Rapport : "A lower bound for monotone arithmetic circuits computing 0-1 permanent"

N. Perrin

25 novembre 2006

1 Introduction

Cet article, co-écrit en 1998 par R. Sengupta et H. Venkateswaran, a pour objectif de renforcer un résultat obtenu par M. Jerrum et M. Snir en 1982. Ce résultat est que dans certains semi-anneaux (pas de soustraction), le calcul par circuits du permanent d'une matrice $n \times n$ ($PERM((x_{i,j})_{1 \leq i,j \leq n}) = \sum_{\pi \in S_n} \prod_{i=1,n}(x_{i,\pi(i)})$) requiert une complexité (taille des circuits) exponentielle en n. Notons que ceci ne prouve pas que $VNP \neq VP$ (le calcul du permanent est un problème $VNP - complet$), car la soustraction pourrait être utilisée pour diminuer la taille des circuits. Dans l'article de Sengupta et Venkateswaran, les circuits introduits (appelés *circuits arithmétiques monotones*) sont sur basés sur le semi-anneau $\mathcal{R}$, et on ne s'autorise pas de constantes en entrées. Le théorème qu'ils démontrent est le suivant :

Tout circuit arithmétique à n^2 variables calculant un 0-1 permanent[1] a une taille supérieure à $n \times (2^{n-1} - 1)$.

Ce théorème ne se déduit pas trivialement du résultat de Jerrum et Snir, car de nombreux polynômes distincts peuvent coïncider si les variables valent 0 ou 1 (e.g. $x + y$ et $x^2 + y^2$). De plus, il est judicieux de s'intéresser au 0-1 permanent, puisque c'est généralement sur des matrices d'adjacences que le permanent est utilisé.

Les restrictions des modèles de calcul sont des méthodes couramment utilisées pour obtenir des résultats concernant des problèmes a priori inattaquables directement ; à la fin de l'article, justement, les auteurs discutent des conséquences d'une généralisation éventuelle de leur théorème à d'autres modèles de calcul.

[1] un 0-1 permanent est un polynôme qui coïncide avec le permanent si les valeurs prises par les variables sont 0 ou 1

2 Contenu

2.1 Définitions

– circuit arithmétique monotone : circuit arithmétiques classiques : avec des entrées, une sortie, des portes $\oplus, \otimes$ de degré entrant 2. La seule différence est qu'on autorise pas certaines entrées à être des constantes : les entrées sont toutes des variables. On définit également la taille du circuit (nombre de portes), sa profondeur, et on supposera que les circuits sont *alternants*, i.e. un fils d'une porte $\oplus$ (resp. $\otimes$) est une porte $\otimes$ (resp. $\oplus$) (tout circuit peut être changé en un circuit alternant équivalent de taille au plus double).

– parse-graphe : un parse-graphe d'un circuit est un sous graphe contenant la sortie et vérifiant :

1. si une porte $\oplus$ appartient au parse-graphe, alors exactement un de ses 2 prédécesseurs appartient au parse-graphe
2. si un porte $\otimes$ appartient au parse-graphe, alors ses 2 prédécesseurs aussi.

Les polynômes correspondant aux différents parse-graphes d'un circuit sont les monômes du polynôme calculé par le circuit.

– polynôme formel : c'est le polynôme calculé par le circuit, définit comme la somme des monômes correspondant aux différents parse-graphes. On remarquera que tous ces monômes ont un coefficient égal à 1.

2.2 La forme des polynômes calculant le 0-1 permanent

Après les définitions préliminaires, les auteurs montrent que le polynôme formel d'un circuit arithmétique monotone calculant un 0-1 permanent est nécessairement de la forme :

$$P((x_{i,j})_{1\leq i,j\leq n}) = \sum_{\pi\in S_n} \prod_{i=1,n} x_{i,\pi(i)}^{k_i}, \text{ où } k_i \in \mathbb{N}$$

Preuve : on sait que le polynôme formel est une somme de monômes ayant tous 1 pour coefficient ; donc si toutes les variables valent 1, le résultat est le nombre de monômes, à savoir $n!$; de plus, pour $\pi \in S_n$, si on choisit tous les $x_{i,j}$ nuls, sauf ceux de la forme $x_{i,\pi(i)}$, alors, comme le résultat du permanent est 1, un unique monôme doit valoir 1 : l'ensemble des variables de ce monôme est donc inclu dans $\{x_{i,\pi(i)}, i = 1..n\}$, et, en fait, égal à $\{x_{i,\pi(i)}, i = 1..n\}$ (car sinon on trouverait des entrées pour lesquelles ce polynôme vaut 1, alors que le permanent vaut 0). Comme il existe $n!$ permutations, les monômes et ces dernières sont en bijection, et on en conclut la forme du polynôme formel : $\sum_{\pi\in S_n} \prod_{i=1,n} x_{i,\pi(i)}^{k_i}$.

2.3 La preuve du théorème, obtenue en adaptant la démarche de Jerrum et Snir

Voici les différentes étapes de la preuve du théorème :

- Premièrement, on considère α, un $\otimes$-noeud d'un parse-graphe G : il "coupe" le parse-graphe en trois sous-graphes, non nécessairement disjoints : le fils gauche et ces ancêtres dans G (graphe $G_{\alpha,1}$), le fils droit et ses ancêtres G (graphe $G_{\alpha,2}$), tout les noeuds qui sont "ancêtres de la racine de G sans passer par α" (graphe $G_{\alpha,3}$), i.e. les noeuds β tels qu'il existe un chemin entre β et la racine du parse-graphe qui ne passe pas par α. On note alors d le nombre de variables présentes *uniquement* dans $G_{\alpha,1}$, $e \leq r - d$ le nombre de variables présentes *uniquement* dans $G_{\alpha,2}$ (r est le nombre de variables présentes dans sous-graphe de G induit par α), et $f \leq n - r$ le nombre de variables présentes *uniquement* dans $G_{alpha,3}$. Quitte à échanger les rôles de $G_{\alpha,1}$ et $G_{\alpha,2}$, on peut supposer $d \leq e \leq r - d$. Grâce à des arguments combinatoires, on montre que le nombre de parse-graphes dans lequel apparaît α, $m(\alpha)$ est tel que :
$$m(\alpha) \leq d!e!f! \leq d!(r-d)!(n-r)!$$
Donc, en notant $\mathcal{S}_\alpha$ l'ensemble des couples (r, d) possibles pour α (α peut apparaître dans des parse-graphes distincts, conduisant à des valeurs de r et d distinctes), on a donc :
$$m(\alpha) \leq min_{(r,d)\in\mathcal{S}_\alpha} d!(r-d)!(n-r)!$$

- Ensuite, on définit la notion de *stub-graphe* d'un parse-graphe, et le poids (W) d'un sous-graphe de parse-graphe. On prouve que si les G_i sont tous les parse-graphes de G, alors la taille de G est égale à $\sum_{i=1}^{n!} W(G_i)$. Puis on montre par induction sur le nombre de noeuds d'un stub-graphe, et en utilisant le lemme $m(\alpha) \leq \ldots$ obtenu précédemment, l'inégalité suivante :
Pour tout stub-graphe H d'un parse-graphe G,
$W(H) \geq \sum_{i=2}^{v(H)} 1/((n-i)!(i-1)!)$
(où $v(H)$ est le nombre de variables dans H — donc, par exemple, pour un parse-graphe G_i de circuit calculant un 0-1 permanent, comme le monôme calculé a n variables, $G_i = n$)

- Or, un parse-graphe est un stub-graphe, donc des 2 résultats précédents on conclut que pour un circuit de taille T calculant le 0-1 permanent, on a :
$T \leq \sum_{i=1}^{n!} \sum_{i=2}^{v(G_i)} 1/((n-i)!(i-1)!) = \sum_{i=1}^{n!} \sum_{i=2}^{n} 1/((n-i)!(i-1)!) =$
$n \times (2^{n-1} - 1)$

3 Conclusions

En guise de conclusion, les auteurs discutent de la possibilité d'étendre leur résultat aux *counting arithmetic circuits* (circuits où les entrées peuvent être x_i, $1 - x_i$, 1, 0), qui permettent des caractérisations de certaines classes de complexité (notamment $\#\mathcal{P}$ et $\#\mathcal{LOGCFL}$). La conséquence intéressante serait : $PERM \notin \mathcal{LOGCFL}$.

4 Accessibilité

L'article est court, lisible et accessible aux non-spécialistes. Les définitions sont clairement introduites, et les démonstrations claires. Quelques discussions auraient peut-être pu être rajoutées : pourquoi tous ces affaiblissements des modèles de calculs ? Il n'est pas assez clair que le théorème prouvé est strictement plus fort que celui de Jerrum et Snir : en effet ces derniers autorisent les constantes du semi-anneaux réel en entrée ; Sengupta et Venkateswaran ne les autorisent pas, et auraient donc dû prouver qu'elles ne sont d'aucune utilité pour le calcul du permanent.

5 Intérêt

Le théorème prouvé n'est pas exceptionnel : ce n'est qu'une petite extension d'un résultat assez ancien. Par ailleurs, il aura probablement peu de conséquences importantes. Mais c'est un résultat qui méritait d'être établi (ce qui est fait dans cet article, qui plus est de façon claire), car cela prouve qu'il n'est sûrement pas facile de contourner la difficulté du permanent en calculant un autre polynôme, qui coïnciderait lorsque les variables valent 0 ou 1. Il faudra donc probablement chercher d'autres angles d'attaque...