écriture binaire de \underline{k} est $k_{n-1} \dots k_1.k_0$ alors $|\underline{k}\rangle = |k_{n-1}\rangle \otimes \dots \otimes |k_1\rangle \otimes |k_0\rangle$. Pour obtenir l'écriture voulue il suffit de renverser les qubits. Cela se fait avec une porte SWAP que l'on a vue lors du chapitre « Portes quantiques ».

Ainsi le circuit complet pour l'exemple de $n = 3$ devient :



En incluant une porte SWAP, nous avons construit un circuit qui réalise la transformée de Fourier :

$$|\underline{k}\rangle \xrightarrow{\hat{F}} \hat{F} |\underline{k}\rangle$$

4. Estimation de phase

La dernière application de la transformée de Fourier que nous allons voir est « l'estimation de phase », c'est le nom physique utilisé pour parler de la détermination d'une valeur propre d'une matrice unitaire. Cette section ne revient pas sur les détails et les motivations concernant les valeurs propres : on renvoie pour cela à un cours d'algèbre sur la réduction des endomorphismes.

4.1. Valeur propre

Définition.

Soit $A \in M_n(\mathbb{C})$ une matrice. Le scalaire $\lambda \in \mathbb{C}$ est une **valeur propre** associée au **vecteur propre** X , si X n'est pas le vecteur nul et :

$$AX = \lambda X$$

Les valeurs propres et les vecteurs propres jouent un rôle fondamental dans l'étude des matrices. Rappelons juste ici qu'une matrice unitaire (c'est-à-dire vérifiant $A^*A = I$) est diagonalisable, c'est-à-dire équivalente à

une matrice diagonale :

$$\begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_n \end{pmatrix}$$

où justement les éléments λ_i sont les valeurs propres de A .

Mettons en avant deux propriétés des valeurs propres.

Lemme 1.

Soit A une matrice. Si λ est une valeur propre associée au vecteur propre X alors $A^n X = \lambda^n X$.

Autrement dit λ^n est une valeur propre de A^n .

Démonstration. La preuve se fait par récurrence en se calquant sur le modèle suivant où $n = 2$:

$$A^2 X = A(AX) = A(\lambda X) = \lambda(AX) = \lambda(\lambda X) = \lambda^2 X.$$

□

Voici le second résultat qui concerne uniquement les matrices unitaires.

Lemme 2.

Soit $A \in M_n(\mathbb{C})$ une matrice unitaire. Si λ est une valeur propre alors $|\lambda| = 1$.

Ainsi on peut écrire $\lambda = e^{2i\pi\theta}$ et la valeur propre est déterminé par sa « phase » θ (la phase étant le nom donné par les physiciens à l'argument). Pour comprendre la preuve, rappelons quelques propriétés (voir le chapitre « Portes quantiques ») :

- A unitaire signifie $A^* A = I$.
- Le produit scalaire est anti-linéaire à gauche et linéaire à droite, donc pour $\lambda \in \mathbb{C}$:

$$\langle \lambda u | v \rangle = \lambda^* \langle u | v \rangle \quad \text{et} \quad \langle u | \lambda v \rangle = \lambda \langle u | v \rangle$$

et permet de calculer la norme : $\|u\|^2 = \langle u | u \rangle$.

- Une matrice unitaire préserve le produit scalaire : $\langle Au | Av \rangle = \langle u | v \rangle$.

Démonstration. Soit λ une valeur propre associée au vecteur propre X .

$$\begin{aligned} \langle AX | AX \rangle = \langle X | X \rangle &\implies \langle \lambda X | \lambda X \rangle = \langle X | X \rangle \\ &\implies \lambda^* \langle X | \lambda X \rangle = \langle X | X \rangle \\ &\implies \lambda^* \lambda \langle X | X \rangle = \langle X | X \rangle \\ &\implies |\lambda|^2 \cdot \|X\|^2 = \|X\|^2 \\ &\implies |\lambda|^2 = 1 \\ &\implies |\lambda| = 1 \end{aligned}$$

On a utilisé les propriétés rappelées précédemment ainsi que le fait qu'un vecteur propre est non nul (donc $\|X\| \neq 0$). □

4.2. Problème de l'estimation de la phase

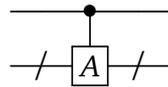
D'un point de vue mathématique le problème est le suivant. On nous donne une matrice $M \in M_N(\mathbb{C})$ unitaire et un vecteur propre X_0 . Il s'agit de calculer la valeur propre $\lambda_0 = e^{2i\pi\theta_0}$ associée à ce vecteur propre. Dans notre situation informatique, on a $N = 2^n$ et $A \in M_{2^n}(\mathbb{C})$ est une matrice unitaire. Le vecteur propre est écrit sous la forme d'un n -qubit $|\psi_0\rangle$. Le but reste toujours de déterminer la valeur propre λ_0 en calculant θ_0 .

4.3. Porte cA

Nous généralisons la porte $CNOT$, qui est une porte cX , c'est-à-dire une porte X conditionnelle. Soit A une matrice unitaire de $M_{2^n}(\mathbb{C})$, à laquelle on associe une porte également notée A .



La barre oblique « / » devant et après la porte A signifie que plusieurs lignes quantiques sont représentées en une seule. Ici l'entrée et la sortie sont des n -qubits. La porte cA (pour *controlled A*) est une porte ayant en entrée un 1-qubit supplémentaire, qui détermine si on applique ou non la porte A .



Si le premier qubit est $|0\rangle$, on ne change pas le n -qubit restant, si le premier qubit est $|1\rangle$, on applique la porte A au n -qubit restant.

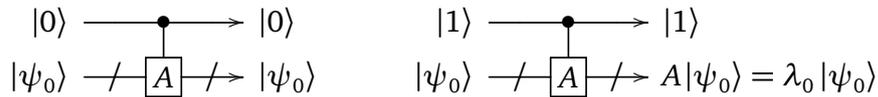


4.4. Porte cA et valeurs propres

Calculons l'action de la porte cA lorsque le n -qubit est le vecteur propre $|\psi_0\rangle$ associé à la valeur propre $\lambda_0 = e^{2i\pi\theta_0}$.

- Si l'entrée est $|0\rangle \otimes |\psi_0\rangle$ alors la sortie est $|0\rangle \otimes |\psi_0\rangle$.
- Si l'entrée est $|1\rangle \otimes |\psi_0\rangle$ alors la sortie est :

$$|1\rangle \otimes A|\psi_0\rangle = |1\rangle \otimes \lambda_0 |\psi_0\rangle = e^{2i\pi\theta_0} |1\rangle \otimes |\psi_0\rangle.$$



Si le 1-qubit de la première ligne est $\alpha |0\rangle + \beta |1\rangle$ alors calculons le $(n + 1)$ -qubit de sortie :

$$\begin{aligned} \alpha |0\rangle + \beta |1\rangle &\xrightarrow{cA} \alpha |0\rangle \otimes |\psi_0\rangle + \beta |1\rangle \otimes A|\psi_0\rangle \\ &= \alpha |0\rangle \otimes |\psi_0\rangle + \beta \lambda_0 |1\rangle \otimes |\psi_0\rangle \\ &= \alpha |0\rangle \otimes |\psi_0\rangle + e^{2i\pi\theta_0} \beta |1\rangle \otimes |\psi_0\rangle \end{aligned}$$

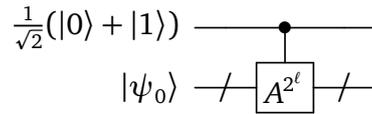
Ainsi le qubit de sortie s'écrit :

$$\boxed{(\alpha |0\rangle + e^{2i\pi\theta_0} \beta |1\rangle) \otimes |\psi_0\rangle}$$

Remarquons qu'après la factorisation par $|\psi_0\rangle$ le premier qubit de sortie est écrit $\alpha |0\rangle + e^{2i\pi\theta_0} \beta |1\rangle$. Cela peut sembler contradictoire avec le fait que la porte cA laisse inchangé le premier qubit qui devrait donc être $\alpha |0\rangle + \beta |1\rangle$, mais ici on inclut le coefficient provenant de la valeur propre ; mathématiquement on a simplement utilisé la bilinéarité $u \otimes (\lambda v) = (\lambda u) \otimes v$.

4.5. Bloc pour l'estimation de phase

La brique de base du circuit va être ce bloc :



La phase θ_0 est un réel qui vérifie $0 \leq \theta_0 < 1$. Supposons qu'il admette l'écriture binaire :

$$\theta_0 = 0.j_1.j_2 \dots j_n = \frac{j_1}{2} + \frac{j_2}{2^2} + \dots + \frac{j_n}{2^n}.$$

Lemme 3.

Le qubit de sortie du bloc précédent est :

$$\frac{1}{\sqrt{2}} (|0\rangle + e^{2i\pi 0.j_{\ell+1} \dots j_n} |1\rangle) \otimes |\psi_0\rangle.$$

Démonstration. Par le lemme 1 la valeur propre de la matrice A^{2^ℓ} associée au vecteur propre $|\psi_0\rangle$ est

$$\lambda_0^{2^\ell} = e^{2i\pi 2^\ell \theta_0}.$$

Or

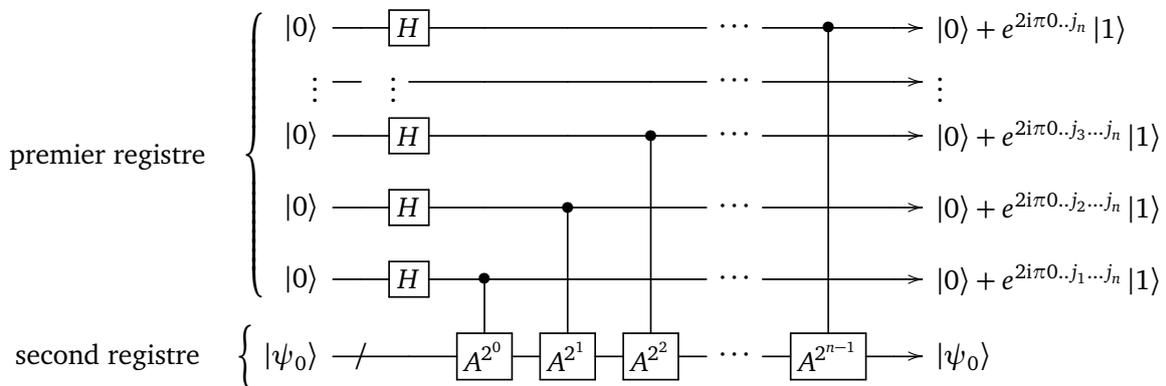
$$\begin{aligned} 2^\ell \theta_0 &= 2^\ell \left(\frac{j_1}{2} + \frac{j_2}{2^2} + \dots + \frac{j_n}{2^n} \right) \\ &= \underbrace{j_1 2^{\ell-1} + \dots + j_\ell}_{\text{partie entière}} + \underbrace{\frac{j_{\ell+1}}{2} + \dots + \frac{j_n}{2^{n-\ell}}}_{\text{partie décimale}} \\ &= k + 0.j_{\ell+1} \dots j_n \end{aligned}$$

Mais pour tout entier k , $e^{2i\pi k} = 1$ donc

$$\lambda_0^{2^\ell} = e^{2i\pi 2^\ell \theta_0} = e^{2i\pi 0.j_{\ell+1} \dots j_n}.$$

Maintenant, la porte cA^{2^ℓ} appliquée à $|0\rangle \otimes |\psi_0\rangle$ a pour sortie $|0\rangle \otimes |\psi_0\rangle$ et appliquée à $|1\rangle \otimes |\psi_0\rangle$ elle a pour sortie $\lambda_0^{2^\ell} |1\rangle \otimes |\psi_0\rangle$. Donc pour l'entrée du bloc $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |\psi_0\rangle$ la sortie est $\frac{1}{\sqrt{2}}(|0\rangle + e^{2i\pi 0.j_{\ell+1} \dots j_n} |1\rangle) \otimes |\psi_0\rangle$. □

4.6. Circuit d'estimation de phase



Le qubit de sortie de ce circuit est présenté sous la forme factorisée par le vecteur propre $|\psi_0\rangle$ (voir le lemme 3). Les coefficients $\frac{1}{\sqrt{2}}$ sont omis.

Proposition 1.

Le qubit de sortie du circuit d'estimation de phase est $|\phi\rangle \otimes |\psi_0\rangle$ où :

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

Ainsi le premier registre de sortie est égal à $\hat{F} |j_1.j_2\dots j_n\rangle$. Donc en composant le circuit à l'aide du circuit inverse de \hat{F} (c'est-à-dire le circuit de \hat{F}^{-1}), on obtient le qubit $|j_1.j_2\dots j_n\rangle$, donc les « décimales » j_1, j_2, \dots, j_n de l'écriture binaire de θ_0 .

Démonstration. Une porte de Hadamard H transforme l'entrée $|0\rangle$ des premières lignes en $\frac{1}{\sqrt{2}} (|0\rangle + |1\rangle)$. Ainsi le circuit est composé de blocs de type cA^{2^ℓ} comme étudiés précédemment. Chacune de ces portes transforme le qubit $\frac{1}{\sqrt{2}} (|0\rangle + |1\rangle) \otimes |\psi_0\rangle$ en $\frac{1}{\sqrt{2}} (|0\rangle + e^{2i\pi 0..j_{\ell+1}\dots j_n} |1\rangle) \otimes |\psi_0\rangle$. Ce qui conduit au résultat. Par le circuit de la section 3 qui réalise la transformée de Fourier, on vérifie immédiatement que la sortie P du premier registre est $\hat{F} |j_1.j_2\dots j_n\rangle$. Ainsi $|j_1.j_2\dots j_n\rangle = \hat{F}^{-1}(P)$. Nous savons réaliser un circuit quantique pour \hat{F} . Comment réaliser un circuit pour \hat{F}^{-1} ? Tout simplement en reprenant le circuit de \hat{F} et en le lisant de droite à gauche (au lieu de la lecture habituelle de gauche à droite). Cette opération est possible car toutes les portes quantiques sont inversibles : donc obtenir la porte A^{-1} , c'est lire l'action d'une porte A de droite à gauche.

Nous obtenons donc l'état quantique de base $|j_1.j_2\dots j_n\rangle$ dont la mesure donne les bits j_1, j_2, \dots, j_n qui permettent ainsi de retrouver la phase $\theta_0 = \frac{j_1}{2} + \frac{j_2}{4} + \dots$ et donc la valeur propre λ_0 . \square

QUATRIÈME PARTIE

| 1.1 >
| J · J >

VIVRE DANS UN MONDE QUANTIQUE

Cryptographie quantique

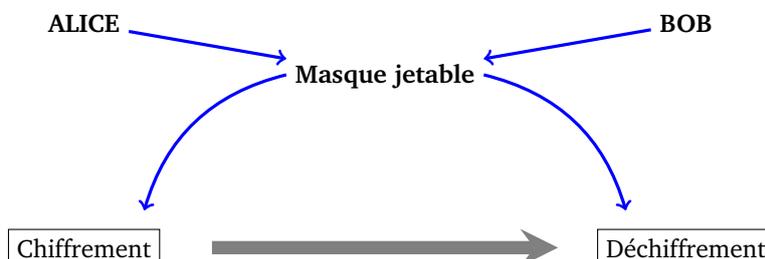
Vidéo ■ partie 16. Cryptographie quantique

Nous étudions le protocole BB84 qui permet le partage d'un secret commun entre deux personnes grâce à la physique quantique.

1. Le chiffrement parfait existe

Commençons par comprendre qu'un secret commun entre deux personnes permet une communication parfaitement sûre. C'est d'ailleurs ce protocole qui était utilisé par le « téléphone rouge » reliant les USA et l'URSS pendant la guerre froide.

1.1. Masque jetable



- Alice veut envoyer un message à Bob. Ce message est composé de 0 et de 1 (par exemple pour un nombre, on utiliserait son écriture binaire : 14 serait codé 1.1.1.0; pour une lettre on utiliserait le code ASCII : « A » serait codé 1.0.0.0.0.0.1).

Exemple : le message est $x = 1.0.1.1.0.1.1$.

- Alice et Bob s'étaient au préalable partagé un « masque jetable », qui est une suite secrète et aléatoire de 0 et de 1.

Exemple : le masque est $m = 0.0.1.0.1.1.1$.



- Alice envoie le message chiffré y obtenu par addition bit à bit (sans retenue) $y = x \oplus m$ (c'est un « ou exclusif » bit à bit).

$$\begin{array}{r}
 1.0.1.1.0.1.1 \\
 \oplus 0.0.1.0.1.1.1 \\
 \hline
 1.0.0.1.1.0.0
 \end{array}$$

Exemple : $y = x \oplus m = 1.0.0.1.1.0.0$.

- Bob déchiffre le message en ajoutant de nouveau le masque m à y : il obtient x . En effet $y \oplus m = x \oplus m \oplus m = x$ (car $0 \oplus 0 = 0$ et $1 \oplus 1 = 0$).

Exemple : $y \oplus m = 1.0.1.1.0.1.1 = x$.

$$\begin{array}{r} 1.0.0.1.1.0.0 \\ \oplus 0.0.1.0.1.1.1 \\ \hline 1.0.1.1.0.1.1 \end{array}$$

Voici les conditions que doit respecter le masque jetable m :

- il doit être un choix aléatoire,
- il doit rester secret,
- il doit être de la même longueur que le message,
- il ne doit servir qu'une seule fois.

1.2. Avantages et inconvénients

Avantages. Ce chiffrement est parfaitement sûr : un espion qui intercepterait le message chiffré y sans connaître le masque jetable m ne serait pas capable ici de décrypter le message. En effet, un 0 du message y peut correspondre aussi bien à 0 ou 1 du message original, de même pour un 1.

	$m = 0$	$m = 1$
$x = 0$	0	1
$x = 1$	1	0

Un espion n'a pas de meilleure méthode que de deviner au hasard si le message original contenait 0 ou 1. Si le message est de longueur n alors la probabilité qu'il décrypte le message complet est $\frac{1}{2^n}$ (ce qui revient à tirer au hasard un message parmi les 2^n messages possibles).

Inconvénients.

Tout d'abord il faut respecter scrupuleusement les consignes pour l'utilisation du masque jetable (choix aléatoire, usage unique,...). Une difficulté réside dans le fait qu'il faut que le masque reste un secret uniquement connu d'Alice et Bob : la méthode la plus simple est qu'Alice et Bob puissent se rencontrer physiquement pour déterminer ensemble le masque jetable. Pour le « téléphone rouge », les masques jetables étaient des listes de nombres transmis régulièrement via une valise diplomatique. Cet échange de masque est un problème pratique majeur puisqu'il nécessite une rencontre entre Alice et Bob. C'est pourquoi d'autres protocoles cryptographiques sont utilisés, comme par exemple RSA, pour permettre des communications chiffrées sans aucune rencontre physique, mais ils ne sont pas parfaitement sûrs.

2. BB84 : un secret commun

Nous présentons maintenant le protocole BB84 (dû à Bennett et Brassard en 1984) qui n'est pas vraiment un protocole cryptographique mais qui permet la création d'un secret commun sous la forme d'une suite de 0 et de 1. Cette suite peut ensuite, par exemple, être utilisée comme masque jetable pour un chiffrement parfait. Ce secret commun peut se construire à distance et on peut être sûr avec une forte probabilité que personne n'a intercepté le secret.

2.1. Deux bases

Alice souhaite envoyer à Bob une information 0 ou 1. Pour réaliser cela, elle va lui envoyer un qubit. Elle a le choix de deux codages différents.

Première base d'envoi « ⊕ ».

Dans cette base deux qubits sont possibles : $|\uparrow\rangle$ et $|\rightarrow\rangle$.

- $|\uparrow\rangle = |0\rangle$ représente l'information 0,
- $|\rightarrow\rangle = |1\rangle$ représente l'information 1.

Seconde base d'envoi « ⊗ ».

Dans cette base deux qubits sont possibles : $|\nearrow\rangle$ et $|\searrow\rangle$.

- $|\nearrow\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ représente l'information 0,
- $|\searrow\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$ représente l'information 1.

D'un point de vue physique ces qubits correspondent à des polarisations de photons : la polarisation à 90° pour la base « ⊕ » et la polarisation à 45° pour la base « ⊗ ». Selon le choix de base et selon l'information 0/1, Alice envoie un des quatre qubits $|\uparrow\rangle, |\rightarrow\rangle, |\nearrow\rangle, |\searrow\rangle$.

On retrouve aussi ces deux mêmes bases lors de la réception qui correspond à une mesure.

Première base de mesure « ⊕ ».

Voici l'information que Bob obtient lorsqu'il mesure le qubit reçu dans la base « ⊕ ».

qubit	$ \uparrow\rangle$	$ \rightarrow\rangle$	$ \nearrow\rangle$	$ \searrow\rangle$
information	0	1	0 ou 1 (50% chaque)	0 ou 1 (50% chaque)

Seconde base de mesure « ⊗ ».

Voici l'information que Bob obtient lorsqu'il mesure le qubit reçu dans la base « ⊗ ».

qubit	$ \uparrow\rangle$	$ \rightarrow\rangle$	$ \nearrow\rangle$	$ \searrow\rangle$
information	0 ou 1 (50% chaque)	0 ou 1 (50% chaque)	0	1

Conclusion. Si Bob effectue la mesure dans la même base que celle d'envoi alors il obtient exactement l'information 0 ou 1 envoyée par Alice. Par contre s'il mesure dans l'autre base que celle d'envoi, il obtient alors un bit 0 ou 1 aléatoire qui n'a rien à voir avec l'information envoyée par Alice.

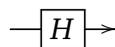
Circuits quantiques

Base « ⊕ ». Pour l'envoi il n'y a rien à faire, le qubit est $|0\rangle$ ou $|1\rangle$. Pour la réception, il s'agit juste d'une mesure classique :



On retrouve bien que, par exemple, $|\nearrow\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ se mesure en 0 ou 1 avec chacun une probabilité $\frac{1}{2}$.

Base « ⊗ ». Pour l'envoi, l'information 0 est codée par $H|0\rangle = |\nearrow\rangle$ et l'information 1 est codée par $H|1\rangle = |\searrow\rangle$. Donc une porte H de Hadamard suffit.



Pour la réception, le circuit est composé d'une porte H suivi d'une mesure :



Par exemple si le qubit reçu est $|\searrow\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$, alors la porte de Hadamard l'envoie sur $|1\rangle$ qui se mesure en 1. (On pourrait aussi utiliser que $H \circ H|\psi\rangle = |\psi\rangle$.)

2.2. Protocole

Voici le protocole de partage d'un secret commun.

1. Alice – envoi.

- Alice choisit des bits 0 ou 1 au hasard.
- Par chaque bit, elle choisit au hasard une base d'envoi \oplus ou \otimes .
- Pour chaque bit, elle a donc quatre situations et elle envoie le qubit correspondant :

bit/base	(0, \oplus)	(1, \oplus)	(0, \otimes)	(1, \otimes)
qubit	$ \uparrow\rangle$	$ \rightarrow\rangle$	$ \nearrow\rangle$	$ \searrow\rangle$

2. Bob – réception.

- Bob reçoit une liste de qubits.
- Pour chaque qubit il choisit au hasard une base de mesure \oplus ou \otimes .
- Bob mesure chaque qubit reçu parmi $|\uparrow\rangle$, $|\rightarrow\rangle$, $|\nearrow\rangle$, $|\searrow\rangle$ dans la base choisie \oplus ou \otimes .

3. Alice & Bob – mise en commun.

- Alice et Bob établissent la liste de leurs bases identiques (les deux ont choisi \oplus ou les deux ont choisi \otimes). Cette discussion peut être publique.
- Alice et Bob ne conservent que les rangs où les choix de base sont identiques. Les autres sont oubliés.
- Alice ne conserve que les bits correspondant à ces rangs.
- Pour chacun de ces rangs, Bob mesure dans la base (commune) et obtient le même bit qu'Alice.

2.3. Exemple

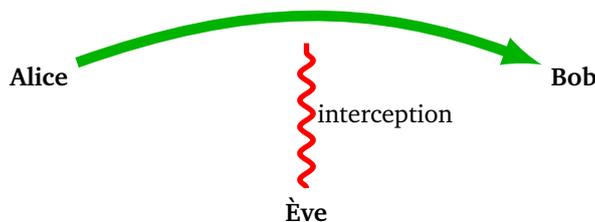
Voici un exemple. À vous de terminer de compléter ce tableau.

Alice bit	1	0	0	1	1	1	0	1	0
Alice base	\oplus	\otimes	\oplus	\otimes	\otimes	\oplus	\oplus	\otimes	\otimes
Qubit	$ \rightarrow\rangle$	$ \nearrow\rangle$	$ \uparrow\rangle$	$ \searrow\rangle$	$ \searrow\rangle$				
Bob base	\oplus	\oplus	\oplus	\otimes	\oplus	\otimes	\oplus	\otimes	\oplus
Base commune ?	oui	non	oui	oui	non				
Bit commun	1		0	1					

Le message commun est 1.0.1...

2.4. Sûreté

Pour l'instant Alice et Bob partagent un message commun. Mais celui-ci est-il secret ? Il faut s'assurer que le message n'a pas été intercepté ou modifié en cours de transmission.



La sécurité repose sur le théorème de non clonage quantique (voir le chapitre « Portes quantiques »). Ève ne peut pas lire un qubit puis le renvoyer à Bob. En effet, toute mesure modifie irrémédiablement le qubit.

Expliquons sur un exemple : Alice envoie l'information 0 dans la base \otimes , c'est-à-dire qu'elle transmet le qubit $|\nearrow\rangle$. Ève doit choisir une base pour sa lecture (car elle ne connaît pas la base d'envoi d'Alice).

- Si elle choisit la base \otimes , alors la mesure de $|\nearrow\rangle$ donne toujours 0, son interception est réussie ;

- si elle choisit la base \oplus , alors la mesure de $|\nearrow\rangle$ donne 0 (avec probabilité $\frac{1}{2}$, interception réussie) ou 1 (avec probabilité $\frac{1}{2}$, interception ratée).

Ève ne sait pas si elle a choisi la bonne base. Si elle a choisi la bonne base alors elle pourrait renvoyer le bon qubit à Bob. Mais si elle a choisi la mauvaise base elle va renvoyer $|\uparrow\rangle$ ou $|\rightarrow\rangle$ à Bob. Lorsque Bob va vérifier avec Alice qu'il a la bonne base, alors la lecture de $|\uparrow\rangle$ ou $|\rightarrow\rangle$ dans la base \otimes va donner 0 ou 1, et le bit sera faux dans la moitié des cas.

Bilan : Ève obtient la bonne information 0/1 dans les $\frac{3}{4}$ des cas (mais sans savoir quand c'est bon ou mauvais). Mais surtout si Ève intervient lors de la transmission, alors Bob obtient un mauvais bit d'information avec une probabilité $\frac{1}{4}$ (parmi les bits du message commun).

Voici donc la fin du protocole.

4. Alice & Bob – vérification de la sécurité.

- Alice et Bob se communiquent publiquement un échantillon de n bits du message commun (par exemple les n premiers bits).
- Si les échantillons ne sont pas exactement les mêmes alors un espion est intervenu, l'ensemble du message est compromis et il faut tout recommencer.
- Si les échantillons sont exactement identiques, alors la transmission est considérée comme sûre (d'autant plus sûre que n est grand). Le reste du message constitue alors le secret commun.

0.1.0.1.0.0.1.0 | 0.1.1.1.0.1.1.0.0.1.0.0.0.0.1.0
échantillon | secret commun

Détaillons les calculs de la sécurité de la transmission.

- Si aucun espion intervient, alors les échantillons d'Alice et Bob sont toujours identiques (probabilité 1, quelle que soit la taille n de l'échantillon).
- Si un espion intervient entre Alice et Bob alors, pour chaque bit, la probabilité qu'il parvienne correctement à Bob est de $\frac{3}{4}$. Donc les échantillons de n bits d'Alice et Bob sont complètement identiques avec probabilité $(\frac{3}{4})^n$. Si n est assez grand, alors cette probabilité est presque nulle. Ce qui signifie qu'on détecte presque sûrement la présence d'un espion.
- Voici des exemples :
 - $n = 10 : (\frac{3}{4})^{10} = 0.0563$, donc dans environ 95% des cas l'espion est repéré,
 - $n = 20 : (\frac{3}{4})^{20} = 0.003 \dots$ donc dans 99.7% des cas l'espion est repéré,
 - $n = 100 : (\frac{3}{4})^{100} \simeq 3 \cdot 10^{-13}$ donc l'espion est repéré sauf 1 fois sur 1 000 000 000 000.

Bilan.

- Alice et Bob partagent un secret commun,
- ils sont raisonnablement certains de ne pas avoir été espionnés,
- ce secret commun peut servir de masque jetable pour une communication chiffrée.

3. Alice et Bob divorcent : qui garde le chien ?

Alice et Bob ne se font plus confiance, et ils doivent décider par téléphone qui garde le chien. L'un pourrait tirer à pile ou face et annoncer le résultat à l'autre mais chacun pense que l'autre peut tricher. Comment faire ?

Nous allons voir la simulation d'un tirage à pile ou face à distance dans le monde quantique. Voici le protocole expliqué simplement : Alice et Bob tirent chacun de leur côté une pièce à pile ou face. S'ils obtiennent tous les deux « pile » ou tous les deux « face » c'est Bob qui gagne, sinon c'est Alice. Le point crucial est de se débrouiller pour qu'aucun des deux ne puisse mentir en annonçant son résultat.

3.1. Protocole

1. Alice choisit une base d'envoi \oplus ou \otimes .

- Alice décide au hasard d'une base d'envoi \oplus ou \otimes (c'est son tirage à pile ou face).
- Elle envoie une série aléatoire de bits, par exemple 0.0.1.0.1.1.
- Elle envoie les qubits correspondant dans la base qu'elle a choisie. Par exemple :
 - si elle a choisi la base \oplus : $|\uparrow\rangle, |\uparrow\rangle, |\rightarrow\rangle, |\uparrow\rangle, |\rightarrow\rangle, |\rightarrow\rangle$,
 - si elle a choisi la base \otimes : $|\nearrow\rangle, |\nearrow\rangle, |\searrow\rangle, |\nearrow\rangle, |\searrow\rangle, |\searrow\rangle$.

2. Bob choisit une base de mesure \oplus ou \otimes .

- Bob décide au hasard d'une base de mesure \oplus ou \otimes (c'est son tirage à pile ou face).
- Il effectue la mesure des qubits reçus dans la base qu'il a choisie.
- Il obtient une suite de mesures 0 ou 1.

3. Bob annonce la base qu'il a choisie pour la mesure.

4. Alice dévoile la base qu'elle avait choisie pour l'envoi ainsi que les bits transmis.

5. Gagnant : si les deux bases coïncident Bob a gagné, sinon c'est Alice.

6. Vérification : Bob vérifie qu'Alice n'a pas menti. Bob a annoncé son choix avant Alice il doit donc vérifier qu'Alice n'a pas triché, pour cela il compare sa mesure avec les bits d'Alice :

- s'il a trouvé la bonne base, alors sa mesure est exactement la même que les bits d'Alice,
- s'il n'a pas trouvé la bonne base, alors il doit avoir en moyenne la moitié des bits corrects et la moitié des bits faux.

Il sait donc s'il a trouvé la bonne base ou pas. Plus de détails sur la vérification sont donnés ci-dessous.

3.2. Vérifications

Tout d'abord Bob ne peut pas tricher, d'une part les mesures qu'il effectue ne permettent pas de déduire quelle base d'envoi Alice avait choisie et d'autre part Bob annonce en premier sa base à Alice.

Voyons comment Bob vérifie le résultat annoncé par Alice.

Imaginons qu'Alice ait choisi la base \oplus et les bits 0.0.1.0.1.1 elle transmet donc les qubits $|\uparrow\rangle, |\uparrow\rangle, |\rightarrow\rangle, |\uparrow\rangle, |\rightarrow\rangle, |\rightarrow\rangle$.

Si Bob a choisi de mesurer les qubits dans la même base \oplus alors il va obtenir après mesure la même suite de bits 0.0.1.0.1.1. Donc dans le cas où il gagne les bits d'Alice et de Bob sont identiques.

Si Bob a choisi l'autre base, ici \otimes , alors la mesure de $|\uparrow\rangle, |\rightarrow\rangle$, conduit à 0 ou 1 aléatoirement. Il va donc, en moyenne, avoir la moitié de bits faux et l'autre moitié corrects. La probabilité que les n bits de Bob coïncident exactement avec les n bits d'Alice est $\frac{1}{2^n}$ et est donc très faible (si n est assez grand).

Bilan : Bob sait s'il a choisi la même base qu'Alice juste en comparant les bits mesurés avec les bits annoncés par Alice.

Par contre, Alice pourrait essayer de tricher : si Bob choisit la base \oplus , elle pourrait mentir pour faire perdre Bob et dire « J'avais choisi la base \otimes » ou inversement. Mais dans ce cas, elle va être démasquée car elle a déjà envoyé les qubits qui ont déjà été mesurés par Bob et ne peut donc plus rien modifier. Or, comme on l'a déjà vu, la mesure des qubits $|\uparrow\rangle, |\uparrow\rangle, |\rightarrow\rangle, |\uparrow\rangle, |\rightarrow\rangle, |\rightarrow\rangle$ dans la base \otimes a très peu de chance de donner exactement 0.0.1.0.1.1.

La cryptographie quantique n'en est encore qu'à ces débuts, c'est tout un domaine à découvrir !

Vidéo ■ partie 17. Code correcteur

Lors de la transmission d'un qubit il peut y avoir des erreurs. Les codes correcteurs permettent de détecter et corriger ces erreurs.

1. Un code correcteur classique

Lorsqu'on transmet un message électronique, le message reçu peut être différent du message envoyé à cause d'erreurs (erreurs de lecture/écriture, interférences,...). Cela peut être sans conséquence, par exemple tout le monde comprend la phrase « UN PETIT PAS POUR L'HOMME » malgré les fautes de frappe, mais pour envoyer un code d'identification du style « 562951413 » une erreur sur un seul chiffre compromet le message.

On distingue deux tâches : détecter s'il y a eu une erreur (si c'est le cas on pourrait envoyer à nouveau le message), mais on peut aussi utiliser des techniques qui permettent de corriger directement certaines erreurs.

1.1. Répétition

L'idée la plus simple pour sécuriser la transmission est de répéter chaque partie du message. Dans toute cette section on considère que le message est composé de 0 et de 1 :

- chaque « 0 » est remplacé avant transmission par « 000 »,
- chaque « 1 » est remplacé par « 111 ».

S'il y a une erreur lors de la transmission, le décodage se fait selon le principe de la majorité :

- 000, 001, 010, 100 sont décodés en « 0 »,
- 111, 110, 101, 011 sont décodés en « 1 ».

Prenons l'exemple du message « 1.0.1 » :

- répétition de chaque bit : « 111.000.111 »,
- le message est transmis mais des erreurs surviennent,
- le message reçu est « 101.001.111 »,
- selon la règle de la majorité, le message décodé est bien le message original « 1.0.1 ».

Bien évidemment, s'il y a trop d'erreurs, par exemple « 000 » est altéré en « 101 », alors le message décodé est erroné.

1.2. Efficacité

Nous allons comparer les erreurs suivant le codage utilisé. Considérons un message de n bits, chaque bit transmis pouvant être altéré avec une probabilité p .

Proposition 1.

- Sans utiliser de codage, le message transmis est entièrement correct avec probabilité $(1 - p)^n$.
- En utilisant le codage de répétition triple, le message décodé est entièrement correct avec probabilité $(1 - p_3)^n$ où $p_3 = p^2(3 - 2p)$.

Les tableaux suivants présentent les probabilités qu'un message de longueur n soit transmis parfaitement correctement, selon différentes valeurs de p , avec ou sans répétition.

Cas $p = 0.1$ (10% des bits sont altérés)			Cas $p = 0.01$ (1% des bits sont altérés)			Cas $p = 0.001$ (1 bit sur mille est altéré)		
n	sans répétition	avec répétition	n	sans répétition	avec répétition	n	sans répétition	avec répétition
10	35%	75%	10	90%	99.7%	10	99%	99.99%
100	0%	5%	100	36%	97%	100	90%	99.97%
1000	0%	0%	1000	0%	75%	1000	37%	99.7%

Conclusion : pour un message long, il est indispensable de mettre en place un système permettant de détecter puis de corriger les erreurs.

Démonstration de la proposition 1.

- Sans utiliser de codage, un bit est transmis correctement avec probabilité $1 - p$, pour que le message reçu soit identique au message initial, il faut que les n bits soient transmis sans être altérés, ce qui arrive avec probabilité $(1 - p)^n$.
- Pour le codage de répétition triple, prenons l'exemple de la transmission du bit « 0 ». Le message reçu est :
 - « 000 » avec probabilité $(1 - p)^3$ (trois bits corrects),
 - « 001 », « 010 », « 100 », chacun avec probabilité $p(1 - p)^2$ (un bit faux, deux bits corrects),
 - « 110 », « 101 », « 011 », chacun avec probabilité $p^2(1 - p)$ (deux bits faux, un bit correct),
 - « 111 » avec probabilité p^3 (trois bits faux).

Pour les deux premiers cas, la règle de la majorité conduit au bon décodage « 0 ». Pour les deux derniers cas, le décodage donne « 1 » et le bit est mal décodé.

La probabilité d'erreur (deux derniers cas) est donc :

$$p_3 = 3p^2(1 - p) + p^3 = p^2(3 - 2p).$$

Chaque bit est donc transmis de façon correcte avec une probabilité $1 - p_3$; les n bits d'une suite sont tous transmis correctement avec probabilité $(1 - p_3)^n$.

□

Exercice.

Faire les calculs de la proposition 1 dans le cas d'une répétition de longueur 5 : $0 \mapsto 00000$ et $1 \mapsto 11111$.

2. Correction d'erreurs en informatique quantique

2.1. Obstacles

Les ordinateurs quantiques sont encore balbutiants et commettent beaucoup d'erreurs. Les codes correcteurs sont donc importants mais se confrontent à des problèmes spécifiques à l'informatique quantique :

- on ne peut pas mesurer un qubit sans le perturber irrémédiablement (effondrement du paquet d'onde),
- on ne peut pas cloner un qubit (voir le théorème de non-clonage quantique du chapitre « Portes quantiques »),
- enfin un qubit $\alpha|0\rangle + \beta|1\rangle$ peut prendre une infinité de valeurs (α, β étant des nombres complexes quelconques) à la différence du cas classique dans lequel l'information est codée par seulement deux valeurs 0 et 1.

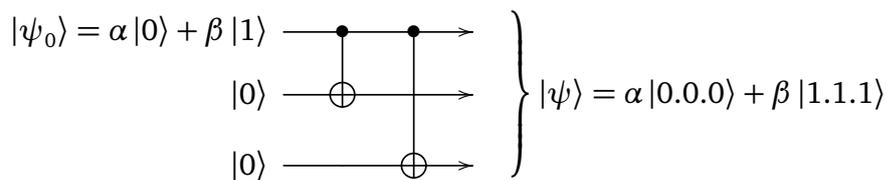
Et pourtant, malgré toutes ces difficultés, il est possible de corriger des erreurs !

Dans toute la suite, on suppose que l'on souhaite transmettre un message formé par un qubit $|\psi_0\rangle = \alpha|0\rangle + \beta|1\rangle$. On commence par expliquer deux idées importantes pour la suite.

2.2. Augmentation d'un qubit

Nous avons vu que le fait de répéter un bit permet de corriger certaines erreurs. Comment faire pour nos qubits ? Nous allons généraliser une porte *FANOUT* (voir le chapitre « Portes quantiques ») à l'aide de deux portes *CNOT*.

Le circuit suivant transforme le 1-qubit $|\psi_0\rangle = \alpha|0\rangle + \beta|1\rangle$ en le 3-qubit $|\psi\rangle = \alpha|0.0.0\rangle + \beta|1.1.1\rangle$.

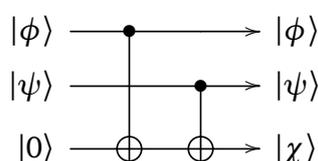


- Pour le circuit ci-dessus si $|\psi_0\rangle = |0\rangle$ alors $|\psi\rangle = |0.0.0\rangle$ et si $|\psi_0\rangle = |1\rangle$ alors $|\psi\rangle = |1.1.1\rangle$. Par linéarité, cela donne le résultat $|\psi\rangle$ attendu pour $|\psi_0\rangle = \alpha|0\rangle + \beta|1\rangle$.
- Noter qu'ici nous n'avons pas dupliqué les coefficients. Le théorème de non-clonage quantique montre qu'aucun circuit ne permettrait de réaliser le 3-qubit $(\alpha|0\rangle + \beta|1\rangle)(\alpha|0\rangle + \beta|1\rangle)(\alpha|0\rangle + \beta|1\rangle)$.
- Noter aussi que si on considère le circuit inverse (de la droite vers la gauche) alors on effectue la transformation inverse : on passe de $|\psi\rangle = \alpha|0.0.0\rangle + \beta|1.1.1\rangle$ à $(\alpha|0\rangle + \beta|1\rangle)|0\rangle|0\rangle$. Ainsi après une mesure des deux derniers qubits, on obtiendrait sur la première ligne notre qubit $|\psi_0\rangle = \alpha|0\rangle + \beta|1\rangle$.

2.3. Décider si deux qubits de base sont égaux (sans les mesurer)

On considère ici un message composé de 1-qubits de base de la forme $|0\rangle$ ou $|1\rangle$. On souhaite vérifier sans aucune mesure si deux qubits de base sont égaux.

Pour cela on utilise un circuit à trois lignes quantiques : les deux premières sont les entrées à comparer, la troisième ligne est une ligne auxiliaire dont la sortie va répondre à la question « Les deux qubits de base sont-ils égaux ? »



Vérifier que :

- Si $|\phi\rangle = |0\rangle$ et $|\psi\rangle = |0\rangle$ alors $|\chi\rangle = |0\rangle$.
- Si $|\phi\rangle = |1\rangle$ et $|\psi\rangle = |1\rangle$ alors $|\chi\rangle = |0\rangle$.
- Si $|\phi\rangle = |0\rangle$ et $|\psi\rangle = |1\rangle$ alors $|\chi\rangle = |1\rangle$.
- Si $|\phi\rangle = |1\rangle$ et $|\psi\rangle = |0\rangle$ alors $|\chi\rangle = |1\rangle$.

Noter que sur les deux premières lignes les qubits $|\phi\rangle$ et $|\psi\rangle$ restent inchangés. La sortie $|\chi\rangle$ vaut $|0\rangle$ si et seulement si $|\phi\rangle$ et $|\psi\rangle$ sont les mêmes qubits de base. La sortie $|\chi\rangle$ vaut $|1\rangle$ si et seulement si $|\phi\rangle$ et $|\psi\rangle$ sont des qubits de base différents.

Remarque : ce circuit permet de tester l'égalité de deux qubits de base, mais ne permet pas de comparer deux 1-qubits quelconques.

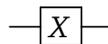
3. Code correcteur pour le flip d'un qubit

3.1. Un circuit qui corrige les erreurs ?

On souhaite transmettre le qubit $|\psi_0\rangle = \alpha|0\rangle + \beta|1\rangle$. On commence par augmenter le qubit en $|\psi\rangle = \alpha|0.0.0\rangle + \beta|1.1.1\rangle$ (on suppose que cette opération se fait sans erreur).

Lors de la transmission de ce 3-qubit il peut y avoir des erreurs. Commençons par le cas où l'erreur est un « flip » d'un des qubits. Par exemple $\alpha|0.0.0\rangle + \beta|1.1.1\rangle$ est mal transmis en $\alpha|0.0.1\rangle + \beta|1.1.0\rangle$.

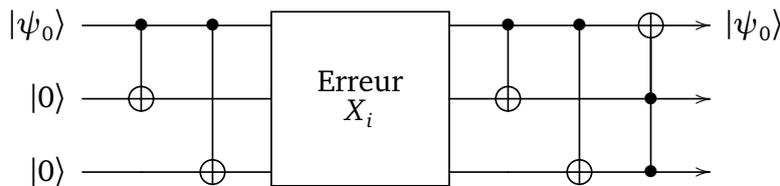
Noter qu'un flip correspond à une porte X sur l'un des trois qubits, porte qui change $|0\rangle$ en $|1\rangle$ et réciproquement.



Comment détecter et corriger cette erreur ?

3.2. Circuit

Voici un circuit qui permet la transmission correct d'un qubit $|\psi_0\rangle$, même si lors de la transmission une erreur de type X se produit.



- X_i désigne l'action d'une porte X sur l'une des lignes $i \in \{1, 2, 3\}$.
- Où que soit cette erreur, la première ligne du circuit renvoie toujours le qubit original $|\psi_0\rangle$.
- Le circuit est composé de 4 portes $CNOT$ et terminé par une porte de Toffoli.
- On rappelle qu'une porte de Toffoli, est l'action d'une porte X (sur la ligne du « \oplus ») à condition que les qubits des deux autres lignes soient tous les deux $|1\rangle$.

3.3. Calculs

Effectuons les calculs qui justifient que le qubit de sortie est bien le qubit original malgré l'erreur.

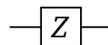
- Cas erreur X_1 (flip sur la première ligne).
 - Cas $|\psi_0\rangle = |0\rangle$. Le 3-qubit avant l'erreur est $|0.0.0\rangle$. Alors le 3-qubit reçu est $|1.0.0\rangle$, les deux portes $CNOT$ le transforment en $|1.1.1\rangle$ et la porte de Toffoli renvoie $|0.1.1\rangle$. Le premier qubit est bien $|0\rangle$.
 - Cas $|\psi_0\rangle = |1\rangle$. Le 3-qubit avant l'erreur est $|1.1.1\rangle$, mais le qubit reçu est $|0.1.1\rangle$, les deux portes $CNOT$ ne le changent pas, et la porte de Toffoli renvoie $|1.1.1\rangle$. Le premier qubit est bien $|1\rangle$.

- Cas $|\psi_0\rangle = \alpha|0\rangle + \beta|1\rangle$. Par linéarité, les calculs précédents donnent comme sortie $\alpha|0.1.1\rangle + \beta|1.1.1\rangle = (\alpha|0\rangle + \beta|1\rangle)|1.1\rangle$. Si on ne retient que le premier qubit on obtient $\alpha|0\rangle + \beta|1\rangle$ qui est bien notre qubit initial $|\psi_0\rangle$.
- Cas erreur X_2 (flip sur la deuxième ligne).
 - Cas $|\psi_0\rangle = |0\rangle$. Avant l'erreur le 3-qubit est $|0.0.0\rangle$, après erreur c'est $|0.1.0\rangle$, les deux portes *CNOT* et la porte de Toffoli ne changent rien. On obtient $|0.1.0\rangle$.
 - Cas $|\psi_0\rangle = |1\rangle$. Avant l'erreur le 3-qubit est $|1.1.1\rangle$, après erreur c'est $|1.0.1\rangle$, les deux portes *CNOT* donnent $|1.1.0\rangle$, la porte de Toffoli ne change rien. On obtient $|1.1.0\rangle$.
 - Cas $|\psi_0\rangle = \alpha|0\rangle + \beta|1\rangle$. Par linéarité, la sortie vaut $\alpha|0.1.0\rangle + \beta|1.1.0\rangle = (\alpha|0\rangle + \beta|1\rangle)|1.0\rangle$. Le premier qubit est encore $\alpha|0\rangle + \beta|1\rangle$ qui est bien le qubit initial $|\psi_0\rangle$.
- Cas erreur X_3 (flip sur la troisième ligne).
 Les calculs sont similaires. $|\psi_0\rangle = \alpha|0\rangle + \beta|1\rangle$ donne après erreur $\alpha|0.0.1\rangle + \beta|1.1.0\rangle$, et la fin du circuit renvoie $\alpha|0.0.1\rangle + \beta|1.0.1\rangle = (\alpha|0\rangle + \beta|1\rangle)|0.1\rangle$. Le premier qubit est de nouveau $|\psi_0\rangle$.

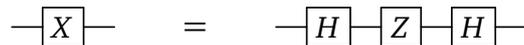
4. Code correcteur pour l'inversion de phase d'un qubit

4.1. Circuit

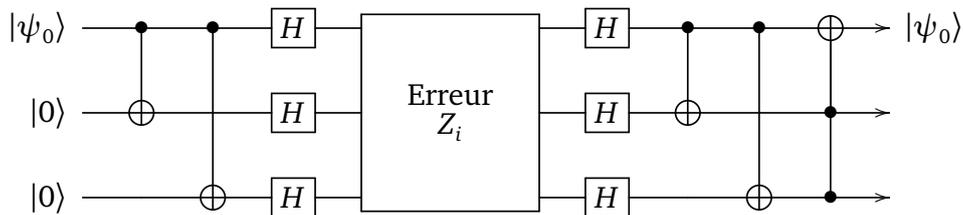
Le flip n'est pas la seule erreur possible. Une autre erreur est l'inversion de phase qui est le changement de $\alpha|0\rangle + \beta|1\rangle$ en $\alpha|0\rangle - \beta|1\rangle$. Le changement de phase correspond à une porte *Z*.



On souhaite transmettre le qubit $|\psi_0\rangle = \alpha|0\rangle + \beta|1\rangle$. On commence par augmenter le qubit en $|\psi\rangle = \alpha|0.0.0\rangle + \beta|1.1.1\rangle$. Ensuite, lors de la transmission, on suppose que se produit un changement de phase sur l'un des trois qubits. On se ramène à la situation précédente en notant qu'une porte *X* est équivalente à une porte *HZH*, où *H* est une porte de Hadamard :



Voici le circuit qui détecte et corrige cette erreur.



Z_i désigne l'action d'une porte *Z* sur l'une des lignes $i \in \{1, 2, 3\}$.

4.2. Calculs

Nous n'avons pas à faire les calculs puisque ce sont les mêmes que pour le flip. En effet, on a rappelé que $HZH = X$, donc l'ensemble des portes de Hadamard et l'erreur Z_i correspondent à une erreur X_i .

Une autre façon de voir les calculs est d'utiliser la notation $|+\rangle$ et $|-\rangle$.

$$|+\rangle = H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \qquad |-\rangle = H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$$

On a aussi réciproquement $H|+\rangle = |0\rangle$ et $H|-\rangle = |1\rangle$.

Un changement de phase Z envoie $\alpha|0\rangle + \beta|1\rangle$ sur $\alpha|0\rangle - \beta|1\rangle$, et peut être simplement défini par :

$$|+\rangle \xrightarrow{Z} |-\rangle \quad |-\rangle \xrightarrow{Z} |+\rangle$$

C'est donc une sorte de flip dans une autre base.

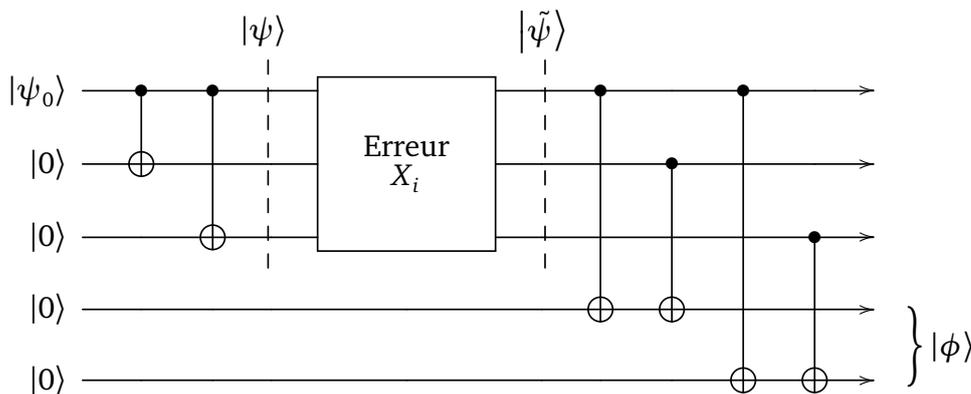
Si au départ $|\psi_0\rangle = |0\rangle$ alors, après augmentation, on a $|\psi\rangle = |0.0.0\rangle$. Puis à l'aide des portes de Hadamard le 3-qubit avant erreur est $|+\rangle|+\rangle|+\rangle$, que l'on note $|+.+.+\rangle$. L'erreur Z_i change l'un des signes, par exemple on obtient $|-.+.+\rangle$, les nouvelles portes de Hadamard le transforment en $|1.0.0\rangle$ (qui correspond bien à un flip classique de $|\psi\rangle$), qui est corrigé par la fin du circuit en $|0.1.1\rangle$ et ainsi le premier qubit est bien $|0\rangle$.

5. Détection d'un flip

Les circuits précédents font très bien leur travail : ils détectent et corrigent les erreurs. Mais ils ne sont pas très pédagogiques car les deux tâches sont effectuées en même temps. Nous allons modifier légèrement ces circuits afin qu'ils détectent les erreurs et on expliquera ensuite comment les corriger.

5.1. Un circuit qui détecte les flips

Voici un circuit qui détecte un flip.



Ce circuit se décompose en deux registres :

Premier registre. Les trois premières lignes correspondent au qubit augmenté $|000\rangle$ ou $|111\rangle$, lors de la transmission survient une erreur qui est ici un flip sur une des trois lignes.

Second registre. Les deux dernières lignes servent à détecter l'erreur. On parle de « lignes auxiliaires ».

5.2. Sortie

Notons $|\tilde{\psi}\rangle$ le 3-qubit du premier registre après transmission (juste après l'erreur éventuelle). Notons $|\phi\rangle$ le 2-qubit obtenu en sortie du second registre (à la fin du circuit).

- **Pas d'erreur.**

Si $|\psi_0\rangle = |0\rangle$ et si $|\tilde{\psi}\rangle = |0.0.0\rangle$ alors $|\phi\rangle = |0.0\rangle$. Il n'y a pas d'erreur donc rien à corriger. De même, si $|\psi_0\rangle = |1\rangle$ et $|\tilde{\psi}\rangle = |1.1.1\rangle$ alors de nouveau $|\phi\rangle = |0.0\rangle$. Il n'y a toujours pas d'erreur donc rien à corriger. Par linéarité, si $|\psi_0\rangle = \alpha|0\rangle + \beta|1\rangle$ alors le 5-qubit final est $(\alpha|0.0.0\rangle + \beta|1.1.1\rangle)|0.0\rangle$.

- **Flip du premier qubit.**

Si $|\tilde{\psi}\rangle = |1.0.0\rangle$ (alors qu'on voulait transmettre $|0.0.0\rangle$) alors $|\phi\rangle = |1.1\rangle$. Il y a une erreur et cette erreur est sur la première ligne. On corrige l'erreur en rajoutant une porte X sur la première ligne. Ainsi après correction on obtient bien un premier registre qui vaut $|0.0.0\rangle$.

De même si $|\tilde{\psi}\rangle = |0.1.1\rangle$ (alors qu'on voulait transmettre $|1.1.1\rangle$) alors de nouveau $|\phi\rangle = |1.1\rangle$. Et on rajoute une porte X sur la première ligne.

Ainsi un qubit $|\psi\rangle = \alpha|0.0.0\rangle + \beta|1.1.1\rangle$ qui serait mal transmis en $|\tilde{\psi}\rangle = \alpha|1.0.0\rangle + \beta|0.1.1\rangle$, donnerait $|\phi\rangle = |1.1\rangle$ et serait bien corrigé en $|\psi\rangle$.

• **Flip du deuxième qubit.**

Si $|\tilde{\psi}\rangle = \alpha|0.1.0\rangle + \beta|1.0.1\rangle$ (au lieu de $\alpha|0.0.0\rangle + \beta|1.1.1\rangle$) alors $|\phi\rangle = |1.0\rangle$ et on rajoute une porte X sur la deuxième ligne.

• **Flip du troisième qubit.** Si $|\tilde{\psi}\rangle = \alpha|0.0.1\rangle + \beta|1.1.0\rangle$ alors $|\phi\rangle = |0.1\rangle$. On corrige l'erreur en rajoutant une porte X sur la troisième.

Noter qu'on n'a jamais effectué de mesure sur le premier registre.

Bilan :

- si $|\phi\rangle = |0.0\rangle$ pas d'erreur,
- si $|\phi\rangle = |1.1\rangle$ erreur de flip sur la première ligne,
- si $|\phi\rangle = |1.0\rangle$ erreur de flip sur la deuxième ligne,
- si $|\phi\rangle = |0.1\rangle$ erreur de flip sur la troisième ligne.

Une fois qu'on sait sur quelle ligne est l'erreur par mesure de $|\Phi\rangle$, il est facile de la corriger en ajoutant une porte X en fin de circuit sur la ligne correspondante.

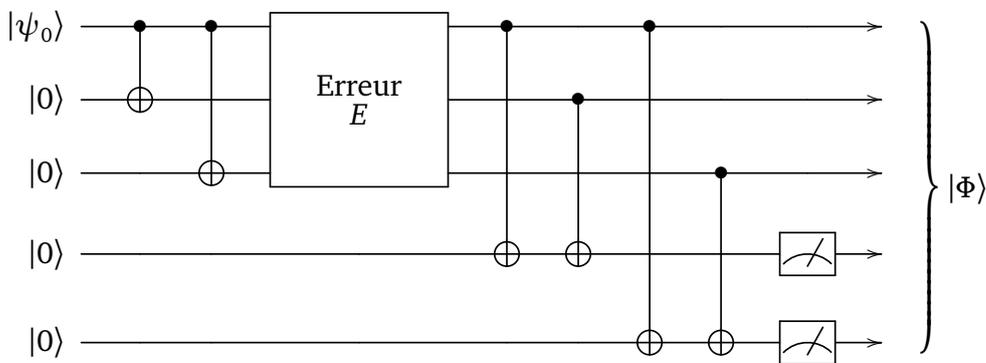
5.3. Erreur par déformation

On souhaite envoyer l'information $|0\rangle$, augmentée en $|0.0.0\rangle$. On suppose que l'erreur est d'un type nouveau, le qubit est légèrement déformé. Prenons l'exemple d'une erreur qui, en sortie du premier registre, fournit :

$$|\tilde{\psi}\rangle = \sqrt{1-\epsilon^2}|0.0.0\rangle + \epsilon|0.0.1\rangle,$$

où $\epsilon > 0$ est un petit réel.

Le circuit suivant détecte ce type d'erreur. C'est le même que le circuit précédent avec en plus la mesure du second registre.



On a vu que juste avant la mesure, si $|\tilde{\psi}\rangle = |0.0.0\rangle$ alors le 5-qubit de sortie est $|\Phi\rangle = |0.0.0\rangle|0.0\rangle$; et si $|\tilde{\psi}\rangle = |0.0.1\rangle$ alors le 5-qubit de sortie est $|\Phi\rangle = |0.0.1\rangle|0.1\rangle$.

Donc pour le qubit $|\tilde{\psi}\rangle = \sqrt{1-\epsilon^2}|0.0.0\rangle + \epsilon|0.0.1\rangle$, la sortie est (avant mesure) :

$$|\Phi\rangle = \sqrt{1-\epsilon^2}|0.0.0\rangle|0.0\rangle + \epsilon|0.0.1\rangle|0.1\rangle.$$

Que se passe-t-il lorsque l'on mesure le second registre ? Deux mesures seulement sont possibles 0.0 ou bien 0.1.

- Si on obtient la mesure 0.0, alors on sait qu'il n'y a rien à corriger. Effectivement, dans ce cas le premier registre s'est effondré en $|\Psi\rangle = |0.0.0\rangle$, il n'y a pas d'erreur.
- Si on obtient la mesure 0.1, alors on sait qu'il y a une erreur qu'il faut corriger en ajoutant un flip sur la troisième ligne. Effectivement dans ce cas le premier registre s'est effondré en $|\Psi\rangle = |0.0.1\rangle$. Après correction on obtient $|0.0.0\rangle$.

Dans tous les cas on obtient, après correction éventuelle, le 3-qubit $|0.0.0\rangle$. Il s'est passé un phénomène appelé *discrétisation de l'erreur par la mesure* : même si l'erreur pouvait prendre une infinité de formes (car il y a une infinité de ϵ possibles), après mesure on se ramène à seulement deux possibilités.

Proposition 2.

Le circuit précédent détecte n'importe quelle erreur du type $E = aI + bX$.

I désigne l'identité et X un flip. Une erreur $E = aI + bX$ transforme un qubit $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ en

$$E|\psi\rangle = (aI + bX)|\psi\rangle = a|\psi\rangle + bX|\psi\rangle.$$

Si lors de la transmission, sur une seule des trois premières lignes, un qubit subit une telle erreur, alors la sortie du second permet de savoir comment corriger cette erreur.

La preuve est la généralisation des calculs faits pour l'exemple avec les « ϵ ».

Démonstration. On note $|\psi_0\rangle = \alpha|0\rangle + \beta|1\rangle$ et $|\psi\rangle = \alpha|0.0.0\rangle + \beta|1.1.1\rangle$. Si $E = I$, alors il n'y pas d'erreur le qubit de sortie avant mesure est $|\Phi_0\rangle = |\psi\rangle|0.0\rangle$.

Si $E = X$, alors on a déjà vu que selon la ligne de l'erreur, le qubit de sortie avant mesure est l'un des $|\tilde{\psi}_1\rangle|1.1\rangle, |\tilde{\psi}_2\rangle|1.0\rangle, |\tilde{\psi}_3\rangle|0.1\rangle$.

Si l'erreur est $aI + bX$ alors, par linéarité le qubit de sortie avant mesure est par exemple

$$|\Phi\rangle = a|\psi\rangle|0.0\rangle + b|\tilde{\psi}_1\rangle|1.1\rangle$$

(ou l'une des deux autres situations).

Lors de la mesure du second registre :

- Si on obtient 0.0, alors on sait qu'il n'y a rien à corriger. Effectivement dans ce cas le premier registre s'est effondré en $|\psi\rangle$, il n'y a pas d'erreur.
- Si on obtient 1.1, alors on sait qu'il y a une erreur qu'il faut corriger en ajoutant un flip sur la première ligne. Effectivement dans ce cas le premier registre s'est effondré en $|\tilde{\psi}_1\rangle = \alpha|1.0.0\rangle + \beta|0.1.1\rangle$. Après correction on obtient $|\psi\rangle = \alpha|0.0.0\rangle + \beta|1.1.1\rangle$.
- De même pour les deux autres situations.

□

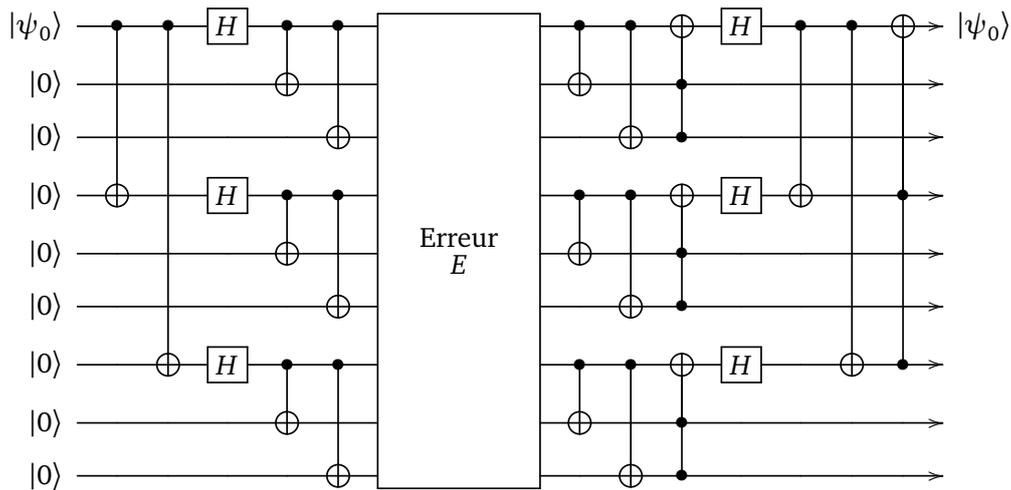
Exercice.

Réaliser un circuit qui détecte une inversion de phase et expliquer ensuite comment corriger l'éventuelle erreur. Montrer que votre circuit détecte n'importe quelle erreur $aI + bZ$ sur une ligne.

6. Code correcteur de Shor

6.1. Circuit

Nous terminons avec un circuit composé de 9 lignes. Ce circuit détecte et corrige une erreur de transmission qui se produirait sur une seule des 9 lignes. Cette erreur peut être un flip X , une inversion de phase Z , mais plus généralement n'importe quelle erreur sur un 1-qubit (mais toujours sur une seule ligne).



6.2. Calculs

Proposition 3.

Le circuit précédent détecte et corrige n'importe quelle erreur du type $E = aI + bX + cY + dZ$ qui arriverait sur une seule de ses lignes.

Avant de justifier ce résultat, il faut passer un peu de temps à comprendre que ce circuit est construit en regroupant le circuit qui corrige un flip et celui qui corrige une inversion de phase. Ensuite le mieux est de le programmer pour vérifier qu'il fonctionne !

Donnons maintenant des explications théoriques. Nous modélisons une erreur E comme la transformation linéaire d'un qubit en un autre qubit. Autrement dit une erreur est définie par une matrice 2×2 , notée E . Rappelons la définition des matrices de Pauli auxquelles on ajoute l'identité :

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Ces quatre matrices sont linéairement indépendantes et forment donc une base de l'espace vectoriel de dimension 4, $M_2(\mathbb{C})$. Ainsi n'importe quelle $E \in M_2(\mathbb{C})$ se décompose :

$$E = aI + bX + cY + dZ$$

où $a, b, c, d \in \mathbb{C}$.

Par linéarité du circuit, on se ramène aux quatre cas $E = I, E = X, E = Y, E = Z$. La structure du circuit, qui regroupe la correction de flip et d'inversion de phase, fait qu'il corrige les erreurs X et Z (et aussi I).

Il ne reste plus qu'à traiter le cas de $E = Y$. Mais nous avons l'égalité :

$$Y = iXZ.$$

Ceci est une égalité de matrices, qui se traduit en une équivalence de portes :

$$\boxed{Y} = \boxed{Z} \boxed{X} \boxed{\times i}$$

Ainsi une erreur Y est la combinaison d'un flip et d'une inversion de phase et sera bien corrigée par notre circuit.

Notes. Ce cours n'est qu'un aperçu d'un vaste domaine. La présentation adoptée ici est basée sur un cours en ligne du CERN *Introduction to quantum computing* par Elias F. Combarro. Une étude plus approfondie est faite dans le livre de Nielsen et Chuang *Quantum computation and quantum information*.

Avantage quantique

Vidéo ■ partie 18. Avantage quantique

Quand est-ce qu'un ordinateur quantique sera plus performant qu'un ordinateur classique ?

1. L'avantage quantique

1.1. Une définition ?

L'**avantage quantique** c'est deux choses : un ordinateur quantique et un problème à résoudre. Cet ordinateur quantique sait résoudre ce problème alors qu'aucun ordinateur classique ne peut le faire en temps raisonnable.

- La notion d'avantage quantique est un peu floue : par exemple, est-ce que le problème à résoudre doit être utile ou pas ? Ce que signifie résoudre un problème est clair : l'ordinateur quantique renvoie la bonne réponse en un temps raisonnable (disons en quelques heures ou quelques jours), mais il n'est pas clair de prouver qu'aucun algorithme ne peut résoudre ce même problème sur un ordinateur classique (peut-être qu'un bon algorithme n'a pas encore été trouvé).
- Certaines compagnies affirment avoir déjà dépassé le cap. Le consensus étant que cet avantage sera atteint durant la décennie 2020.
- Le terme « avantage quantique » est maintenant préféré à « suprématie quantique ». Outre l'aspect moins vindicatif du terme « avantage », à moyenne échéance, il est probable que les ordinateurs classiques et les ordinateurs quantiques cohabiteront ; on peut en effet imaginer que les ordinateurs classiques délégueront certaines tâches complexes aux ordinateurs quantiques.

D'autres définitions sont à inventer pour mesurer l'efficacité d'un ordinateur quantique et comparer les technologies mises en œuvre en tenant compte du nombre de qubits, des connexions entre ces qubits, du nombre de portes implémentées, du taux d'erreurs...

1.2. Factorisation

La factorisation des grands entiers est une bonne illustration. Rappelons qu'étant donné un entier il s'agit de lui trouver deux facteurs tels que $n = p \times q$. Tout d'abord c'est un problème utile, car la sécurité de nombreuses communications repose sur ce problème.

Ordinateurs classiques. Les meilleurs algorithmes actuels sur des ordinateurs classiques permettent de factoriser des entiers jusqu'à 250 chiffres (800 bits) (voir le chapitre « Arithmétique »). Les calculs se font sur des centaines d'ordinateurs en parallèle et prennent plusieurs semaines. La complexité de ces algorithmes de factorisation augmente de manière exponentielle avec le nombre de bits. La recommandation minimale pour la longueur d'une clé RSA sûre est actuellement de 2048 bits (600 chiffres). Une telle factorisation est hors de portée de tous les ordinateurs et algorithmes actuels pour encore plusieurs années.

Ordinateurs quantiques. L'algorithme de Shor démontre théoriquement l'avantage des ordinateurs quantiques car il permet de factoriser rapidement des grands entiers.

En 2020 les ordinateurs quantiques possèdent jusqu'à 50 qubits et savent factoriser des entiers à 5 chiffres (14 bits). Pour factoriser un entier de 2048 bits en quelques heures, il faudrait une machine quantique à 20 millions de qubits, ce qui ne sera pas atteint avant une ou deux décennies !

2. Simulation d'un ordinateur quantique

Les ordinateurs quantiques sont fondamentalement différents des ordinateurs classiques, cependant certains circuits quantiques simples peuvent être réalisés de façon efficace sur un ordinateur classique.

Théorème 1 (Gottesman – Knill).

N'importe quel circuit quantique, composé uniquement de portes de Hadamard H , de portes $CNOT$, de portes de Pauli X , Y , Z et de portes de phase S , initialisé avec des états $|0\rangle$ et terminé par des mesures, peut être simulé efficacement par un ordinateur classique.

- « Efficacement » signifie en temps polynomial par rapport à la donnée du circuit.
- La porte S , appelé « porte phase » ou « porte $\frac{\pi}{4}$ », est définie par la matrice :

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

- En fait les portes de Pauli X , Y , Z peuvent être générées uniquement avec des portes H et S (c'est un bon exercice).
- Cependant les portes contenues dans l'énoncé ne permettent pas de générer toutes les portes quantiques : cet ensemble de portes n'est donc pas universel. Par exemple la porte de Toffoli, la porte \sqrt{CNOT} ou la porte $\frac{\pi}{8}$ ne peuvent pas être générées à partir des portes du théorème.
- L'ordinateur classique doit être capable de simuler le hasard. Par exemple la mesure du qubit $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ donne 0 ou 1 avec probabilité $\frac{1}{2}$ ce qui revient à jouer à pile ou face.

Voici des exemples de circuits que nous avons rencontrés et qui peuvent être simulés efficacement sur un ordinateur classique :

- la communication par codage super-dense (voir le chapitre « Découverte de l'informatique quantique »),
- la téléportation quantique (voir le chapitre « Téléportation quantique »),
- les codes correcteurs d'erreur (voir la section « Détection d'un flip » du chapitre « Code correcteur »).

3. Arbre de calculs

La simulation d'un ordinateur quantique par un ordinateur classique se confronte non seulement à des problèmes de temps de calculs mais aussi à des problèmes de mémoire. En effet, dès que l'on dépasse 50 qubits, il y a 2^{50} états de base, soit plus de 10^{15} états à stocker.

Nous allons voir une modélisation du calcul des états d'un circuit sous la forme d'un arbre. En parcourant l'arbre branche par branche, on teste toutes les possibilités sans utiliser trop de mémoire à chaque fois. La méthode n'apporte pas un gain de temps, qui reste exponentiel en fonction du nombre n de qubits, mais la taille de la mémoire utilisée est linéaire en n .

Nous expliquons cette modélisation par des exemples. Nous partons d'un circuit, avec un état initial. Il s'agit d'obtenir tous les états possibles que l'on pourrait obtenir après mesure.

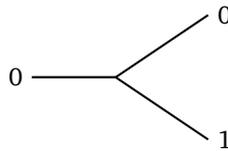
La brique fondamentale à comprendre est l'arbre pour une porte H de Hadamard.

Exemple.

Soit le circuit quantique initialisé par $|0\rangle$ suivi d'une simple porte de Hadamard (sans mesure sur la figure de gauche, avec mesure à droite) :

$$|0\rangle \xrightarrow{H} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \qquad |0\rangle \xrightarrow{H} \begin{array}{|c} \hline \diagup \quad \diagdown \\ \hline \end{array} \rightarrow \begin{array}{l} 0 \\ \text{ou} \\ 1 \end{array}$$

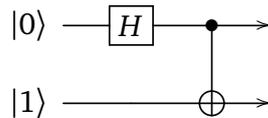
La sortie est $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$. Si on termine par une mesure alors la sortie est 0 ou 1. Voici l'arbre de calculs.



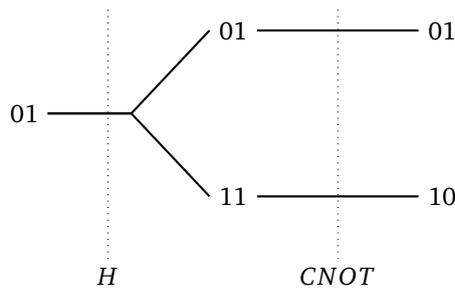
L'arbre de calculs représente tout simplement les deux possibilités. Lorsque l'on passe une porte H les feuilles de l'arbre 0 et 1 représentent la superposition des états $|0\rangle$ et $|1\rangle$. Autrement dit les feuilles représentent toutes les mesures possibles. On a simplifié l'écriture en omettant les coefficients $\frac{1}{\sqrt{2}}$.

Voici un exemple avec deux lignes quantiques.

Exemple.



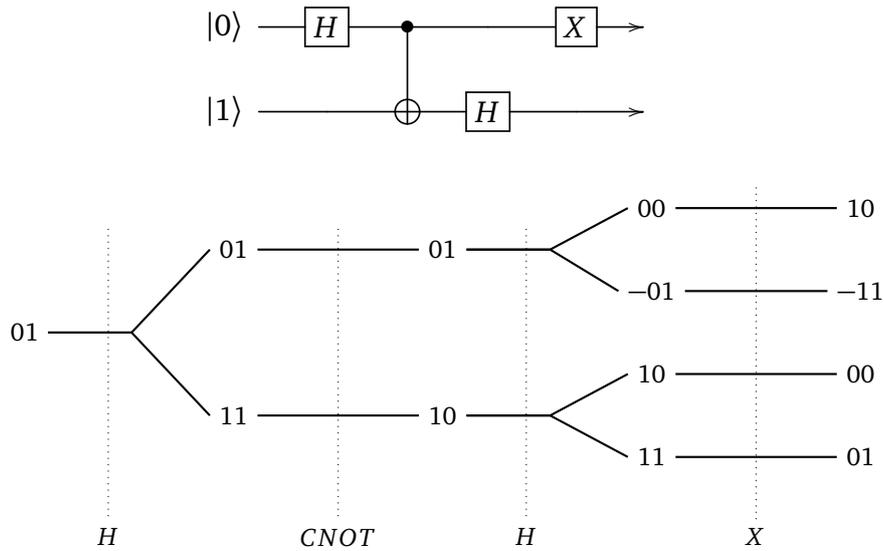
La sortie est le qubit $|\psi\rangle = \frac{1}{\sqrt{2}}(|0.1\rangle + |1.0\rangle)$. Une mesure donnerait donc 0.1 ou bien 1.0, ce que l'on retrouve aux feuilles de notre arbre.



Noter qu'au passage de la porte $CNOT$ l'arbre ne se ramifie pas (chaque branche se poursuit en une seule branche).

Continuons avec un exemple pour comprendre le fonctionnement : à chaque porte H correspond une bifurcation en deux branches. L'intérêt de cet arbre est que les calculs d'une branche sont mis en commun tant qu'il n'y a pas de bifurcation.

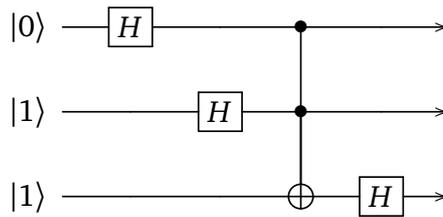
Exemple.



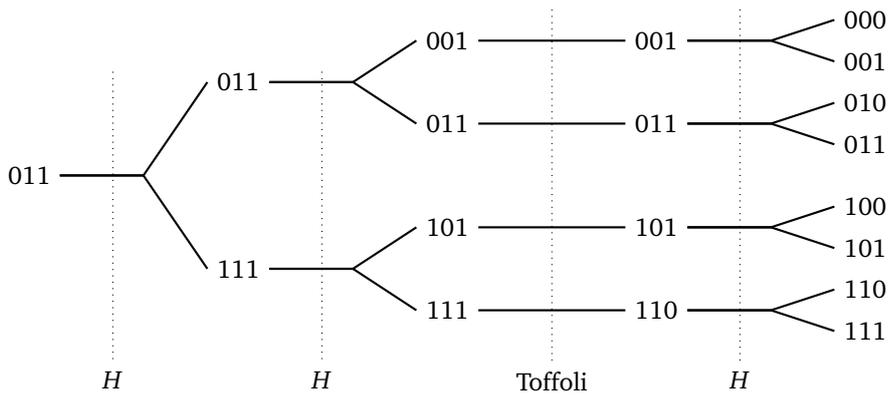
Le qubit de sortie est $|\psi\rangle = \frac{1}{2}(|0.0\rangle + |0.1\rangle + |1.0\rangle - |1.1\rangle)$

Exemple.

Terminons avec un circuit contenant une porte de Toffoli.



Pour simplifier l'arbre ci-dessous, on omet les signes et les coefficients devant les qubits.



Notes. La référence pour les arbres de calculs est Andrew Shi, [Recursive path-summing simulation of quantum computation](#) (2017).

Notes et bibliographie

Retrouvez ce cours en vidéos :

[Chaîne « Quantum » sur Youtube](#)

Vous pouvez construire facilement des circuits quantiques en ligne :

[Quirk : Quantum Circuit Simulator](#)

Vous pouvez aussi installer *Qiskit* pour *Python* :

[Qiskit](#)

Le livre de référence est *Quantum computation and quantum information* (Cambridge university press, 2010) de Mickael Nielsen et Isaac Chuang. Cet ouvrage est à la fois abordable et complet.

La première partie de ce cours est inspirée d'une série de vidéos de Mickael Nielsen :

[« Quantum computing for the determined »](#)

Un des rares cours en français est « Introduction à l'informatique quantique » par Y. Leroyer et G. Sénizergues à l'Enseirb-Matmeca [disponible ici](#).

Le chapitre « Algorithme de Shor » est basé sur l'article *Shor's algorithm for factoring large integers* par C. Lavor, L.R.U. Manssur, R. Portugal (2003), disponible [sur arXiv](#).

Remerciements

Je remercie Michel Bodin, Stéphanie Bodin, François Recher et Jean-Michel Torres pour leurs relectures.

Vous pouvez récupérer l'intégralité des codes *Python* ainsi que tous les fichiers sources sur la page *GitHub* d'Exo7 : « [GitHub : Quantum](#) ».



Ce livre est diffusé sous la licence *Creative Commons – BY-NC-SA – 4.0 FR*.
Sur le site Exo7 vous pouvez télécharger gratuitement le livre en couleur.

- algorithme d'Euclide, 144
- algorithme de Deutsch–Jozsa, 96, 118
- algorithme de Grover, 125
- algorithme de Shor, 171, 180, 218
- algorithme probabiliste, 126
- avantage quantique, 217

- base orthonormale, 113
- binaire, 66
- bit, 66
- bra, 49, 60, 113

- chat de Schrödinger, 80
- chiffrement
 - BB84, 203
 - parfait, 202
 - RSA, 150, 217
- circuit quantique, 7, 21, 218
- codage super-dense, 16, 31
- code correcteur, 208
- complexité, 72
- congruence, 147
- constante de Planck, 75
- cryptographie, 202

- division euclidienne, 143
- dualité onde/corpuscule, 76

- équation de Schrödinger, 81
- état de Bell, 11
- état quantique, 2
- exponentiation rapide, 149

- factorisation d'un entier, 217
- fonction d'onde, 80
- fraction continue, 169

- grand O, 71
- groupe
 - ordre, 174
 - produit, 179
 - racine carrée, 176
 - théorème des restes chinois, 179
 - $\mathbb{Z}/n\mathbb{Z}$, 148

- hachage, 127

- indiatrice d'Euler, 148
- intrication quantique, 11, 18, 55, 83

- ket, 3, 49, 60, 112

- lemme de Gauss, 145

- matrice
 - adjointe, 59, 112
 - carrée, 56
 - déterminant, 58
 - identité, 58
 - inverse, 58
 - produit, 57
 - spéciale unitaire, 64
 - transposée, 59
 - unitaire, 61, 113

- mesure, 5

- modulo, 147

- nombre complexe, 33
 - argument, 37
 - conjugué, 35
 - exponentielle, 38
 - formule de Moivre, 38
 - module, 34
 - partie imaginaire, 34
 - partie réelle, 34

- nombre premier, 145
- norme, 48, 50

- oracle, 97, 107, 128

pgcd, 144
 porte, 6, 114
 CCNOT, 106
 CNOT, 10, 13, 104, 218
 contrôlée, 136, 194, 198
 de Hadamard, 6
 de Pauli, 8, 45, 218
 de Toffoli, 15, 106
 FANOUT, 105, 210
 H, 6
 logique, 67
 NOT, 6
 phase, 218
 $\frac{\pi}{4}$, 218
 $\frac{\pi}{8}$, 115
 \sqrt{CNOT} , 8
 R_k , 194
 S, 194, 218
 SWAP, 104, 196
 T, 115, 194
 universelle, 69, 107
 X, 6, 8, 45, 211, 218
 Y, 8, 45, 216, 218
 Z, 8, 45, 136, 212, 218
 principe d'incertitude d'Heisenberg, 75
 produit scalaire, 49, 60, 112
 produit tensoriel, 53

 qiskit, 21
 qubit
 calculs, 12, 36
 définition, 3, 35
 2-qubit, 9, 36
 équivalence, 39
 états de Bell, 91
 mesure, 83
 mesure partielle, 89, 93
 norme, 13, 36
 notation entière, 120
 n-qubit, 14
 produit, 54
 qiskit, 24, 26
 réalisation, 82

 sphère de Bloch, 42, 52
 superposition, 3, 9, 82
 suprématie quantique, 217

 téléportation quantique, 85
 théorème d'Euler, 149
 théorème de Bézout, 144
 (petit) théorème de Fermat, 145
 théorème de Gottesman–Knill, 218
 théorème de non-clonage quantique, 115
 transformation de Grover, 133
 transformation de Hadamard, 121
 transformée de Fourier, 184

 valeur propre, 196
 vecteur dual, 48
 vecteur propre, 196