Bornes inférieures de fonctions booléennes

Basile Couëtoux

Résumé

Nous avons cherché à mettre en évidence les limites d'une borne inférieure utilisée dans le cadre d'une preuve affinant la borne inférieure connue sur la complexité en requêtes des propriétés de graphes monotones. Nous avons de même cherché ce que donnait la preuve équivalente en quantique en utilisant des outils nouveaux permettant de transferer des résultats existant dans le modèle probabiliste dans le modèle quantique, et aussi d'apporter des preuves dans le modèle probabilistes à l'aide de résultats quantiques.

1 Introduction

La difficulté calculatoire d'un problème ou d'une fonction peut être mesurée selon différents critères. Le modèle de calcul le plus connu est celui de la machine de Turing. Dans ce modèle la complexité en temps et en espace d'un algorithme peut être étudiée. Cependant les complexités optimales en temps et en espaces restent inconnues pour de nombreux problèmes. La complexité en requêtes permet d'apporter des informations d'une manière différentes sur la complexité d'une fonction. La complexité en requêtes d'une fonction est le nombre nécessaire de requêtes sur l'entrée pour calculer la valeur de la fonction dans le pire des cas sur les entrées possibles. Les requêtes ne doivent pas trivialiser le problème. Nous nous intéresserons uniquement à la complexité en requêtes.

Le calcul de la complexité en requêtes d'une fonction se décompose en deux parties. La borne supérieure consiste à donner un algorithme qui calcule la fonction et ainsi établir la complexité de cet algorithme comme une borne supérieure de la complexité. La borne inférieure consiste à montrer que la fonction n'est pas calculable en faisant moins de requêtes sur l'entrée que cette borne dans le pire des cas. Si la borne inférieure et la borne supérieure sont identiques, la complexité en requêtes de cette fonction est déterminée.

Les problèmes de borne inférieure en complexité sont des résultats qui se distinguent en général dans l'informatique théorique par leur coté non constructif. Les bornes inférieures peuvent être démontrées à l'aide de réductions à d'autres problèmes, ou bien directement à l'aide d'outils développés à cet effet, c'est d'ailleurs ces outils que nous considérons ici. Nous utilisons des méthodes récentes de bornes inférieures qui ont souvent deux versions, une quantique, une probabiliste. Ces méthodes s'inscrivent dans le courant de l'algorithmique quantique qui démontre des résultats en probabiliste grâce à ses méthodes

Le modèle de complexité en requête est l'arbre de décision.

Nous nous sommes intéressés à la conjecture de "Graph Evasiveness".

> Un algorithme calculant une propriété de graphe monotone sur un graphe de taille v a besoin $\frac{n(n-1)}{2}$ requêtes.

Par exemple il est nécessaire de connaître l'intégralité d'un graphe pour déterminer si il y a un triangle dans ce graphe.

Dans le cas d'un algorithme déterministe il a été démontré que si v est premier alors on a besoin de demander l'intégralité des arêtes pour déterminer cette propriété de graphe. Ce qui implique que la conjecture est vraie dans le cas déterministe . On a la même conjecture dans le cas probabiliste cependant la borne inférieure actuelle est de $\widetilde{\Theta}(v^{4/3})$ (avec '$\widetilde{\Theta}$' qui dénote la suppression des facteurs logarithmique). La conjecture dans le cas quantique est que la complexité des propriétés de graphe monotones s'échelonnent entre $\Theta(v)$ et $\Theta(v^2)$ mais la borne inférieure actuelle découle directement de celle qui existe en probabiliste et qui est $\Theta(v^{2/3})$.

2 plan

- Etude d'une borne inférieure probabiliste
- Limitation de cette borne
- Description d'une méthode liant des bornes pobabiliste à des bornes quantiques
- Equivalent partiel de la borne probabiliste dans le modèle quantique

3 Préliminaires

3.1 Distribution sur les entrées

Les entrées considérées sont des mots binaires de taille n, n restera un paramètre dans le reste du rapport. Si $x \in \{0,1\}^n$, alors $x[i]$ est la valeur du i-ème bit mot x.
Soit $s_i(x)$ le mot dont la i-ème lettre a été changée : $(s_i(x))[i] = 1 - x[i]$

Le poids utilisé sur ces mots est le poids de Hamming :

$$|x| = card(i : x[i] = 1)$$

Soit μ la distribution sur $\{0,1\}^n$ paramétrée par p :

$$\mu(x) = p^{|x|}(1-p)^{n-|x|}$$

La distribution est celle obtenue par n tirages indépendants de bits, telle que pour chaque bit la probabilité d'avoir un 1 est p et la probabilité d'avoir un 0 est $1 - p$.
Soit x^i le mot dont la i-ème lettre a été remise aléatoirement : $P(x^i[i] = 1) = p$

Les théorèmes, auxquels nous nous sommes intéressés, se sont ramenés au cas des fonctions booléennes monotones transitives qui incluent les propriétés de graphes monotones pour étudier les résultats sur cette conjecture. Dans la suite du rapport nous considérerons aussi les fonctions monotones transitives.

Une fonction *monotone* f de $\{0,1\}^n$ dans $\{0,1\}$ est une fonction telle que pour tout $x, y \in \{0,1\}^n$ si $x[i] \leq y[i]$ alors $f(x) \leq f(y)$.

Une fonction *transitive* f de $\{0,1\}^n$ dans $\{0,1\}$ est une fonction telle que pour tout $i, j \in \{1..n\}$ il existe σ permutation de $\{1..n\}$ telle que $\sigma(i) = j$ et $f \circ \sigma = f$.

La transitivité représente le fait, que pour les propriétés de graphe, les sommets ne sont pas étiquetés. Mais par exemple, la transitivité ne préserve pas les chemins, ainsi les propriétés de graphes sont strictement inclues dans les fonctions transitives.

3.2 arbre de décision

Définition 1. *Un* arbre de décision déterministe *est un arbre binaire dont les nœuds internes sont étiquetés par des entiers et ont un fils gauche et un fils droit.*

Le parcours de cet arbre se fait de façon inductive. Lorsque l'on se trouve sur un nœud interne étiqueté par i, le i-ème bit de l'entrée est interrogé. Si ce i-ème bit vaut 1 alors le parcours continue sur le fils droit, sinon le parcours continue sur le fils gauche. Lorsqu'on se trouve sur une feuille le parcours s'arrête.

Définition 2. Un arbre de décision calcule une fonction *si pour toute entrée x, le parcours de l'arbre sur l'entrée x se termine par une feuille étiquetée $f(x)$*

Soit $d(T, x)$ la profondeur de la feuille que l'on atteint dans l'arbre T sur l'entrée x La profondeur de l'arbre $q(T)$ détermine sa complexité en requêtes, en effet la profondeur de l'arbre représente le nombre de requêtes dans le pire des cas : $d(T) = \max_x d(T, x)$. Nous utiliserons ce concept d'arbre de décision dans la seconde partie.

Définition 3. *Un* arbre de décision probabiliste *est une distribution d'arbres de décision déterministes. Soit T un arbre de décision probabiliste, distribuées sur $(T_i)_{i \in N}$ selon la distribution λ.*

Définition 4. Un arbre de décision probabiliste calcule une fonction f *si l'ensemble des T_i sont des arbres de décisions déterministes qui calculent f. Le nombre de requêtes effectuées sur une entrée x par un arbre de décision probabiliste est : $r(T, x) = \sum_i \lambda(i) d(T_i, x)$*

Définition 5. *Un* circuit quantique *est un circuit constitué d'une succession de portes unitaires indépendantes de l'entrée et de portes unitaires oracles accédant à un bit de l'entrée.*

Les circuits sont utilisés car pour représenter un arbre quantique on devrait superposer des arbres de décision a chaque étape du parcours de l'arbre.

Définition 6. *:*

1. *Soit $D(f)$ la plus petite profondeur d'un arbre de décision déterministe calculant f :*

$$D(f) = \min_{T \text{ calculant } f} d(T)$$

2. *Soit $R(f)$ la plus petite profondeur d'un arbre de décision probabiliste calculant f :*

$$R(f) = \min_{T \text{ calculant } f} r(T)$$

3. *Soit $Q(f)$ le nombre de porte oracle minimum dans un circuit quantique calculant f.*

La profondeur moyenne sur les entrées $\delta(T)$ d'un arbre de décision déterministe :

$$\delta(T) = \sum_{x \in \{0,1\}^n} \mu(x) d(T, x)$$

Ainsi la complexité en moyenne sur les entrées pour f dans le cas déterministe est :

$$\delta(f) = \min_{T \text{ calculant } f} \delta(T)$$

La profondeur en moyenne sur les entrées d'un arbre de décision probabiliste :

$$\rho(T) = \sum_i \lambda(i)\delta(T_i)$$

De même la complexité en moyenne sur les entrées de f dans le cas probabiliste est :

$$\rho(f) = \min_{T \text{ calculant } f} \rho(T)$$

Théorème 1 ([1]). *Par un principe de* Yao Min-max $\delta(f)$=$\rho(f)$

Ce théorèeme est expliqué aussi dans [**?**] Par la suite on associeras δ à ρ Dans cette perspective de profondeur moyenne,soit fonction $\delta_i(T,p)$ qui calcule la probabilité de lire le bit i dans l'arbre T quand les entrées sont distribuées selon μ.

3.3 Influence d'une fonction

L'*influence totale* est une mesure sur une fonction qui permet d'évaluer sa complexité.

$$\mathit{inf}(f,\mu) = \sum_{x,i:f(x)\neq f(x^i)} \mu(x)$$

L'influence totale d'une fonction peut être décomposée selon les *influences des bits*.

$$\mathit{inf}_i(f,\mu) = \sum_{x:f(x)\neq f(x^i)} \mu(x)$$

L'influence dénote la régularité d'une fonction, en effet une faible influence montrera que la fonction est constante sur de grandes parties connexes de l'hypercube $\{0,1\}^n$. De même l'influence du i-ème bit représente la probabilité que ce bit soit nécessaire pour calculer la fonction.

Nous mettrons en rapport avec la variance définie comme suit pour les fonctions booléenne

$$Var(f,p) = 4P_{x:\mu}(f(x)=1)P_{x:\mu}(f(x)=0) = E_{x,y:\mu}(f(x)\neq f(y))$$

En considérant la fonction $F = (-1)^f$:

$$\mathit{Var}(f,p) = E_{x:\mu}(F(x)^2) - E_{x:\mu}(F(x))^2$$

3.4 Exemples de fonctions monotones transitives

Nous utiliserons ces fonctions pour tester les bornes inférieures que nous étudions

Théorème 2. *Fonction OR : est-ce que* $|x| \geq 1$ *?*

$$\begin{aligned} Q(f) &= \Omega(\sqrt{n}) \\ R(f) &= \Omega(n) \end{aligned}$$

Fonction MAJORITY : est-ce que $|x| \geq n/2$ *?*

$$\begin{aligned} Q(f) &= \Omega(n) \\ R(f) &= \Omega(n) \end{aligned}$$

Fonction THRESHOLD(g) :est-ce que $|x| \geq g(n)$ *?*

$$\begin{aligned} Q(f) &= \Omega(\sqrt{g(n)}) \\ R(f) &= \Omega(g(n)) \end{aligned}$$

4 Une preuve récente d'une borne inférieure probabiliste

La borne inférieure pour les fonctions monotones transitives en probabiliste était $\widetilde{\Omega}(n^{\frac{2}{3}})$ où le dénote la suppression des facteurs *log*. Cette nouvelle preuve apporte une vision différente du problème ainsi qu'un raffinement de la preuve puisque la borne prouvée est

Théorème 3 ([5]). *Soit* $f : \{0,1\}^n \rightarrow \{0,1\}$ *une fonction monotone transitive :*

$$R(f) \geq \Omega(\frac{1}{\sqrt{p(1-p)}} v^{\frac{4}{3}})$$

La preuve untilise deux lemmes :

Lemme 1 ([5]). *Soit une fonction* $f : \{0,1\}^n \rightarrow \{0,1\}$*, un arbre de décision* T *calculant* f*, pour tout paramètre* p *:*

$$Var(f,p) \leq \sum_{i=1}^{n} \delta_i(T,p) inf_i(f,p)$$

Lemme 2 ([5]). *Soit* f *une fonction booléenne monotone :*

$$R(f) \geq \frac{1}{p(1-p)} (Inf(f,p))^2$$

Démonstration. Considerons p_c tel que $Var(f,p_c) = 1$ cette valeur existe car f est monotone. De plus la fonction est transitive donc $inf_{max}(f,p) = \frac{inf(f,p)}{n}$

$$\begin{aligned} R(f)^2 &\geq \frac{n^2}{(\text{Inf}(f,p_c))^2} \\ R(f) &\geq \frac{1}{p_c(1-p_c)} (\text{Inf}(f,p_c))^2 \\ R(f)^3 &\geq n^2 p_c(1-p_c) \end{aligned}$$

□

4.1 preuve du Lemme 1

Démonstration. T est fixé pour le reste de la preuve. x et y seront des entrées distribuées selon μ dans toute la preuve.

Intuition : La preuve de ce lemme utilise une suite de mots sur $\{0,1\}^n$ pour modéliser un chemin entre x et y sur l'hypercube $\{0,1\}^n$ grâce à l'arbre de décision T.

Soit $J_i(x,y)$ les i premiers bits interrogés ainsi les bits qui ne seront jamais interrogés par l'arbre T sur l'entrée x. La suite de mots $u_i(x,y)$ est définie de cette façon :

$$\begin{aligned} u_i(x,y)[j] &= y[j] \text{ si } j \in J_i(x,y) \\ &= x[j] \text{ sinon} \end{aligned}$$

La suite $v_i(x,y)$ est définie de manière analogue à $u_i(x,y)$ mais en inversant x et y pour la définition des valeurs (sans changer $J_i(x,y)$) :

$$\begin{aligned} v_i(x,y)[j] &= x[j] \text{ si } j \in J_i(x,y) \\ &= y[j] \text{ sinon} \end{aligned}$$

Nous montrons par la suite que $u_i(x,y)$ et $v_i(x,y)$ sont distribués indépendamment selon μ. Par définition, $f(u_0(x,y)) = f(x)$ car tous les bits interrogés par T sur l'entrée x sont les même dans x que dans $u_0(x,y)$ et $u_n(x,y) = y$. Donc il découle par un argument d'inégalité triangulaire que :

$$Var(f,p) = E_{x,y:\mu}(f(x) \neq f(y)) \leq \sum_{i=1}^{n} E_{x,y:\mu}(f(u_{i-1}(x,y)) \neq f(u_i(x,y)))$$

Or $u_i(x,y) = u_{i-1}(x,y)^{j_i(x)})$.

$$Var(f,p) = E_{x,y:\mu}(f(x) \neq f(y)) \leq \sum_{i=1}^{n} \sum_{w} E_{x,y:\mu}(f(w) \neq f(w^{j_i(x)} | u_{i-1}(x,y) = w))$$

De plus $j_i(x) = j_{i-1}(v_{i-1})$ par définition.

$$Var(f,p) = E_{x,y:\mu}(f(x) \neq f(y)) \leq \sum_{i=1}^{n} \sum_{w,w'} E_{x,y:\mu}(f(w) \neq f(w^{j_i(w')} | u_{i-1}(x,y) = w \wedge v_{i-1}(x,y) = w')$$

D'après la proposition 1 $u_{i-1}(x,y)$ et $v_{i-1}(x,y)$ sont distribués indépendamment selon μ donc on peut les instancier par les variables aléatoire w et w'.

$$Var(f,p) \leq \sum_{i=1}^{n} E_{w,w':\mu}(f(x') \neq f(w^{j_i(w')})$$

soit $\delta_{i,j}$ la probabilité de lire le bit j à l'étape i de l'arbre de décision sur une entrée distribuée selon μ

$$\begin{aligned} Var(f,p) &\leq \sum_{i=1}^{n} \sum_{j=1}^{n} E_{x:\mu}(f(x) \neq f(x^j)) \delta_{i,j}(T,p) \\ &\leq \sum_{j=1}^{n} \sum_{i=1}^{n} inf_j(f,p) \delta_{i,j}(T,p) \end{aligned}$$

Mais par définition :

$$\sum_{i=1}^{n} \delta_{i,j}(T,p) = \delta_j(T,p)$$

Donc :

$$Var(f,p) \leq \sum_{j=1}^{n} inf_j(f,p)\delta_j(T,p)$$

□

Soit $inf_{max}(f,p) = \max_i inf_i(f,p)$ alors :

Corrolaire 1. *Soit* $f : \{0,1\}^n \to \{0,1\}$*, alors pour tout* p *:*

$$R(f) \geq \frac{Var(f,p)}{inf_{max}(f,p)}$$

Démonstration.

$$Var(f,p) \leq \sum_{j=1}^{n} inf_{max}(f,p)\delta_j(T,p)$$

Selon la définition de $\delta_j(T,p)$

$$\delta(T) \geq \frac{Var(f,p)}{inf_{max}(f,p)}$$

Cela étant vrai pour tout arbre T calculant f :

$$\Delta(f) \geq \frac{Var(f,p)}{inf_{max}(f,p)}$$

Or $\Delta(f)$ est la profondeur moyenne optimale des arbre de décision déterministe et probabiliste. ainsi $\Delta(f) \leq R(f)$.

$$R(f) \geq \frac{Var(f,p)}{inf_{max}(f,p)}$$

□

Soit $L(f,p) = \frac{Var(f,p)}{inf_{max}(f,p)}$ et $L(f) = \max_p L(f,p)$ dans la suite du rapport.

Proposition 1. *Si* x, y *sont distribués indépendamment selon* μ *alors* u_i *et* v_i *sont distribués indépendamment selon* μ

Démonstration. Soit $j_i(x,y)$ le j-ème bit lu sur l'entrée x dans l'arbre T. Montrons que si $j \in J_i(x,y)$ alors $j \in J_i(x^j, y')$.
Considérons le parcours dans l'arbre de décision sur l'entrée x jusqu'à ce qu'on lise le bit j, le bit j n'a pas encore été lu donc le parcours effectué sur l'entrée x^j est le même en conséquence le j-ème bit est lu au même moment sur l'entrée x et sur l'entrée x^j. Ainsi si le bit j est lu sur l'entrée x alors il est lu sur l'entrée x^j. Donc la fonction suivante est une bijection.

$$h_{i,j} : \{0,1\}^n \to \{0,1\}^n$$

$$\begin{aligned} (x,y) &\to (x, y^j) \text{ si } j \in J_i \\ &\to (x^j, y) \text{ sinon} \end{aligned}$$

En effet cette fonction est une bijection. Supposons que $h_{i,j}(x,y) = h_{i,j}(x',y')$ tel que $(x,y) \neq (x',y')$, alors $j \in J_i(x,y)$, $j \in J_i(x',y')$ et $x' = x^j$. Or ceci est en contradiction avec la propriété précédente, $(x,y) = (x',y')$, $h_{i,j}$ est une injection. Par argument cardinal $h_{i,j}$ est une bijection.

Soit w tel que $w[j] = 1$ alors

$$(1-p) \times P_{x,y:\mu}(u_i(x,y) = w) = p \times P_{x,y:\mu}(u_i(h_{i,j}(x,y)) = w)$$

Or comme $u_i(x,y)[j] = 1 - u_i(h_{i,j}(x,y))[j]$ par définition de $h_{i,j}$:

$$\begin{aligned} P_{x,y:\mu}(u_i(x,y)[j] = 1) &= p \times P_{x,y:\mu}(u_i(x,y)[j] = 1) + (1-p) \times P_{x,y:\mu}(u_i(x,y)[j] = 1) \\ &= p \times P_{x,y:\mu}(u_i(x,y)[j] = 1) + p \times P_{x,y:\mu}(u_i(x,y)[j] = 0) \\ &= p \end{aligned}$$

donc le j-ème bit vaut 1 avec une probabilité p indépendamment des autres bits. $u_i(x,y)$ est distribué selon μ.

Montrons désormais que u_i et v_i sont indépendants entre eux et que v_i est distribué selon μ. Par définition $\mu(x)\mu(y) = \mu(u_i(x,y))\mu(v_i(x,y))$

$$P_{x,y:\mu}(v_i(x,y) = w' \wedge u_i(x,y) = w) = \mu(w)\mu(w')$$

$$P_{x,y:\mu}(v_i(x,y) = w' \wedge u_i(x,y) = w) = \mu(w') \times P_{x,y:\mu}(u_i(x,y) = w)$$

donc la valeur de v_i ne dépend pas de la valeur de u_i car le membre de droite ne dépend que de w', ainsi les distribution de u_i et v_i sont indépendantes. De plus :

$$P_{x,y:\mu}(v_i(x,y) = w') = \mu(w)$$

Ainsi v_i est distribué selon μ. $\square$

4.2 Limitations de la borne $L(f)$

En évaluant la borne inférieure L(f) sur quelques fonctions usuelles je me suis aperçu que celle ci ne correspondait jamais avec les bornes inférieures déjà connue, cette nouvelle borne inférieure étant généralement bien en dessous.

Théorème 4.

$$\begin{aligned} L(OR) &= \Theta(1) \\ L(MAJORITY) &= \Theta(\sqrt{n}) \\ L(THRESHOLD(g)) &= \Theta(\sqrt{g(n)}) \end{aligned}$$

Pour prouver cette propriété nous nous limiterons a prouver la propriété pour la fonction THRESHOLD(g), la fonction OR et MAJORITY n'étant que des cas particuliers de cette dernière pour $g = 1$ et $g = n/2$.

La preuve se décomposera en trois lemmes selon les valeurs de p, et dans la preuve $g(n) < n/2$ pour des considérations de symétrie :

Lemme 3. *Soit f=THRESHOLD(g),si* $p = \frac{g(n)+h(n)g(n)}{n}$ *avec* $|h(n)| < (g(n))^{-\frac{1}{2}}$

$$L(f,p) < O(\sqrt{g(n)})$$

Démonstration.
$$L(f,p) = \frac{Var(f,p)}{inf_{max}(f,p)}$$
Or $Var(f,p) \leq 1$ d'où :
$$L(f,p) \leq \frac{1}{inf_{max}(f,p)}$$
L'ensemble de mots x tels que $f(x) \neq f(x^i)$ sont des mots qui contiennent $g(n)-1$ bits de valeur 1 hors du i-ème bit. Ainsi il y en a $\mathcal{C}_{n-1}^{g(n)-1}$ mots qui contiennent $g(n)-1$ bits de valeur 1 et dont le i-ème bit vaut 0 et le même nombre de mots qui contiennent $g(n)$ bits de valeur 1 et dont le i-ème bit vaut 1. Ces mots forment l'ensemble des mots tels que $f(x) \neq f(x^i)$. La fonction étant transitive :
$$inf_{max}(f,p) = inf_i(f,p)$$
Ainsi
$$\begin{aligned} inf_{max}(f,p) &= \mathcal{C}_{n-1}^{g(n)-1}(p^{g(n)-1}(1-p)^{n-g(n)+1} + p^{g(n)}(1-p)^{n-g(n)}) \\ &= \mathcal{C}_{n-1}^{g(n)-1}p^{g(n)-1}(1-p)^{n-g(n)} \end{aligned}$$
Considérons un équivalent de $\mathcal{C}_n^k$ grâce à la formule de Stierling $\mathcal{C}_n^k \sim \frac{1}{\sqrt{2\pi n}}\left(\frac{n}{e}\right)^n$:
$$\mathcal{C}_n^k = \Theta(\sqrt{\frac{k(n-k)}{n}}\frac{n^n}{k^k(n-k)^{n-k}})$$
Pour plus de lisibilité on pose :
$$\delta = \sqrt{\frac{n-1}{(g(n)-1)(n-g(n))}}$$
Ainsi en remplaçant $\mathcal{C}_{n-1}^{g(n)-1}$ par la formule obtenue ci-dessus
$$\begin{aligned} inf_{max}(f,p) &= \Theta(\delta\frac{(n-1)^{n-1}}{(g(n)-1)^{g(n)-1}(n-g(n))^{n-g(n)}}p^{g(n)-1}(1-p)^{n-g(n)}) \\ &= \Theta\left(\delta(1-\frac{1}{n})^{n-1}\left(\frac{np}{g(n)-1}\right)^{g(n)-1}\left(\frac{n-np}{n-g(n)}\right)^{n-g(n)}\right) \end{aligned}$$
On remplace par l'équivalent : $(1-\frac{1}{n})^{n-1} \sim \frac{1}{e}$
$$inf_{max}(f,p) = \Theta\left(\delta\left(\frac{np}{g(n)-1}\right)^{g(n)-1}\left(\frac{n-np}{n-g(n)}\right)^{n-g(n)}\right)$$
Nous allons désormais trouver des équivalent aux deux membres principaux en remplaçant p par sa valeur $\frac{g(n)(1+h(n))}{n}$:
$$inf_{max}(f,p) = \Theta\left(\delta\left(\frac{g(n)(1+h(n))}{g(n)-1}\right)^{g(n)-1}\left(\frac{n-g(n)(1+h(n))}{n-g(n)}\right)^{n-g(n)}\right)$$
Nous utilisons la forme $a = e^{ln(a)}$:
$$inf_{max}(f,p) = \Theta\left(\delta e^{ln(1+\frac{g(n)h(n)+1}{g(n)-1})(g(n)-1)+ln(1-\frac{g(n)h(n)}{n-g(n)})(n-g(n))}\right)$$

On remplace le logarithme par son développement limité à l'ordre 2 : $ln(1+h) = h - h^2/2 + o(h^2)$

$$inf_{max}(f,p) = \Theta\left(\delta e^{1 - \frac{(g(n)h(n)+1)^2}{2(g(n)-1)} - \frac{g(n)^2 h(n)^2}{2(n-g(n))} + o()}\right)$$

Désormais on utilise le fait que $h(n) = O\left(\frac{1}{\sqrt{g(n)}}\right)$ et que $g(n) \leq n/2$ pour majorer par une constante la valeur absolue de l'argument de l'exponentielle :

$$inf_{max}(f,p) = \Theta(\delta)$$

Ainsi en se ramenant à l'inégalité :

$$L(f,p) = O\left(\frac{1}{\delta}\right)$$
$$L(f,p) = O(\sqrt{g(n)})$$

□

Lemme 4. *Soit f=THRESHOLD(g),si* $p = \frac{g(n)+h(n)g(n)}{n}$ *avec* $h(n) \geq (g(n))^{-\frac{1}{2}}$

$$L(f,p) < O(\sqrt{g(n)})$$

Démonstration.

$$\begin{aligned} L(f,p) &= \frac{Var(f,p)}{inf_{max}(f,p)} \\ &\leq \frac{P_{x:\mu}(f(x)=0)}{inf_{max}(f,p)} \end{aligned}$$

La fonction f vaut 0 si $|x| < g(n)$ donc

$$P_{x:\mu}(f(x)=0) = \sum_{k=1}^{g(n)-1} \mathrm{C}_n^k p^k (1-p)^{n-k}$$

On remplace aussi inf_max en utilisant l'égalité :

$$L(f,p) \leq \sum_{k=1}^{g(n)-1} \frac{\mathrm{C}_n^k (1-p)^{g(n)-k}}{\mathrm{C}_{n-1}^{g(n)-1} p^{g(n)-k-1}}$$

Comme $\mathrm{C}_{n-1}^{g(n)-1} = \mathrm{C}_n^{g(n)} \frac{n-g(n)}{n} \geq \mathrm{C}_n^{g(n)}(1-p)$

$$L(f,p) \leq \sum_{k=1}^{g(n)-1} \frac{\mathrm{C}_n^k (1-p)^{g(n)-k-1}}{\mathrm{C}_n^{g(n)} p^{g(n)-k-1}}$$

On utilise l'inégalité suivante qui peut être montrée par récurrence aisément :

$$\frac{\mathrm{C}_n^k}{\mathrm{C}_n^{g(n)}} \leq \left(\frac{g(n)}{n-g(n)}\right)^{g(n)-k} \tag{1}$$

Ce qui donne.

$$L(f,p) \leq \sum_{k=1}^{g(n)-1} \left(\frac{g(n)(1-p)}{(n-g(n))p}\right)^{g(n)-k-1}$$

Dans cette partie de la preuve

$$p = \frac{g(n)(1+h(n))}{n} > \frac{g(n)(1+\frac{1}{\sqrt{g(n)}})}{n}$$

ce qui implique d'une part $\frac{n-p}{n-g(n)} < 1$ et ainsi :

$$L(f,p) \leq \sum_{k=1}^{g(n)-1} \left(\frac{g(n)}{n \times p}\right)^{g(n)-k-1}$$

et d'autre part ce la nous permet de remplacer p en gardant la majoration

$$L(f,p) < \sum_{k=1}^{g(n)-1} \left(\frac{1}{1+\frac{1}{\sqrt{g(n)}}}\right)^{g(n)-k-1}$$

On majore par la somme de la série géométrique d'argument $\frac{1}{1+\frac{1}{\sqrt{g(n)}}}$

$$\begin{aligned} Ł(f,p) &< \frac{1}{1-\frac{1}{1+\frac{1}{\sqrt{g(n)}}}} \\ &< \sqrt{g(n)}+1 \end{aligned}$$

□

Lemme 5. *Soit f=THRESHOLD(g),si* $p = \frac{g(n)+h(n)g(n)}{n}$ *avec* $h(n) \leq -(g(n))^{-\frac{1}{2}}$

$$L(f,p) < -\sqrt{g(n)}$$

Démonstration.

$$\begin{aligned} L(f,p) &= \frac{Var(f,p)}{inf_{max}(f,p)} \\ &\leq \frac{P_{x:\mu}(f(x)=1)}{inf_{max}(f,p)} \end{aligned}$$

La fonction f vaut 1 si $|x| \geq g(n)$ donc

$$P_{x:\mu}(f(x)=1) = \sum_{k=g(n)}^{n} \mathrm{C}_n^k p^k (1-p)^{n-k}$$

On remplace aussi inf_max en utilisant l'égalité :

$$L(f,p) \leq \sum_{k=g(n)}^{n} \frac{\mathrm{C}_n^k p^{k-g(n)}}{\mathrm{C}_{n-1}^{g(n)-1}(1-p)^{k+1-g(n)}}$$

Comme $\mathrm{C}_{n-1}^{g(n)-1} = \mathrm{C}_n^{g(n)} \frac{n-g(n)}{n} \geq \mathrm{C}_n^{g(n)} p$

$$L(f,p) \leq \sum_{k=g(n)}^{n} \frac{\mathrm{C}_n^k p^{k+1-g(n)}}{\mathrm{C}_n^{g(n)}(1-p)^{k+1-g(n)}}$$

On utilise l'inégalité (1)
Ce qui donne.

$$L(f,p) \leq \sum_{k=g(n)}^{n} \left(\frac{(n-g(n))p}{g(n)(1-p)} \right)^{k+1-g(n)}$$

Dans cette partie de la preuve

$$p = \frac{g(n)(1-h(n))}{n} < \frac{g(n)(1-\frac{1}{\sqrt{g(n)}})}{n}$$

ce qui implique d'une part $\frac{n-g(n)}{1-p)} < 1$ et ainsi :

$$L(f,p) \leq \sum_{k=g(n)}^{n-g(n)} \left(\frac{n \times p}{g(n)} \right)^{k+1-g(n)}$$

et d'autre part ce la nous permet de remplacer p en gardant la majoration

$$L(f,p) < \sum_{k=1}^{g(n)-1} \left(1 - \frac{1}{\sqrt{g(n)}} \right)^{g(n)-k-1}$$

On majore par la somme de la série géométrique d'argument $1 - \frac{1}{\sqrt{g(n)}}$

$$\begin{aligned} Ł(f,p) &< \frac{1}{1-1+\frac{1}{\sqrt{g(n)}}} \\ &< \sqrt{g(n)} \end{aligned}$$

□

5 Un outil récent

Ce théorème fait partie d'une nouvelles générations de méthodes qui obtiennent des bornes inférieures à la fois en quantique et en probabiliste.

Théorème 5 ([3])**.** *Soit une fonction booléenne f, soitp un distribution sur Σ, q une distribution de probabilité sur Σ^2 et $p'_{x,i}$ une famille distributions de probabilité sur Σ telles que si $f(x) = f(y)$ alors $q(x,y) = 0$. alors*

$$Q(f) \geq \Theta \left(\min_{i:x_i \neq y_i} \frac{\sqrt{p(x)p'_{x,i}(y)p(y)p'_{y,i}(x)}}{q(x,y)} \right)$$

$$R(f) \geq \Theta \left(\min_{i:x_i \neq y_i} \frac{\max(p(x)p'_{x,i}(y), p(y)p'_{y,i}(x))}{q(x,y)} \right)$$

A partir de ce genre de théorème, une recherche d'une pondération interessante est nécessaire pour trouver des évaluation sur des fonctions données. Ce genre de théorème crée de plus un lien entre la complexité en requêtes probabiliste et quantique, car une fois qu'une pondération est trouvée en quantique ou en probabiliste, l'équivalent probailiste ou quantique peut en découler.

ce que cette borne peut donner sur des fonctions monotones transitive :

Théorème 6 ([4]).

$$\begin{array}{rcl} Q(OR) & \geq & \Theta(\sqrt{n}) \\ R(OR) & \geq & \Theta(n) \end{array}$$

Démonstration.

$$\begin{array}{rcl} q(x,y) & = & \frac{1}{n} \text{ si } f(x) \neq f(y) \text{ et } y = s_i(x) \\ & = & 0 \text{ sinon} \\ p(x) & = & \frac{1}{2} \text{ si } si|x| = 0 \\ & = & \frac{1}{2n} \text{ si } |x| = 1 \\ & = & 0 \text{ sinon} \\ p'_{x,i}(y) & = & 1 \text{ si } y = s_i(x) \\ & = & 0 \text{ sinon} \end{array}$$

Tout ce qui est défini au dessus forme bien des distributions pour la fonction OR. Alors si $f(x) \neq f(x^i)$ cela veut dire que $|x| = 0$ et ainsi $p(x) = \frac{1}{2}$, $p(x^i) = \frac{1}{2n}$, $p'x, i(y) = 1$, $p'y, i(x) = 1$, donc

$$\begin{array}{rcl} \min\limits_{i:x_i \neq y_i} \frac{\sqrt{p(x)p'_{x,i}(y)p(y)p'_{y,i}(x)}}{q(x,y)} & = & \frac{\sqrt{\frac{1}{2n}\frac{1}{2}}}{\frac{1}{n}} \\ & = & \frac{\sqrt{n}}{2} \\ \min\limits_{i:x_i \neq y_i} \frac{\max(p(x)p'_{x,i}(y), p(y)p'_{y,i}(x))}{q(x,y)} & = & \frac{\max(\frac{1}{2n}, \frac{1}{2})}{\frac{1}{n}} \\ & = & \frac{n}{2} \end{array}$$

Ce qui montre que :

$$\begin{array}{rcl} Q(f) & \geq & \Theta(\sqrt{n}) \\ R(f) & \geq & \Theta(n) \end{array}$$

□

Théorème 7 ([4]).

$$\begin{array}{rcl} Q(MAJORITY) & \geq & \Theta(n) \\ R(MAJORITY) & \geq & \Theta(n) \end{array}$$

Démonstration. Soit $f = MAJORITY$:

$$\begin{aligned} q(x,y) &= \frac{1}{\mathrm{C}_n^{E(\frac{n}{2})} n} \text{ si } f(x) \neq f(y) \text{ et } y = s_i(x) \\ &= 0 \text{ sinon} \\ p(x) &= \frac{1}{\mathrm{C}_n^{E(\frac{n}{2})}} \text{ si } si|x| = E(\frac{n}{2}) \\ &= \frac{1}{\mathrm{C}_n^{E(\frac{n}{2})}} \text{ si } |x| = E(\frac{n}{2}) + 1) \\ &= 0 \text{ sinon} \\ p'_{x,i}(y) &= 1 \text{ si } y = s_i(x) \\ &= 0 \text{ sinon} \end{aligned}$$

Tout ce qui est défini au dessus forme bien des distributions pour la fonction MAJORITY. Alors si $f(x) \neq f(x^i)$ cela veut dire que $|x| = 0$ et ainsi $p(x) \sim \frac{1}{\sqrt{n}}$, $p(x^i) = \frac{1}{\sqrt{n}}$, $p'x, i(y) = 1$, $p'y, i(x) = 1$, donc

$$\begin{aligned} \min_{i:x_i \neq y_i} \frac{\sqrt{p(x)p'_{x,i}(y)p(y)p'_{y,i}(x)}}{q(x,y)} &= \Theta(n) \\ \min_{i:x_i \neq y_i} \frac{\max(p(x)p'_{x,i}(y), p(y)p'_{y,i}(x))}{q(x,y)} &= \Theta(n) \end{aligned}$$

Ce qui montre que :

$$\begin{aligned} Q(f) &\geq \Theta(n) \\ R(f) &\geq \Theta(n) \end{aligned}$$

□

Théorème 8 ([4])**.**

$$\begin{aligned} Q(THRESHOLD(g)) &\geq \Theta(\sqrt{g(n)n}) \\ R(THRESHOLD(g)) &\geq \Theta(n) \end{aligned}$$

Démonstration. Soit $f = THRESHOLD$:

$$\begin{aligned} q(x,y) &= \frac{1}{2\mathrm{C}_n^{g(n)} g(n)} \text{ si } f(x) \neq f(y) \text{ et } y = s_i(x) \\ q(x,y) &= \frac{1}{2\mathrm{C}_n^{g(n)-1} (n - g(n))} \text{ si } f(x) \neq f(y) \text{ et } y = s_i(x) \\ &= 0 \text{ sinon} \\ p(x) &= \frac{1}{2\mathrm{C}_n^{g(n)-1}} \text{ si } |x| = g(n) - 1 \\ &= \frac{1}{2\mathrm{C}_n^{g(n)}} \text{ si } |x| = g(n) \\ &= 0 \text{ sinon} \\ p'_{x,i}(y) &= 1 \text{ si } y = s_i(x) \\ &= 0 \text{ sinon} \end{aligned}$$

Tout ce qui est défini au dessus forme bien des distributions pour la fonction MAJORITY. Alors si $f(x) \neq f(x^i)$ cela veut dire que $|x| = 0$ et ainsi $p(x) \sim \frac{1}{\sqrt{n}}$, $p(x^i) = \frac{1}{\sqrt{n}}$, $p'x, i(y) = 1$, $p'y, i(x) = 1$, donc

$$\begin{aligned}
\min_{i:x_i \neq y_i} \frac{\sqrt{p(x)p'_{x,i}(y)p(y)p'_{y,i}(x)}}{q(x,y)} &= \Theta(\sqrt{g(n)n}) \\
\min_{i:x_i \neq y_i} \frac{\max(p(x)p'_{x,i}(y), p(y)p'_{y,i}(x))}{q(x,y)} &= \Theta(n)
\end{aligned}$$

Ce qui montre que :

$$\begin{aligned}
Q(f) &\geq \Theta(\sqrt{n \times g(n)}) \\
R(f) &\geq \Theta(n)
\end{aligned}$$

□

5.1 équivalent du lemme 2 en quantique

Théorème 9. *Soit $f : \{0,1\}^n \to \{0,1\}$, alors pour tout p*

$$Q(f) \geq \frac{\mathit{inf}(f,p)}{\sqrt{p(1-p)}}$$

Shi a déjà prouvé cette borne inférieure, nous allons la prouver de nouveau en utilisant des outils plus récents.

Démonstration.

$$\begin{aligned}
q(x,y) &= \frac{\mu(x)p^{1-x[i]}(1-p)^{x[i]}}{\mathit{inf}(f,p)} \text{ si } f(x) \neq f(y) \text{ et } y = s_i(x) \\
&= 0 \text{ sinon} \\
p(x) &= \mu(x) \\
p'_{x,i}(y) &= 1 \text{ si } y = s_i(x) \\
&= 0 \text{ sinon}
\end{aligned}$$

Par défintion les fonction définie ci dessus sont bien des distributions.
Soit x, i tels que $f(x) \neq f(s_i(x))$ alors :

$$\frac{\sqrt{p(x)p(s_i(x))}}{q(x, s_i(x))} = \frac{\mathit{inf}(f,p)}{\sqrt{p(1-p)}}$$

□

6 conclusion

Nous avons étudié une borne inférieure probabiliste traitant des propriétés de graphe monotones pour voir ce qui dans la preuve de cette borne pouvait se transposer au quantique. Mais il apparu qu'il est difficile d'utiliser d'avoir l'équivalent quantique de la première partie de cette preuve. Cependant ce lemme semble améliorable car nous avons démontré une limitation partielle de cette borne. Ensuite nous avons retrouvé l'équivalent quantique du second lemme en s'intéressant à un théorème permettant à la fois de trouver des bornes inférieures en quantique et en probabiliste en trouvant des pondérations adéquates sur les entrées.

La limitation de la borne inférieure suggère ainsi une possible amélioration de cette borne et ainsi de la borne générale sur les propriétés de graphe monotones. Si on arrivait à transférer la structure de la preuve au quantique, il pourrait apparître la raison plus précise de cette limitation de la borne inférieure en probabiliste. De cette façon le passage par le quantique nous ramener à une preuve plus claire et meilleure dans le cas probabiliste.

Ce stage m'a apporté une nouvelle vision sur les bornes inférieures. En effet j'ai vu de nombreuses preuves de bornes inférieures utilisant des réductions à d'autres problèmes connus pour déterminer leur diffculté. J'ai ainsi découvert des méthodes générales utilisant la difficulté qu'il existe à différencier deux entrées, qui est souvent appellée méthode adversaire. J'avais auparant vu ce critère en théorie des jeux, mais celui ci ne traitait pas le cas général et s'interessait juste à une fonction précise et ainsi un preuve ad hoc. Ces nouvelles méthodes font apparaître plus clairement une certaine topologie discrète des fonctions alors que j'avais l'habitude de regarder la structure de l'entrée plutot que la srtruture de la fonction.

Références

[1] D. Aharonov. Quantum computation – A review. In *Annual Review of Computational Physics*, volume VI. World Scientific, 1998.

[2] D. Aharonov and A. Ta-Shma. Adiabatic quantum state generation and statistical zero knowledge. These proceedings.

[3] S. Laplante and F. Magniez. Lower bounds for randomized and quantum query complexity using Kolmogorov arguments.*In Proceedings of 19th IEEE Conference on Computational Complexity*, pages 214-304, 2004.

[4] A. Ambainis. "Quantum lower bounds by quantum arguments". Journal of Computer and System Sciences, 64(2002), pages 750-767.

[5] R. O'Donnell, M. Saks, O. Schramm, R. Servedio. Every decision tree has an influential variable. FOCS 2005