Chapitre 4 : Complexité quantique

ENSIIE - Informatique quantique pour la recherche opérationnelle

Dimitri Watel (dimitri.watel@ensiie.fr)

2024

Plan

1. Complexité et classes de complexité
2. Simuler efficacement des machines quantiques
3. Bornes inférieures de complexité quantique

Algorithme de Deutsch-Jozsa : petit rappel

Soit une fonction $f : (\mathbb{Z}/2\mathbb{Z})^n \to \mathbb{Z}/2\mathbb{Z}$

Fonction équilibrée ou constante

- Une fonction **constante** si

$$\forall x, f(x) = 0 \text{ ou } \forall x, f(x) = 1$$

- Une fonction **équilibrée** si

$$\#\{x | f(x) = 0\} = \#\{x | f(x) = 1\} = 2^{n-1}$$

Problème à résoudre

Soit f constante ou équilibrée, f est-elle constante?

~~Suprématie~~ Avantage quantique théorique

Soit t la complexité en temps nécessaire pour calculer f

Meilleur algorithme classique

Si aucune information n'est donnée sur f, il n'existe pas d'algorithme déterministe classique dont la complexité soit meilleure que $\Omega(2^{n-1}t)$.

Algorithme quantique

Il existe un algorithme quantique dont la complexité est $O(t+n)$.

!! N'allons pas trop vite : complexité ? O Ω? !!

Complexité en temps classique

Complexité = mesure des ressources (ici le temps).

Définition : complexité classique

La complexité en temps d'un algorithme est une mesure qui, en fonction de l'entrée, donne le nombre d'opérations élémentaires nécessaires pour produire la sortie.

Un opérations élémentaires peut être définie arbitrairement selon les cas. Généralement:

- opérations arithmétiques ou booléennes,
- affectations, conditions, appel de fonction
- lecture dans un tableau ou d'un pointeur
- création d'un nombre aléatoire
- ...

O, Ω

(! Définitions peu rigoureuses)

La notation O

La notation $O(g)$ indique que la complexité est bornée par une fonction de l'ordre de g.
Il s'agit d'une borne supérieure de la complexité.

La notation Ω

La notation $\Omega(g)$ indique que la complexité domine une fonction de l'ordre de g.
Il s'agit d'une borne inférieure de la complexité.

Echec classique

Meilleur algorithme classique

Si aucune information n'est donnée sur f, il n'existe pas d'algorithme déterministe classique dont la complexité soit meilleure que $\Omega(2^{n-1}t)$.

Preuve (indices):

- Soit $\mathcal{A}$ qui effectue moins de 2^{n-1} appels à f
- Soient f_1 et f_2 telles que
  - $f_1(x_1) = f_1(x_2) = \cdots = f_1(x_{2^{n-1}}) = f_1(x_{2^{n-1}} + 1)$
  - $f_2(x_1) = f_2(x_2) = \cdots = f_2(x_{2^{n-1}}) \neq f_2(x_{2^{n-1}} + 1)$
- $\mathcal{A}$ n'a aucune information sur f
- Donc l'exécution de $\mathcal{A}$ ne peut différer quand $f = f_1$ ou f_2
- Donc $\mathcal{A}$ donne une mauvaise réponse pour f_1 ou f_2

Complexité quantique

Complexité d'un circuit

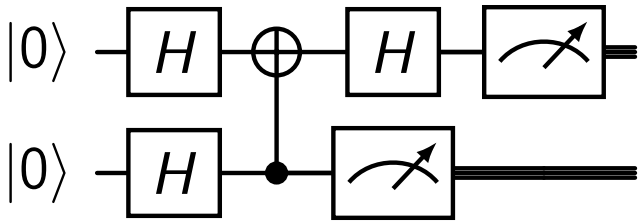
La complexité d'un circuit se mesure

- en nombre de portes : temps de construction
- en nombre de qbits n : temps d'initialisation
- en profondeur p : temps d'exécution

$m > p, n$ mais plusieurs exécutions du circuit

Complexité quantique

Exemple:



- $n = 2$,
- $m = 6$ si on compte CNOT pour 1, sinon 7
- $p = 4$

Complexité quantique

Un algorithme quantique est un algorithme classique qui construit et exécute un ou plusieurs circuits quantiques.

Complexité d'un algorithme quantique

Un algorithme qui exécute

- t opérations élémentaires classiques
- q fois k circuits de taille (m, n, p)

a une complexité

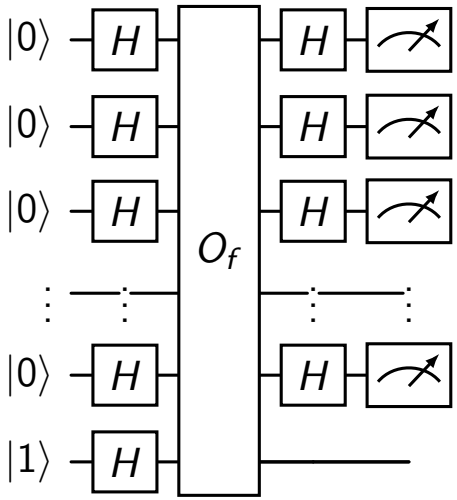
$$O(t + q(p + n) + k \cdot m)$$

Cette définition peut varier selon les modèles d'ordinateurs quantiques qu'on utilise.

Algorithme de Deutsch-Jozsa

Algorithme quantique

Il existe un algorithme quantique dont la complexité est $O(t+n)$.



- Nombre de porte et profondeur de O_f : $O(t)$
- Nombre de qbit : $n+1$
- Nombre de portes : $3n + O(t)$
- Profondeur : $3 + O(t)$
- $2n$ opérations classiques (lectures et comparaison de la sortie)
- Nombre de circuits construits et exécutés : 1

Complexité de l'algorithme : $O(t+n)$

Petit espoir pour le classique

Algorithme probabiliste

Pour tout $\varepsilon > 0$, il existe un algorithme probabiliste qui répond avec une probabilité meilleure que $1 - \varepsilon$ dont la complexité est $O(t + n)$.

- $k = \log(\frac{1}{\varepsilon}) + 1$
- Générer k entiers $x_1, x_2 \ldots, x_k$ distincts au hasard
- Vérifier si $f(x_1) = f(x_2) = \cdots = f(x_k)$
- Si oui répondre "f constante" sinon "f équilibrée"

Avantage quantique

L'algorithme quantique est strictement meilleur mais

pas d'avantage exponentiel pour ce problème.

2e essai. Algorithme de Bernstein-Vazirani : petit rappel

Soit un nombre binaire secret $s \in \{0,1\}^n$ et $f(x) = \underline{x} \odot \underline{s} = \bigoplus_{i=1}^{n} x_i s_i$.

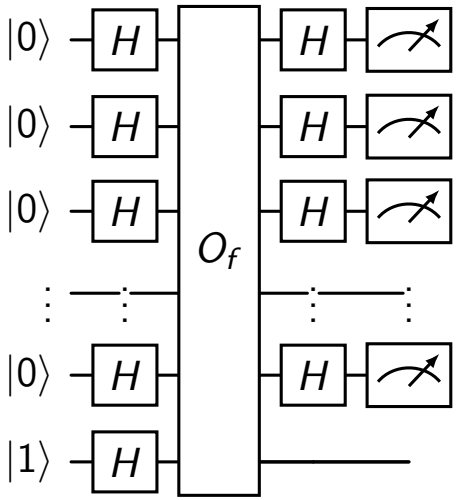
Problème à résoudre

Trouver s.

Complexité de l'algorithme de Bernstein-Vazirani

Algorithme quantique

Il existe un algorithme quantique dont la complexité est $O(n)$.



- Nombre de porte et profondeur de O_f : au pire $O(n)$ et $O(1)$
- Nombre de qbit : $n+1$
- Nombre de portes : $3n + O(n)$
- Profondeur : $O(1)$
- n opérations classiques (lectures et stockage de la sortie)
- Nombre de circuits construits et exécutés : 1

Complexité de l'algorithme : $O(n)$

Gros espoir pour le classique

Algorithme déterministe

Il existe un algorithme déterministe dont la complexité est $O(n)$.

Il suffit de mesurer $f(e_i)$ pour tous les vecteurs e_i de la base canonique pour trouver s.

Quel intérêt??

Problème plus difficile : Recursive Fourier Sampling

Soient $2^n + 1$ nombres binaires secrets $s \in \{0,1\}^n$ et $r_x \in \{0,1\}^n$ pour $x \in \{0,1\}^n$.
Soit

- $f(x,y) = \underline{r_x} \odot \underline{y}$
- $g(x,y)$ tel que $g(x,r_x) = \underline{s} \odot \underline{x}$

Problème à résoudre

Trouver s.

Algorithme classique

Algorithme déterministe

Il existe un algorithme déterministe dont la complexité est $O(n^2)$.

- Pour tout vecteur de la base canonique e_i, trouver r_{e_i} en $O(n)$.
- Pour tout vecteur de la base canonique e_i, calculer $f(e_i, r_{e_i})$

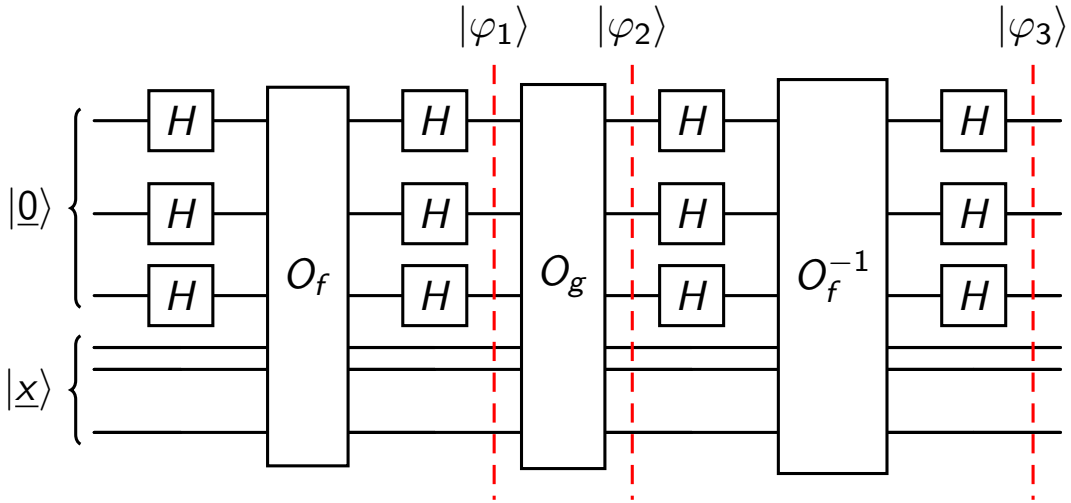
Meilleur algorithme probabiliste

Il n'existe pas d'algorithme probabiliste (ou déterministe) dont la complexité soit meilleur que $\Omega(n^2)$.

Algorithme quantique : oracle

Oracle $g(x, r_x)$

Il existe un circuit quantique qui, avec $|\underline{0}\rangle \otimes |\underline{x}\rangle$, 2 appels à f et 1 appel à g renvoie le qbit $(-1)^{s \odot x} |\underline{0}\rangle \otimes |\underline{x}\rangle$.



$$|\varphi_1\rangle = \left|\underline{r_x}\right\rangle \otimes |\underline{x}\rangle$$

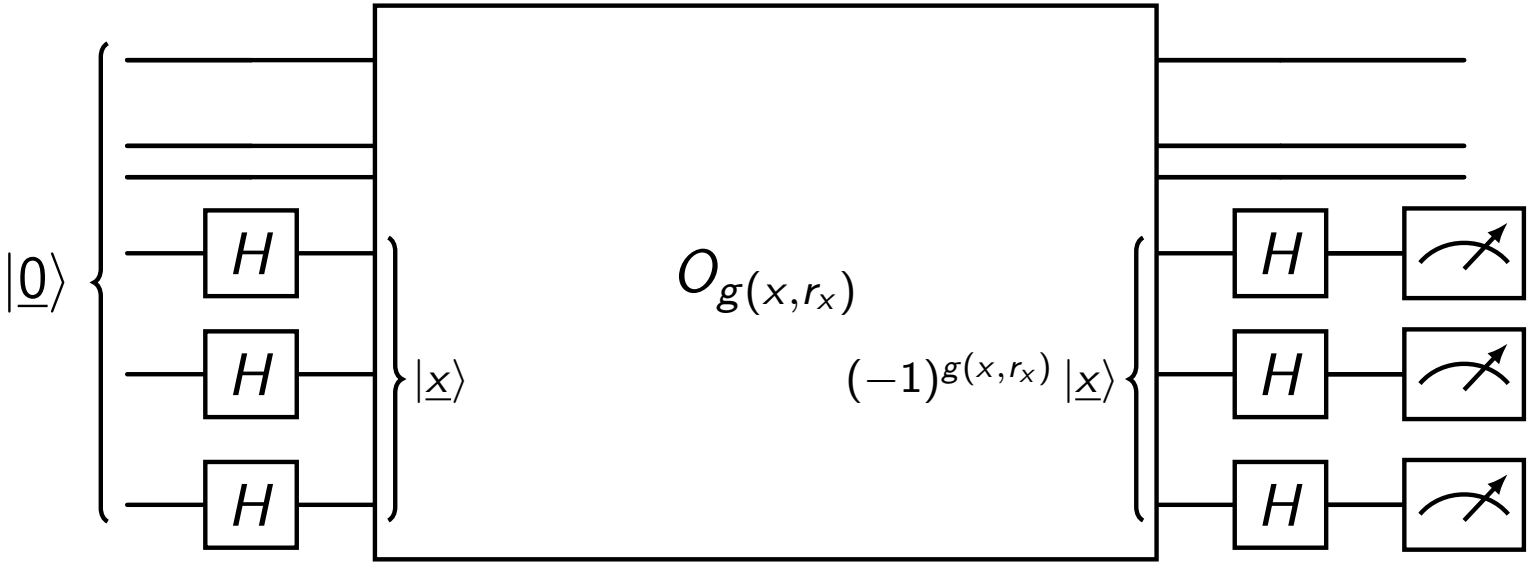
$$|\varphi_2\rangle = (-1)^{g(x,r_x)} \cdot \left|\underline{r_x}\right\rangle \otimes |\underline{x}\rangle$$

$$|\varphi_3\rangle = (-1)^{g(x,r_x)} \cdot |\underline{0}\rangle \otimes |\underline{x}\rangle$$

Algorithme quantique

Algorithme de Bernstein Vazirani récursif

L'algorithme de BV récursif résout le problème en $O(n)$.



Algorithme quantique

- Nb portes et profondeur de $O_{g(x,r_x)}$: au pire $O(n)$ et $O(1)$
- Nombre de qbit : $2n$
- Nombre de portes : $2n + O(n)$
- Profondeur : $3 + O(1)$
- n opérations classiques (lectures et stockage de la sortie)
- Nombre de circuits construits et exécutés : 1

Complexité de l'algorithme : $O(n)$

Conclusion

Meilleur algorithme probabiliste

Il n'existe pas d'algorithme probabiliste (ou déterministe) dont la complexité soit meilleur que $\Omega(n^2)$.

Algorithme quantique

Il existe un algorithme quantique dont la complexité est $O(n)$.

$\Rightarrow$ Gain quadratique

Peut-on faire mieux ?

Plus de niveaux

Niveau supplémentaire

Quand on rajoute un niveau:

- On duplique le circuit précédent (sauf les mesures)
- On rajoute un oracle O_g de taille $O(n)$ au milieu et $2n$ portes H autours

$P(i)$: la profondeur au niveau i (sans les mesures)

$$P(i+1) = 2P(i) + 3$$
$$\Rightarrow P(i) = 2^{i-1}P(1) + (2^i - 1) \cdot 3 = O(2^i)$$

Complexité en temps $\simeq nP(i) = O(n2^i)$

Avec $\log(n)$ niveaux

Meilleur algorithme probabiliste

Il n'existe pas d'algorithme probabiliste (ou déterministe) dont la complexité soit meilleur que $\Omega(n^{\log(n)})$.

Algorithme quantique

Il existe un algorithme quantique dont la complexité est $O(n2^{\log(n)}) = O(n^2)$.

$\Rightarrow$ Gain exponentiel !!!

Classification des problèmes de décision

Classe P

Ensemble des problèmes que l'on peut résoudre en temps polynomial ($O(n), O(n^2), \dots$).

Classe PSPACE

Ensemble des problèmes que l'on peut résoudre en espace polynomial ($O(n), O(n^2), \dots$).

$$P \subseteq BPP \subseteq PSPACE$$

Classification probabiliste des problèmes de décision

Classe BPP

Ensemble des problèmes pour lesquels il existe un algorithme probabiliste polynomial qui répond correctement avec une probabilité au moins $O(\frac{2}{3})$.

Répéter un tel algorithme fait tendre la probabilité vers 1.
BPP est une extension de P avec l'opération élémentaire Générer un bit aléatoire.

$$P \subseteq BPP \subseteq PSPACE$$

Classification quantique des problèmes de décision

Classe BQP

Ensemble des problèmes pour lesquels il existe un algorithme quantique polynomial qui répond correctement avec une probabilité au moins $O(\frac{2}{3})$.

Répéter un tel algorithme fait tendre la probabilité vers 1.
BQP est une extension de BPP.

$$P \subseteq BPP \subseteq BQP \subseteq PSPACE$$

Exemples

Problèmes dans P

Savoir si une liste est triée, Chercher une plus court chemin dans un graphe, Construire un flot maximum, Résoudre un programme linéaire, ...

Exemple pour BPP

Problème de Deutsch-Jozsa (problème avec oracle).
Soient 3 circuits algébriques, est-ce que exactements deux de ces circuits produisent les mêmes résultats ?

Exemple pour BQP

Problème de Bernstein-Vazirani (problème avec oracle).
Factorisation d'entiers
(! ce ne sont pas des problèmes de décision)

Non déterminisme à Camelot

- Merlin et Arthur discutent.
- Merlin a une puissance infinie, pas Arthur.
- Il demande à Merlin si la réponse est OUI.
- Merlin lui dicte un message.
- Arthur peut étudier le message pour répondre OUI ou NON.

À Camelot

Un problème est dans NP si,

- quand la réponse est OUI, il existe un message de Merlin tel que Arthur comprendra le message et répondra OUI.
- quand la réponse est NON, pour tout message de Merlin, Arthur comprendra le message et répondra NON.

Merlin = Opérations non déterministes <u>Choix</u> de l'algorithme
Arthur = Tout le reste

Reexemple

Entrées: n
Sorties: Oui si n est n'est pas premier, Non sinon

Merlin génère un entier s
Arthur vérifie si s divise n
Si s divise n **Alors**
    Renvoyer Oui
Sinon
    Renvoyer Non

MA : version Camelot

Classe BPNP ou MA - Merlin Arthur

Arthur dispose maintenant d'un générateur aléatoire.
Un problème est dans MA si,

- quand la réponse est OUI, il existe un message de Merlin tel que Arthur comprendra le message et répondra OUI avec une probabilité au moins $\frac{2}{3}$.
- quand la réponse est NON, pour tout message de Merlin, Arthur comprendra le message et répondra NON une probabilité au moins $\frac{2}{3}$.

$$P \subseteq BPP \subseteq MA \subseteq PSPACE$$

$$P \subseteq NP \subseteq MA \subseteq PSPACE$$

QMA : version Camelot

Classe QMA - Quantum Merlin Arthur

Arthur ne rigole plus et dispose d'un ordinateur quantique. Merlin envoie des qbits et non des bits classiques.

Un problème est dans QMA si,

- quand la réponse est OUI, il existe un message de Merlin tel que Arthur comprendra le message et répondra OUI avec une probabilité au moins $\frac{2}{3}$.
- quand la réponse est NON, pour tout message de Merlin, Arthur comprendra le message et répondra NON avec une probabilité au moins $\frac{2}{3}$.

$$P \subseteq BPP \subseteq BQP \subseteq QMA; P \subseteq BPP \subseteq MA \subseteq QMA \subseteq PSPACE$$

$$P \subseteq NP \subseteq MA \subseteq QMA \subseteq PSPACE$$

Exemples de problème dans toutes ces classes

Problèmes dans QMA

- (DIST) : Soit un opérateur quantique Φ, existe-t-il deux entrées $|\underline{x}\rangle$ et $|\underline{y}\rangle$ dont les sorties $\Phi(|\underline{x}\rangle)$ et $\Phi(|\underline{y}\rangle)$ sont *distinguables* ?
- (ID) : Soit un circuit quantique C, est-ce que C produit un résultat *distinguable* de l'identité ?

Remarque : on suppose que (ID) $\notin$ BQP ; mais sa version classique est dans BPP.

Pour aller plus loin

Autres classes

- RP, PP, RQP, PQP
- ZPP, ZQP, EQP
- BPP-Path
- AM ou AM[k], QAM, QAM[k]
- P-CTC et BQP-CTC (voyage dans le temps)

$BPP \neq BQP$?

Théorème

Il existe un problème avec oracle qui est dans BQP et qui n'est pas dans BPP.

- Version généralisée du problème de Bernstein-Vazirani.
- Pas de résultat sans oracle

$QMA \subseteq PSPACE$

Théorème

$QMA \subseteq PSPACE$

- Pour chaque message possible de Merlin, simuler les réponses d'Arthur en espace polynomial.

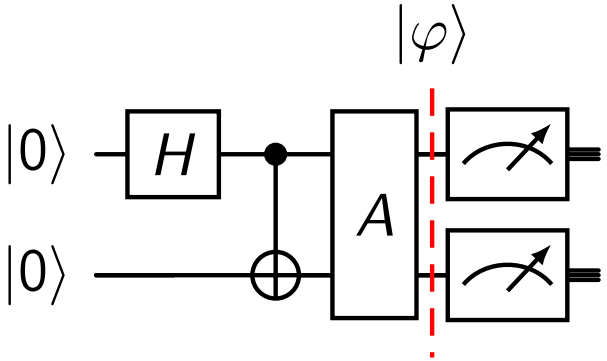
Comment simuler les réponses en espace polynomial ???

Plan

1. Complexité et classes de complexité
2. Simuler efficacement des machines quantiques
3. Bornes inférieures de complexité quantique

Simulation naïve

Essayons de simuler : il suffit de calculer l'état de sortie $|\varphi\rangle$.



avec

$$A = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ \frac{1+i}{2} & \frac{-1-i}{2} & \frac{1-i}{2} & \frac{-1+i}{2} \\ \frac{1-i}{2} & \frac{-1+i}{2} & \frac{1+i}{2} & \frac{-1-i}{2} \end{pmatrix}.$$

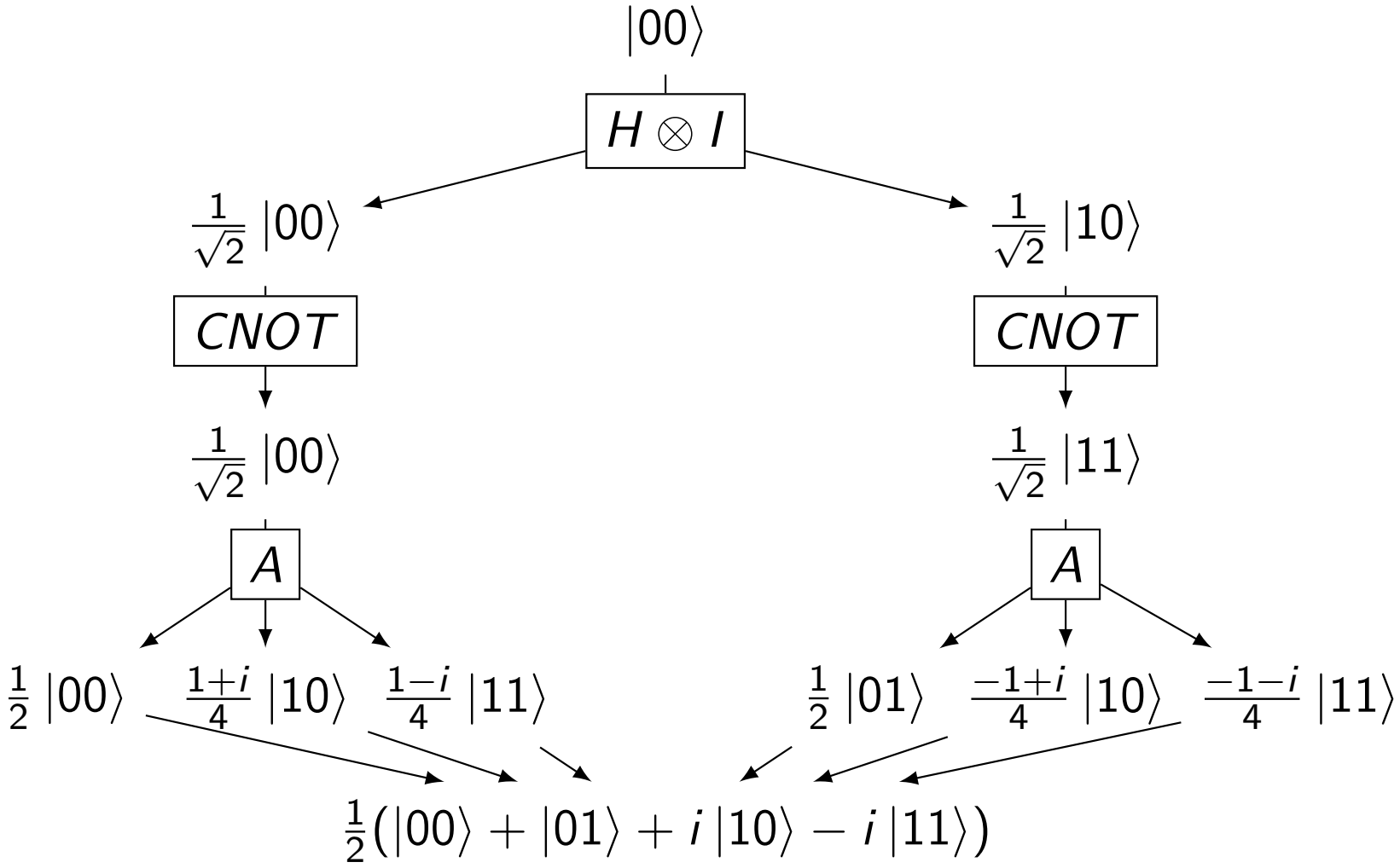
$$|\varphi\rangle = \frac{1}{2}(|00\rangle + |01\rangle + i\,|10\rangle - i\,|11\rangle)$$

On simule ensuite sa mesure.

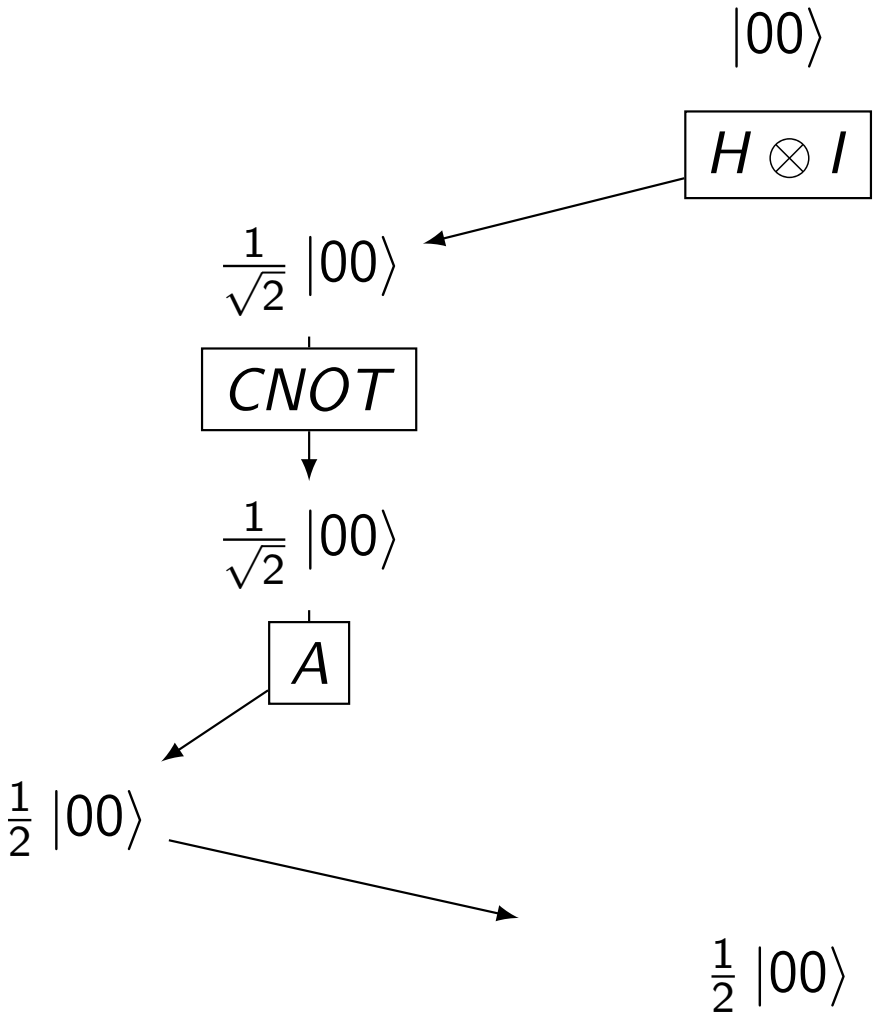
Complexité de la méthode : exponentiel en temps et en espace.
A chaque étape, on maintient au pire un n-qbit en mémoire.

Simulation naïve équivalente

On peut exhiber cet arbre de calcul:



Simulation naïve équivalente



Simulation naïve équivalente

$$|00\rangle$$

$$\boxed{H \otimes I}$$

$$\frac{1}{\sqrt{2}}|00\rangle$$

$$\boxed{CNOT}$$

$$\frac{1}{\sqrt{2}}|00\rangle$$

$$\boxed{A}$$

$$\frac{1+i}{4}|10\rangle$$

$$\frac{1}{2}(|00\rangle + \frac{1+i}{2}|10\rangle)$$

Simulation naïve équivalente

$$|00\rangle$$

$$\boxed{H \otimes I}$$

$$\frac{1}{\sqrt{2}}|00\rangle$$

$$\boxed{CNOT}$$

$$\frac{1}{\sqrt{2}}|00\rangle$$

$$\boxed{A}$$

$$\frac{1-i}{4}|11\rangle$$

$$\frac{1}{2}\left(|00\rangle + \frac{1+i}{2}|10\rangle + \frac{1-i}{2}|11\rangle\right)$$

Simulation naïve équivalente

$$|00\rangle$$

$$\boxed{H \otimes I}$$

$$\frac{1}{\sqrt{2}}|10\rangle$$

$$\boxed{CNOT}$$

$$\frac{1}{\sqrt{2}}|11\rangle$$

$$\boxed{A}$$

$$\frac{1}{2}|01\rangle$$

$$\frac{1}{2}(|00\rangle + |01\rangle + \frac{1+i}{2}|10\rangle + \frac{1-i}{2}|11\rangle)$$

Simulation naïve équivalente

$$|00\rangle$$

$$\boxed{H \otimes I}$$

$$\frac{1}{\sqrt{2}}|10\rangle$$

$$\boxed{CNOT}$$

$$\frac{1}{\sqrt{2}}|11\rangle$$

$$\boxed{A}$$

$$\frac{-1+i}{4}|10\rangle$$

$$\frac{1}{2}(|00\rangle + |01\rangle + i\,|10\rangle + \frac{1-i}{2}|11\rangle)$$

Simulation naïve équivalente

$$|00\rangle$$

$$\boxed{H \otimes I}$$

$$\frac{1}{\sqrt{2}}|10\rangle$$

$$\boxed{CNOT}$$

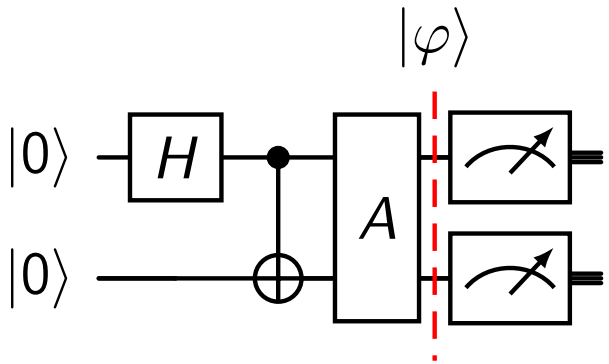
$$\frac{1}{\sqrt{2}}|11\rangle$$

$$\boxed{A}$$

$$\frac{-1-i}{4}|11\rangle$$

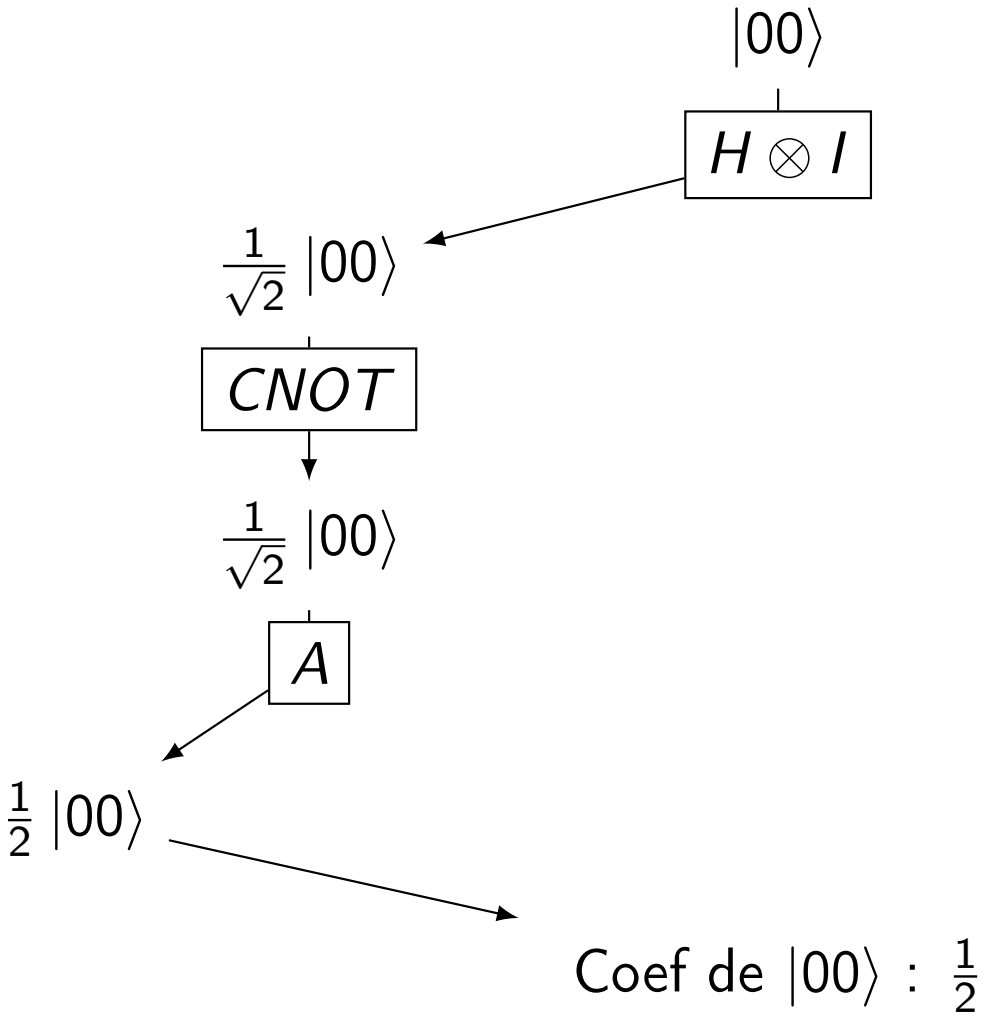
$$\frac{1}{2}(|00\rangle + |01\rangle + i|10\rangle - i|11\rangle)$$

Simulation efficace en espace



On peut faire mieux : objectif, simuler la mesure de $|\varphi\rangle$, pas trouver $|\varphi\rangle$.
On tire un nombre aléatoire r entre 0 et 1, supposons $r = 0.66$.

Simulation efficace en espace



Simulation efficace en espace

$$|00\rangle$$

$$\boxed{H \otimes I}$$

$$\frac{1}{\sqrt{2}}|00\rangle$$

$$\boxed{CNOT}$$

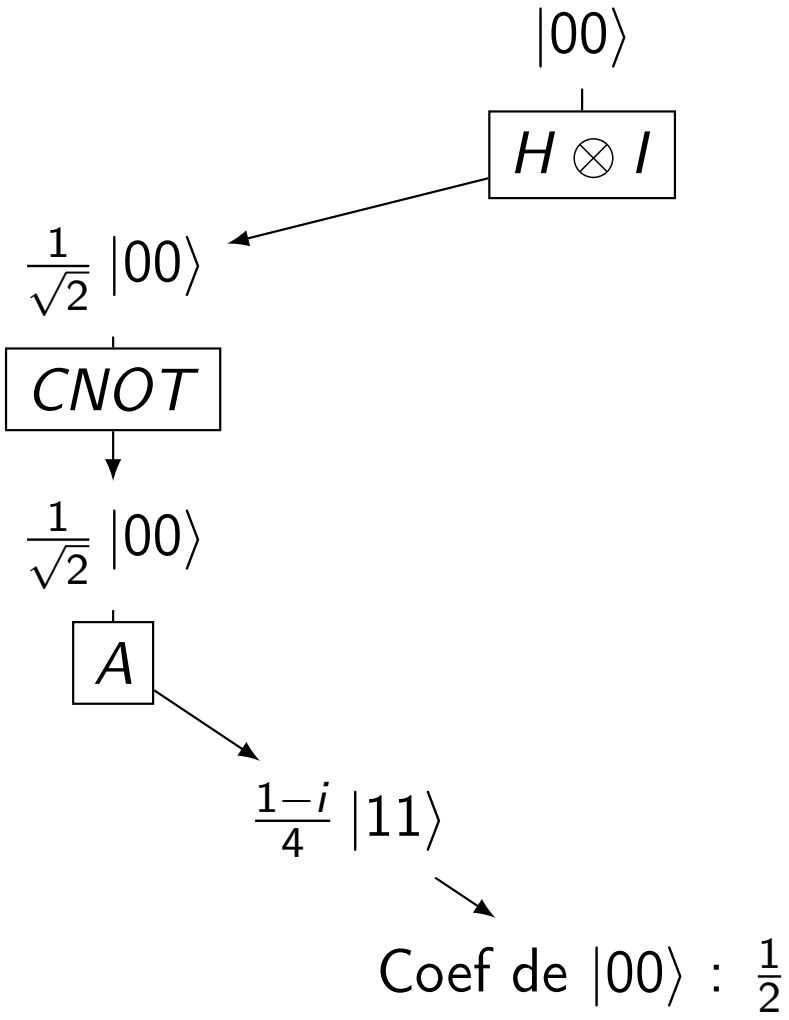
$$\frac{1}{\sqrt{2}}|00\rangle$$

$$\boxed{A}$$

$$\frac{1+i}{4}|10\rangle$$

Coef de $|00\rangle$: $\frac{1}{2}$

Simulation efficace en espace



Simulation efficace en espace

$$|00\rangle$$

$$\boxed{H \otimes I}$$

$$\frac{1}{\sqrt{2}}|10\rangle$$

$$\boxed{CNOT}$$

$$\frac{1}{\sqrt{2}}|11\rangle$$

$$\boxed{A}$$

$$\frac{1}{2}|01\rangle$$

Coef de $|00\rangle$: $\frac{1}{2}$

Simulation efficace en espace

$$|00\rangle$$

$$\boxed{H \otimes I}$$

$$\frac{1}{\sqrt{2}}|10\rangle$$

$$\boxed{CNOT}$$

$$\frac{1}{\sqrt{2}}|11\rangle$$

$$\boxed{A}$$

$$\frac{-1+i}{4}|10\rangle$$

Coef de $|00\rangle$: $\frac{1}{2}$

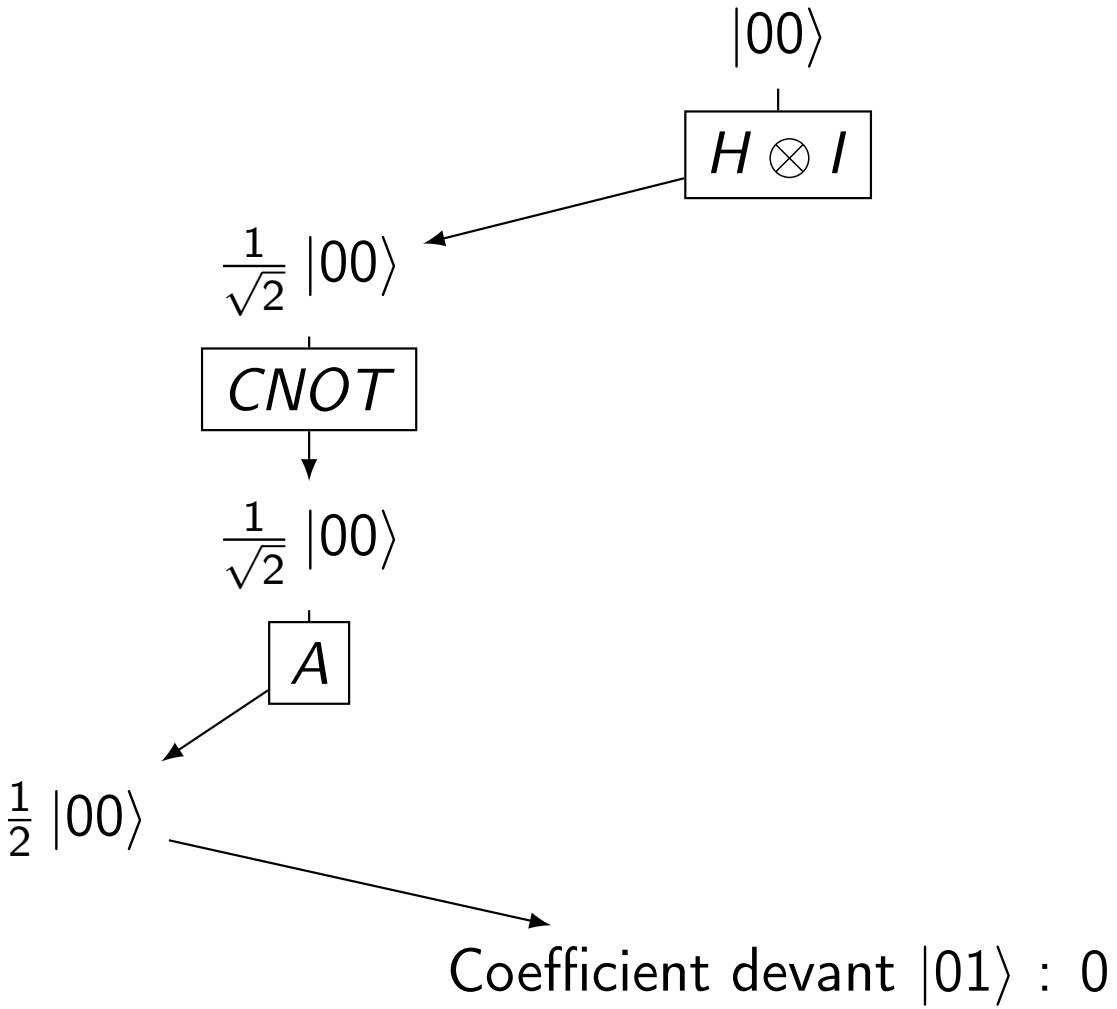
Simulation efficace en espace

$$|00\rangle \xrightarrow{H \otimes I} \frac{1}{\sqrt{2}}|10\rangle \xrightarrow{CNOT} \frac{1}{\sqrt{2}}|11\rangle \xrightarrow{A} \frac{-1-i}{4}|11\rangle$$

Proba de mesurer 00 : $\frac{1}{4} < r = 0.66$
$\Rightarrow r \leftarrow 0.66 - 0.25 = 0.41$
On continue avec 01.

Coef de $|00\rangle$: $\frac{1}{2}$

Simulation efficace en espace



Simulation efficace en espace

$$|00\rangle$$

$$\boxed{H \otimes I}$$

$$\frac{1}{\sqrt{2}}|00\rangle$$

$$\boxed{CNOT}$$

$$\frac{1}{\sqrt{2}}|00\rangle$$

$$\boxed{A}$$

$$\frac{1+i}{4}|10\rangle$$

Coefficient devant $|01\rangle$: 0

Simulation efficace en espace

$$|00\rangle$$

$$\boxed{H \otimes I}$$

$$\frac{1}{\sqrt{2}}|00\rangle$$

$$\boxed{CNOT}$$

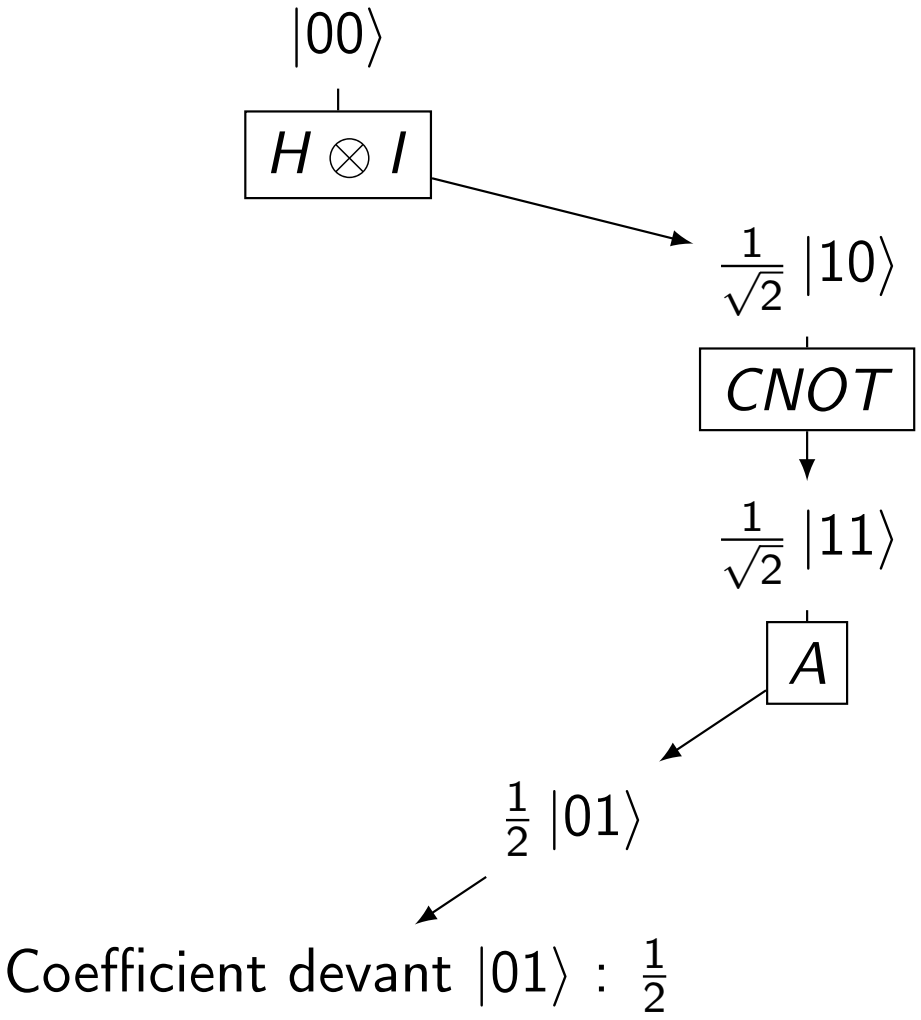
$$\frac{1}{\sqrt{2}}|00\rangle$$

$$\boxed{A}$$

$$\frac{1-i}{4}|11\rangle$$

Coefficient devant $|01\rangle$: 0

Simulation efficace en espace



Simulation efficace en espace

$$|00\rangle$$

$$\boxed{H \otimes I}$$

$$\frac{1}{\sqrt{2}}|10\rangle$$

$$\boxed{CNOT}$$

$$\frac{1}{\sqrt{2}}|11\rangle$$

$$\boxed{A}$$

$$\frac{-1+i}{4}|10\rangle$$

Coefficient devant $|01\rangle$: $\frac{1}{2}$

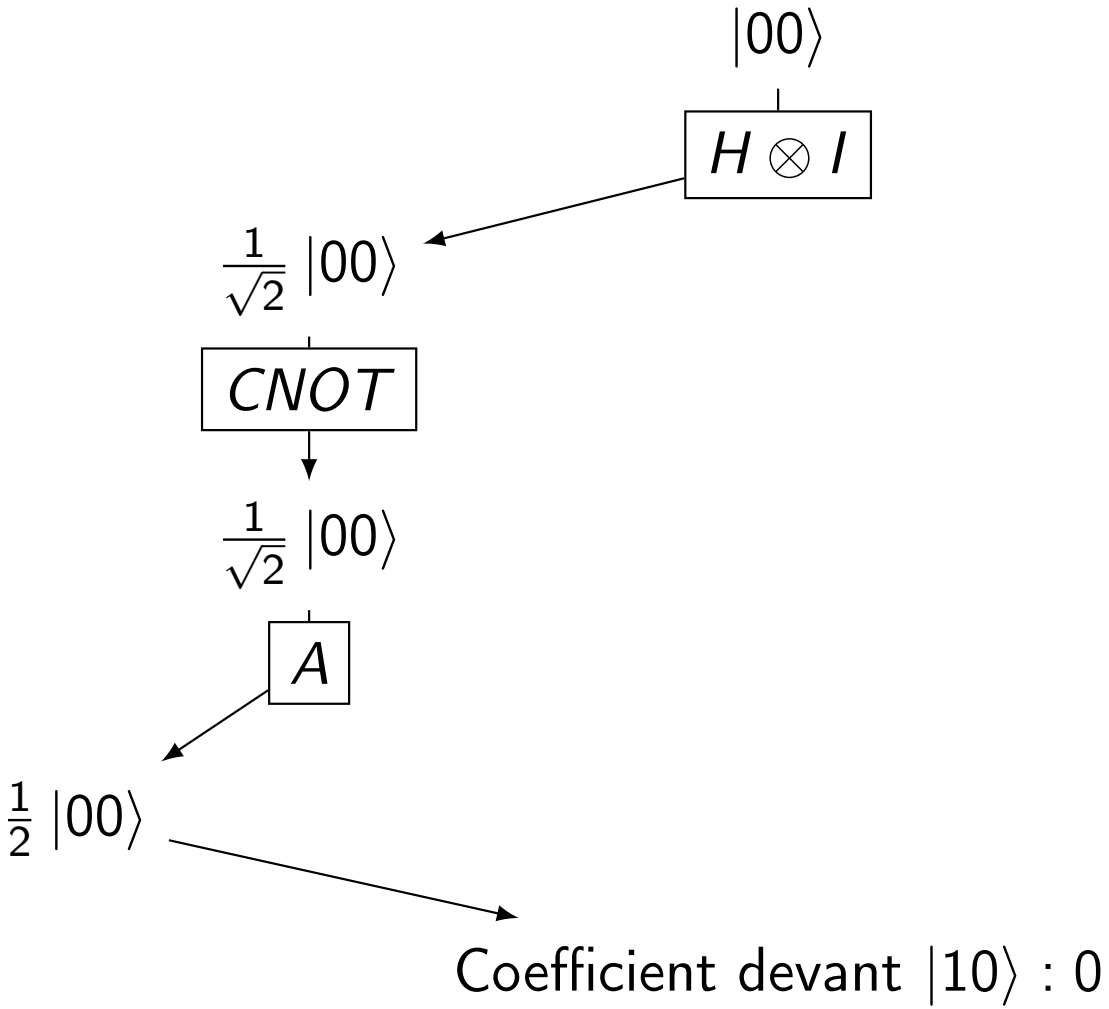
Simulation efficace en espace

$$|00\rangle \xrightarrow{H \otimes I} \frac{1}{\sqrt{2}}|10\rangle \xrightarrow{CNOT} \frac{1}{\sqrt{2}}|11\rangle \xrightarrow{A} \frac{-1-i}{4}|11\rangle$$

Proba de mesurer 01 : $\frac{1}{4} < r = 0.41$
$\Rightarrow r \leftarrow 0.41 - 0.25 = 0.16$
On continue avec 10.

Coefficient devant $|01\rangle$: $\frac{1}{2}$

Simulation efficace en espace



Simulation efficace en espace

$$|00\rangle$$

$$\boxed{H \otimes I}$$

$$\frac{1}{\sqrt{2}}|00\rangle$$

$$\boxed{CNOT}$$

$$\frac{1}{\sqrt{2}}|00\rangle$$

$$\boxed{A}$$

$$\frac{1+i}{4}|10\rangle$$

Coefficient devant $|10\rangle$: $\frac{1+i}{4}$

Simulation efficace en espace

$$|00\rangle$$

$$\boxed{H \otimes I}$$

$$\frac{1}{\sqrt{2}}|00\rangle$$

$$\boxed{CNOT}$$

$$\frac{1}{\sqrt{2}}|00\rangle$$

$$\boxed{A}$$

$$\frac{1-i}{4}|11\rangle$$

Coefficient devant $|10\rangle$: $\frac{1+i}{4}$

Simulation efficace en espace

$$|00\rangle$$

$$\boxed{H \otimes I}$$

$$\frac{1}{\sqrt{2}}|10\rangle$$

$$\boxed{CNOT}$$

$$\frac{1}{\sqrt{2}}|11\rangle$$

$$\boxed{A}$$

$$\frac{1}{2}|01\rangle$$

Coefficient devant $|10\rangle$: $\frac{1+i}{4}$

Simulation efficace en espace

$$|00\rangle$$

$$\boxed{H \otimes I}$$

$$\frac{1}{\sqrt{2}}|10\rangle$$

$$\boxed{CNOT}$$

$$\frac{1}{\sqrt{2}}|11\rangle$$

$$\boxed{A}$$

$$\frac{-1+i}{4}|10\rangle$$

Coefficient devant $|10\rangle$: $\frac{i}{2}$

Simulation efficace en espace

$$|00\rangle \xrightarrow{H \otimes I} \frac{1}{\sqrt{2}}|10\rangle \xrightarrow{CNOT} \frac{1}{\sqrt{2}}|11\rangle \xrightarrow{A} \frac{-1-i}{4}|11\rangle$$

Proba de mesurer 10 : $\frac{1}{4} > r = 0.16$
$\Rightarrow$ On renvoie 10.

Coefficient devant $|10\rangle$: $\frac{i}{2}$

A un instant donné, on ne garde pas tout l'état final superposé en mémoire, juste le coefficient devant un état de base déterminé ; et un seul chemin de l'arbre.

Théorème

On peut simuler tout circuit quantique en espace polynomial.

Théorème

$QMA \subseteq PSPACE$

Petit problème : les grosses portes

A est une matrice de taille $2^n \times 2^n$. $\Rightarrow$ Espace exponentiel nécessaire pour stocker le circuit ?

Portes universelles

Pour tout $\varepsilon > 0$, on peut approcher A à ε près avec les portes de Toffoli et de Hadamard.

Bonne nouvelle ?
Oui et Non! $\Rightarrow$ $O(4^n)$ portes pour coder A.

Intérêt?

Grâce aux portes universelles, un ordinateur quantique peut ne contenir que des portes de Toffoli et de Hadamard. Il suffit ensuite de pouvoir faire passer les qbits dans un nombre exponentiels de telles portes.

Autres portes universelles

- $CNOT$ et toutes les portes sur 1 qbit : $R_x(\theta)$, $R_y(\theta)$, $R_z(\theta)$, $P(\varphi)$ pour tout θ et tout φ
- $CNOT$, H, $S = \sqrt{Z}$ et $T = \sqrt[4]{Z}$
- $CNOT$, $R_y(\pi/4)$, $P(\pi/2)$
- Avec une porte A sur 2 qbit choisie uniformément, $CNOT$, A, $P(\pi/2)$ a une forte probabilité d'être un ensemble universel.

Source : `https://www.scottaaronson.com/qclec/16.pdf`

Plan

1. Complexité et classes de complexité
2. Simuler efficacement des machines quantiques
3. Bornes inférieures de complexité quantique

Problème de recherche

Problème à résoudre

Soit un nombre binaire secret $s \in \{0,1\}^n$ et $f(x) = 1$ si $x = s$ et 0 sinon. Trouver s.

Borne supérieure

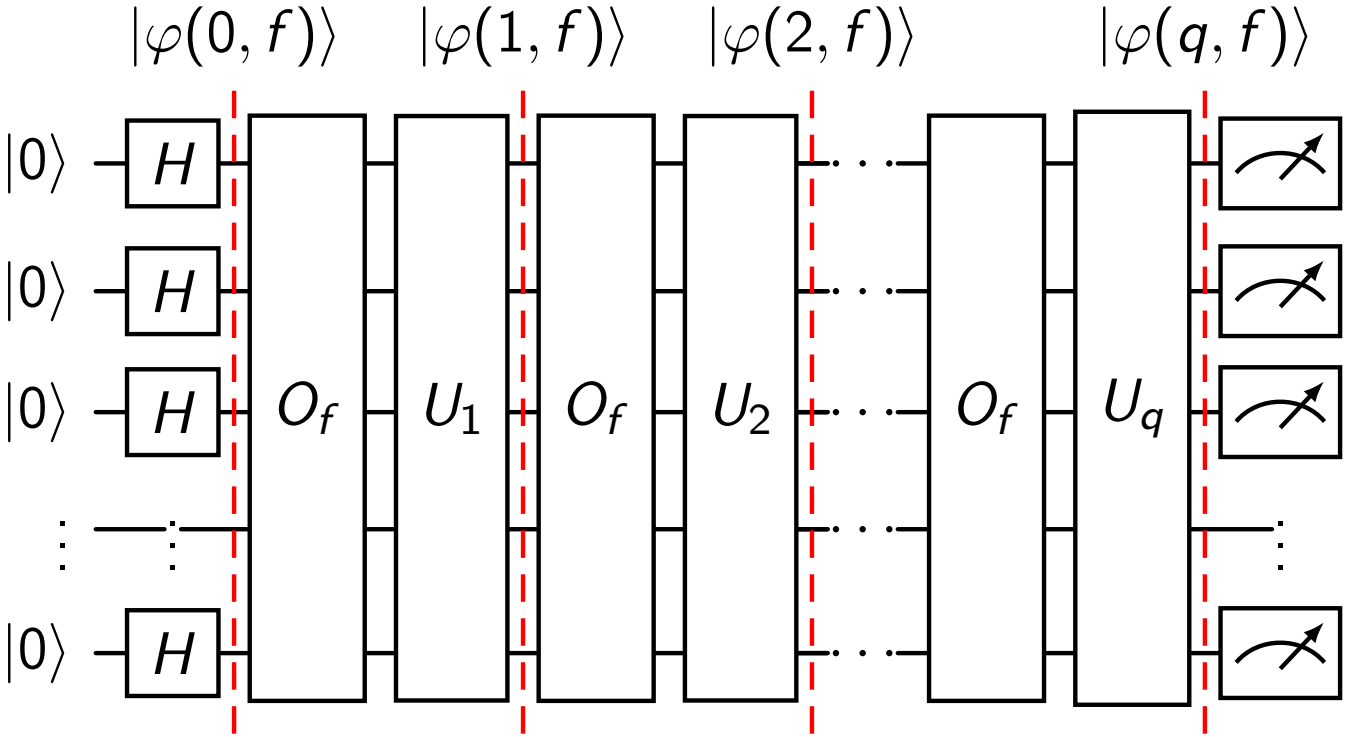
L'algorithme quantique de Grover résout ce problème en faisant au plus $O(\sqrt{2^n})$ appels à f.

Borne inférieure

Peut-on prouver que Grover est optimal en temps ?
i.e. tout algorithme quantique qui résout ce problème doit faire $\Omega(\sqrt{2^n})$ appels à f.

Circuit abstrait

Supposons que le circuit suivant résout le problème.



On peut borner inférieurement la valeur de q.

Remarques de bon sens

On note f_s la fonction telle que $f(s) = 1$ et $f(x \neq s) = 0$.

Constatation

- Si $y \neq z$ alors $\left|\underline{y}\right\rangle \perp |\underline{z}\rangle$.
- $|\varphi(q, f_y)\rangle \simeq \left|\underline{y}\right\rangle$
- $|\varphi(q, f_z)\rangle \simeq |\underline{z}\rangle$
- $|\varphi(0, f_y)\rangle = |\varphi(0, f_z)\rangle$

Conclusion

q dépend de la capacité des portes U_i à éloigner $|\varphi(i, f_y)\rangle$ et $|\varphi(i, f_z)\rangle$ pour tout y et z.

Résultats

Soit zr la fonction constante égale à 0.

Lemme

$$\sum_{y=0}^{2^n-1} \| |\varphi(q, zr)\rangle - |\varphi(q, f_y)\rangle \| \geq \sqrt{2} \cdot (2^n - 1)$$

Lemme

$$\sum_{y=0}^{2^n-1} \| |\varphi(q, zr)\rangle - |\varphi(q, f_y)\rangle \| \leq 2\sqrt{2^n} q$$

Theoreme

$$q = \Omega(\sqrt{2^n})$$