Notes de cours
Semaine 1

Cours Turing+

1 Introduction

Voici le problème que nous allons étudier tout au long de ce module : soient m, n deux nombres entiers positifs ; on appelle *fonction booléenne* une fonction $f : \{0,1\}^n \to \{0,1\}^m$. De cette fonction, on aimerait apprendre certaines propriétés, par exemple :

- existe-t-il un point $(x_1, x_2, \ldots, x_n) \in \{0,1\}^n$ tel que $f(x_1, x_2, \ldots, x_n) = (0, 0, \ldots, 0)$?

- en quel point $(x_1, x_2, \ldots, x_n) \in \{0,1\}^n$ la fonction $f(x_1, x_2, \ldots, x_n)$ est-elle minimale (i.e., contient-elle le nombre maximum de zéros) ?

- la fonction f possède-t-elle une période (à définir précisément encore) ? et si oui, laquelle ?

- etc.

Pour répondre à ces questions, nous supposerons toujours disposer d'une méthode rapide pour évaluer la fonction f en un point donné $(x_1, x_2, \ldots, x_n)$ (en informatique, on parle aussi d'*oracle*). Il est bien sûr toujours possible d'évaluer la fonction f en tous les points possibles de $\{0,1\}^n$ pour obtenir des réponses à nos questions, mais si n est grand, ce nombre d'évaluations vaut 2^n, qui est un nombre absurdement élevé (pour rappel, si $n = 1'000$, alors $2^n \simeq 10^{300}$, donc un 1 suivi de 300 zéros...) ; cette stratégie est clairement infaisable en pratique.

Notre but sera donc de répondre aux différentes questions posées en utilisant un nombre minimal d'évaluations de la fonction f. Et nous étudierons chaque question en utilisant des circuits classiques d'une part, et des circuits quantiques d'autre part. Pour les différentes questions étudiées, nous aurons ainsi l'occasion de constater la supériorité des circuits quantiques.

2 Circuits classiques

Les "briques de base" qui composent un circuit classique manipulant des bits (0 et 1) sont les portes AND (ET), OR (OU), NOT (NON) et COPY (COPIE). Pour rappel, voici leur

description :

- la porte AND prend deux bits x_1, x_2 en entrée et sa sortie vaut 1 si et seulement si $x_1 = 1$ et $x_2 = 1$.

- la porte OR prend deux bits x_1, x_2 en entrée et sa sortie vaut 1 si et seulement si $x_1 = 1$ ou $x_2 = 1$ (non-exclusif).

- la porte NOT prend un bit x en entrée et sa sortie vaut 1 si et seulement si $x = 0$.

- la porte COPY prend un bit x en entrée et sa sortie est composée des deux bits de même valeur x, x.

Note : En pratique, cette dernière "porte" n'est souvent pas considérée comme une porte en tant que telle, car elle peut être implémentée dans un circuit électrique en connectant simplement les deux fils de sortie au fil d'entrée.

Pourquoi donc parler de "briques de base" en parlant de ces quatre portes ? A cause du théorème suivant :

Théorème (Emil Post, 1921). Toute fonction booléenne $f : \{0,1\}^n \to \{0,1\}^m$ peut être représentée par un circuit composé exclusivement de portes AND, OR, NOT et COPY.

Nous ne donnerons pas ici une démonstration de ce théorème, mais l'illustrerons plutôt sur un exemple, dans le cas où $n = 3$, $m = 1$ et la fonction boolénne f est donnée par le tableau suivant :

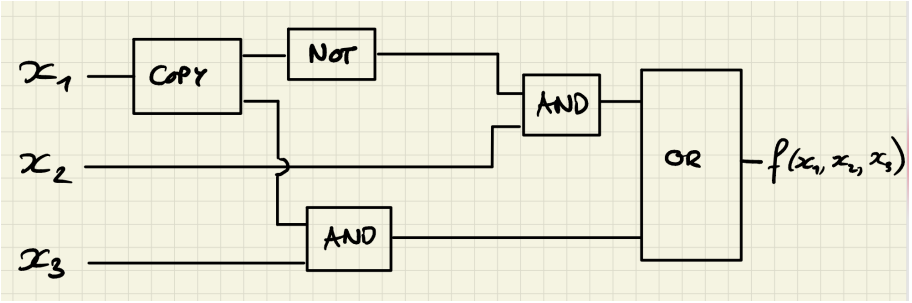
x_1	x_2	x_3	$f(x_1, x_2, x_3)$
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

Vous pouvez vérifier que l'expression suivante pour f mène bien aux sorties indiquées dans le tableau ci-dessus :

$$f(x_1, x_2, x_3) = ((\text{NOT } x_1) \text{ AND } x_2) \text{ OR } (x_1 \text{ AND } x_3)$$

Par exemple, $f(0, 1, 0) = (1 \text{ AND } 1) \text{ OR } (0 \text{ AND } 0) = 1 \text{ OR } 0 = 1$.

Au final, cela donne le circuit de la page suivante (à noter qu'une porte COPY est encore nécessaire pour pouvoir faire un double usage du bit d'entrée x_1).



Note : Il existe une méthode systématique (utilisée dans la démonstration du théorème de Post) pour construire un circuit correspondant à une fonction f donnée, mais la complexité du circuit résultant peut se révéler gigantesque (i.e., avec un nombre de portes exponentiel en n).

3 Portes réversibles

Ainsi, l'ensemble des quatre portes AND, OR, NOT et COPY est dit "universel". Pour autant, trois de ces portes ont un défaut notable si on pense à les utiliser dans un circuit quantique : elles ne sont pas *réversibles* : deux bits se transforment un seul bit après le passage d'une porte AND ou OR, et un bit se dédouble au passage d'une porte COPY.

Pour des raisons qui deviendront claires plus tard, les circuits quantiques ne peuvent pas admettre de portes irréversibles. Ceci dit, il existe un autre ensemble de portes réversibles permettant d'émuler le comportement de chacune de ces quatre portes. Voici ces nouvelles portes.

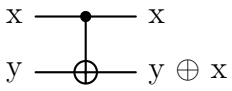
La porte NOT

Cette porte est clairement réversible, vu qu'un bit passant deux fois à travers celle-ci retrouve sa valeur de départ ; elle sera donc utilisable dans un circuit quantique.

Notation : Si x est le bit d'entrée, alors on note la sortie NOT $x = \overline{x}$.

La porte Controlled-NOT (CNOT)

Cette porte ("NON contrôlé" en français) prend deux bits en entrée et en sortie, et est symbolisée par



où $y \oplus x$ désigne l'opération XOR (ou exclusif) définie par $y \oplus x = 1$ si et seulement si $x = 1$ ou $y = 1$, mais pas simultanément.

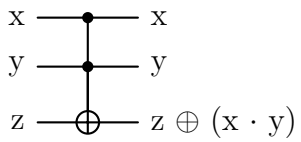
L'explication du nom de cette porte vient du fait que lorsque $x = 0$, le second bit y reste inchangé en sortie, tandis que si $x = 1$, alors la sortie du second bit vaut $y \oplus 1 = \overline{y} = \text{NOT } y$. Ainsi, la valeur d'entrée du bit x contrôle si l'opération NOT est effectuée sur le bit y ou pas.

Il est également clair que la porte CNOT est réversible, car de la même façon que pour la porte NOT, si on applique deux fois de suite cette porte aux deux bits x et y, ceux-ci retrouvent leur valeur initiale, car x reste de toutes façons inchangé et $(y \oplus x) \oplus x = y \oplus (x \oplus x) = y$.

A noter finalement que cette porte permet d'émuler la porte COPY vue précédemment : il suffit en effet de fixer $y = 0$ en entrée pour trouver deux fois le bit x en sortie.

La porte de Toffoli, ou Controlled-Controlled-NOT (CCNOT)

La porte de Toffoli ou CCNOT est quant à elle une généralisation de la porte CNOT qui prend trois bits en entrée et en sortie. Elle est symbolisée par



où $x \cdot y$ est la multiplication des deux bits x et y, qui vaut 1 si et seulement si $x = 1$ et $y = 1$, ce qui correspond à x AND y.

Pour les mêmes raisons que celles vues précédemment, la porte CCNOT est réversible. Elle permet aussi d'émuler naturellement la porte AND (en fixant $z = 0$ en entrée et en laissant tomber les deux premières sorties x et y).

Exercice : Il est également possible d'émuler une porte OR à partir des portes NOT, CNOT et CCNOT ; voyez-vous comment faire ?

Ainsi, l'ensemble des portes NOT, CNOT et CCNOT est également universel, car par les observations précédentes et le théorème de Post, il est possible de représenter toute fonction booléenne par un circuit composé uniquement de portes NOT, CNOT et CCNOT.

Question : En fait, seule une de ces trois portes suffit : voyez-vous laquelle ?

4 Bits quantiques ("qubits") et notation de Dirac

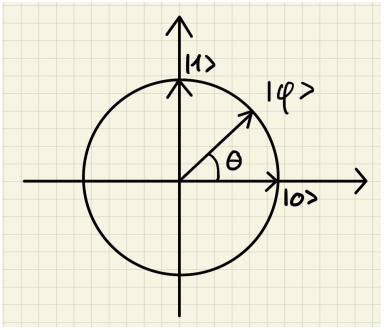
Un qubit

Classiquement, un bit peut prendre deux valeurs, 0 ou 1. En physique quantique, l'état d'un bit quantique ou "qubit" est caractérisé par un *vecteur unité* dans $\mathbb{R}^2$, le vecteur horizontal $\begin{pmatrix}1\\0\end{pmatrix}$ étant associé à la valeur 0 et le vecteur vertical $\begin{pmatrix}0\\1\end{pmatrix}$ étant associé à la valeur 1.

Tout l'intérêt du calcul quantique réside dans le fait qu'un qubit peut se trouver dans un état $\begin{pmatrix}\alpha_0\\\alpha_1\end{pmatrix}$, avec $\alpha_0^2 + \alpha_1^2 = 1$, qui est une *superposition* des états 0 et 1 : ainsi, le qubit est en quelque sorte *à la fois* dans l'état 0 et dans l'état 1, ce qui permet de paralléliser les calculs de manière efficace.

Note : En vrai, l'état d'un bit quantique est un vecteur unité dans $\mathbb{C}^2$, et les nombres α_0 et α_1 peuvent donc être des nombres complexes, mais nous n'aurons pas besoin de nombres complexes dans ce cours.

Voici comment se représenter l'état d'un qubit visuellement :



Introduisons ici la notation de Dirac utilisée en physique quantique : les états sont représentés en général par des "kets" $|\varphi\rangle \in \mathbb{R}^2$. On note

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad \text{et} \quad |\varphi\rangle = \begin{pmatrix} \alpha_0 \\ \alpha_1 \end{pmatrix}$$

A noter que $|\varphi\rangle = \alpha_0 |0\rangle + \alpha_1 |1\rangle$ et que ce vecteur est toujours sur le cercle unité, vu la condition $\alpha_0^2 + \alpha_1^2 = 1$. De ce fait, il existe aussi un angle $0 \leq \theta \leq 2\pi$ (l'angle que fait le vecteur $|\varphi\rangle$ avec l'horizontale) tel que $\cos(\theta) = \alpha_0$ et $\sin(\theta) = \alpha_1$.

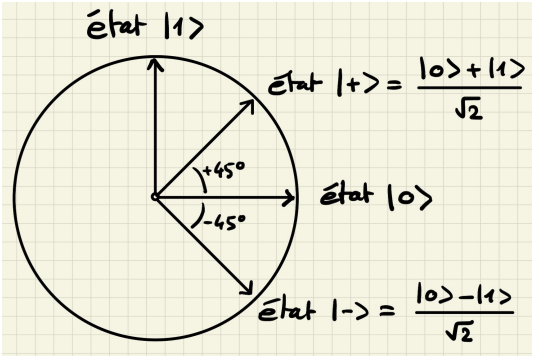
Parmi tous les états supersposés, citons-en deux qui joueront un rôle important dans ce qui va suivre : les états $|+\rangle$ et $|-\rangle$ définis (respectivement) par

$$|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = \cos(+45°)\,|0\rangle + \sin(+45°)\,|1\rangle$$

et

$$|-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = \cos(-45°)\,|0\rangle + \sin(-45°)\,|1\rangle$$

illustrés sur la figure suivante :



Quelques remarques s'imposent encore :

- Les vecteurs $|0\rangle = \begin{pmatrix}1\\0\end{pmatrix}$ et $|1\rangle = \begin{pmatrix}0\\1\end{pmatrix}$ forment une *base* de l'espace vectoriel $\mathbb{R}^2$ (appelée traditionnellement la *base computationnelle*).

- Le *produit scalaire* (usuel) entre deux vecteurs $|\varphi\rangle = \alpha_0 |0\rangle + \alpha_1 |1\rangle$ et $|\psi\rangle = \beta_0 |0\rangle + \beta_1 |1\rangle$ dans $\mathbb{R}^2$ est défini ainsi:

$$\langle\varphi|\psi\rangle = \alpha_0\beta_0 + \alpha_1\beta_1$$

et appelé un "braket" (le vecteur ligne $\langle\varphi| = (\alpha_0, \alpha_1)$ transposé du vecteur colonne $|\varphi\rangle = \begin{pmatrix}\alpha_0\\\alpha_1\end{pmatrix}$ étant lui-même appelé un "bra").

- La *norme* du vecteur $|\varphi\rangle = \alpha_0 |0\rangle + \alpha_1 |1\rangle$ est quant à elle donnée par

$$\| |\varphi\rangle \| = \sqrt{\langle\varphi|\varphi\rangle} = \sqrt{\alpha_0^2 + \alpha_1^2}$$

et vaut donc 1 si φ est un vecteur unité (représentant l'état d'un qubit).

- Avec le produit scalaire défini plus haut, la base $\{|0\rangle, |1\rangle\}$ est ainsi une base *orthonormée* de $\mathbb{R}^2$, car $\langle 0|0\rangle = \langle 1|1\rangle = 1$ et $\langle 0|1\rangle = \langle 1|0\rangle = 0$.

Exercice : Vérifiez que $\{|+\rangle, |-\rangle\}$ est également une base orthonormée de $\mathbb{R}^2$ (avec le même produit scalaire).

Deux qubits

L'état (joint) de deux qubits est décrit lui aussi par un vecteur unité, mais cette fois-ci dans $\mathbb{R}^4 = \mathbb{R}^2 \otimes \mathbb{R}^2$. La notation $\otimes$ désigne ici le *produit tensoriel*, mais il n'y a pas besoin pour l'instant de décrire formellement ce produit pour comprendre ce qui va suivre.

Dans $\mathbb{R}^4$, on définit les quatre *états de base* suivants (orthonormés selon le produit scalaire usuel dans $\mathbb{R}^4$) :

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

et tout état $|\varphi\rangle$ dans $\mathbb{R}^4$ est ainsi donné par une superposition de ceux-ci :

$$|\varphi\rangle = \begin{pmatrix}\alpha_{0,0}\\\alpha_{0,1}\\\alpha_{1,0}\\\alpha_{1,1}\end{pmatrix} = \alpha_{0,0} |0,0\rangle + \alpha_{0,1} |0,1\rangle + \alpha_{1,0} |1,0\rangle + \alpha_{1,1} |1,1\rangle$$

où $\alpha_{0,0}^2 + \alpha_{0,1}^2 + \alpha_{1,0}^2 + \alpha_{1,1}^2 = 1$. Remarquez que même si l'on parle ici de *deux* qubits, l'état joint de ces deux qubits est toujours représenté par un vecteur unité (et non de norme 2, par exemple).

Etats produits et intriqués

Si un qubit est déjà en soi un objet beaucoup plus riche qu'un bit classique, c'est encore plus vrai lorsqu'on parle de deux (qu-)bits !

Considérons quelques exemples d'états pour deux qubits :

- Il y a tout d'abord les états de base (qui représentent les états dits classiques), qu'on peut écrire sous la forme suivante:

$$|0,0\rangle = |0\rangle \otimes |0\rangle\,, \quad |0,1\rangle = |0\rangle \otimes |1\rangle\,, \quad |1,0\rangle = |1\rangle \otimes |0\rangle \quad \text{et} \quad |1,1\rangle = |1\rangle \otimes |1\rangle$$

La notation du produit tensoriel $\otimes$ revient ici : formellement, le produit tensoriel de deux états à 1 qubit $|\varphi_1\rangle = \alpha_0 |0\rangle + \alpha_1 |1\rangle$ et $|\varphi_2\rangle = \beta_0 |0\rangle + \beta_1 |1\rangle$ est défini ainsi :

$$|\varphi_1\rangle \otimes |\varphi_2\rangle = (\alpha_0 |0\rangle + \alpha_1 |1\rangle) \otimes (\beta_0 |0\rangle + \beta_1 |1\rangle) = \alpha_0\beta_0 |0,0\rangle + \alpha_0\beta_1 |0,1\rangle + \alpha_1\beta_0 |1,0\rangle + \alpha_1\beta_1 |1,1\rangle$$

On voit donc que le produit tensoriel se distribue comme un produit standard.

- Voici deux autres exemples d'états :

$$|\varphi\rangle = |+\rangle \otimes |+\rangle = \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle) \otimes \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle) = \frac{1}{2} (|0,0\rangle + |0,1\rangle + |1,0\rangle + |1,1\rangle)$$

et

$$|\varphi\rangle = |+\rangle \otimes |-\rangle = \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle) \otimes \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle) = \frac{1}{2} (|0,0\rangle - |0,1\rangle + |1,0\rangle - |1,1\rangle)$$

Tous les états vus jusqu'à présent sont ce qu'on appelle des *états produits*, i.e., des états de la forme $|\varphi\rangle = |\varphi_1\rangle \otimes |\varphi_2\rangle$, où $|\varphi_1\rangle\,, |\varphi_2\rangle \in \mathbb{R}^2$. Mais il existe aussi des états dans $\mathbb{R}^4$ qui ne peuvent pas se mettre sous cette forme : il s'agit d'*états intriqués*, comme par exemple l'*état de Bell* :

$$|\varphi\rangle = \frac{1}{\sqrt{2}} (|0,0\rangle + |1,1\rangle)$$

La particularité d'un tel état est que les deux qubits sont étroitement liés dans cet état : si le premier vaut 0, alors nécessairement le second vaut 0 (et il en va de même pour la valeur 1) : nous vous en dirons plus au prochain épisode...